



リファレンス：文とオプション

SAP Sybase IQ 16.0 SP03

ドキュメント ID：DC01143-01-1603-01

改訂：2013 年 12 月

Copyright © 2013 by SAP AG or an SAP affiliate company. All rights reserved.

このマニュアルの内容を SAP AG による明示的な許可なく複製または転載することは、形態や目的を問わず禁じられています。ここに記載された情報は事前の通知なしに変更されることがあります。

SAP AG およびディストリビュータが販売しているソフトウェア製品には、他のソフトウェアベンダ独自のソフトウェアコンポーネントが含まれているものがあります。国内製品の仕様は変わることがあります。

これらの資料は SAP AG および関連会社 (SAP グループ) が情報のみを目的として提供するものであり、いかなる種類の表明または保証も行わないものではなく、SAP グループはこの資料に関する誤りまたは脱落について責任を負わないものとします。SAP グループの製品およびサービスに関する保証は、かかる製品およびサービスに付属している明確な保証文書がある場合、そこで明記されている保証に限定されます。ここに記載されているいかなる内容も、追加保証を構成するものとして解釈されるものではありません。

ここに記載された SAP および他の SAP 製品とサービス、ならびに対応するロゴは、ドイツおよび他の国における SAP AG の商標または登録商標です。その他の商標に関する情報および通知については、<http://www.sap.com/corporate-en/legal/copyright/index.epx#trademark> を参照してください。

目次

対象読者	1
SQL 文	3
一般的な SQL 構文要素	3
構文の表記規則	4
文の適応性インジケータ	5
SELECT クエリブロック内のロー制限句	5
ALLOCATE DESCRIPTOR 文 [ESQL]	7
ALTER AGENT 文	9
ALTER DATABASE 文	10
ALTER DBSPACE 文	13
ALTER DOMAIN 文	18
ALTER EVENT 文	20
ALTER FUNCTION 文	22
ALTER INDEX 文	24
ALTER LDAP SERVER 文	28
ALTER LOGICAL SERVER 文	31
ALTER LOGIN POLICY 文	32
ログインポリシーオプション	35
LDAP ログインポリシーオプション	39
マルチプレックスログインポリシーの設定	40
論理サーバへのアクセス許可設定	41
ALTER LS POLICY 文	42
ALTER MULTIPLEX RENAME 文	45
ALTER MULTIPLEX SERVER 文	46
ALTER PROCEDURE 文	48
ALTER ROLE 文	50
ALTER SEQUENCE 文	53
ALTER SERVER 文	54

ALTER SERVICE 文	57
ALTER SPATIAL REFERENCE SYSTEM 文	60
ALTER TABLE 文	67
ALTER TEXT INDEX 文	86
ALTER TEXT CONFIGURATION 文	87
ALTER TRIGGER 文	91
ALTER USER 文	92
ALTER VIEW 文	96
無効な従属ビューの特定と修正	99
BACKUP DATABASE 文	100
BEGIN ... END 文	108
BEGIN PARALLEL IQ ... END PARALLEL IQ 文	111
BEGIN TRANSACTION 文 [T-SQL]	113
CALL 文	115
CASE 文	118
CHECKPOINT 文	120
CLEAR 文 [Interactive SQL]	121
CLOSE 文 [ESQL] [SP]	122
COMMENT 文	123
COMMIT 文	129
CONFIGURE 文 [Interactive SQL]	132
CONNECT 文 [ESQL] [Interactive SQL]	133
CREATE AGENT 文	136
CREATE DATABASE 文	137
CREATE DBSPACE 文	150
CREATE DOMAIN 文	154
CREATE EVENT 文	157
CREATE EXISTING TABLE 文	163
CREATE EXTERNLOGIN 文	166
CREATE FUNCTION 文	169
CREATE FUNCTION 文 (Java UDF)	175
CREATE INDEX 文	179

CREATE LDAP SERVER 文	188
CREATE LOGICAL SERVER 文	192
CREATE LOGIN POLICY 文	194
ログインポリシーオプション	197
LDAP ログインポリシーオプション	201
マルチプレックスログインポリシーの設定	202
CREATE LS POLICY 文	203
CREATE MESSAGE 文 [T-SQL]	206
CREATE MULTIPLEX SERVER 文	208
CREATE PROCEDURE 文	210
プロシージャ内でのテナポラリテーブルの参 照	217
CREATE PROCEDURE 文 [T-SQL]	218
CREATE PROCEDURE 文 (外部プロシージャ)	221
CREATE PROCEDURE 文 (Java UDF)	229
CREATE PROCEDURE 文 (テーブル UDF)	231
CREATE ROLE 文	235
CREATE SCHEMA 文	238
CREATE SEQUENCE 文	239
CREATE SERVER 文	241
CREATE SERVICE 文	244
CREATE SPATIAL REFERENCE SYSTEM 文	247
CREATE SPATIAL UNIT OF MEASURE 文	255
CREATE TABLE 文	257
CREATE TEXT CONFIGURATION 文	276
CREATE TEXT INDEX 文	278
CREATE TRIGGER 文	279
CREATE USER 文	287
CREATE VARIABLE 文	289
CREATE VIEW 文	292
DEALLOCATE DESCRIPTOR 文 [ESQL]	295
宣言セクション [ESQL]	296

DECLARE 文	297
DECLARE CURSOR 文 [ESQL] [SP]	299
DECLARE CURSOR 文 [T-SQL]	305
DECLARE LOCAL TEMPORARY TABLE 文	306
DELETE 文	309
DELETE (位置付け) 文 [ESQL] [SP]	312
DESCRIBE 文 [ESQL]	314
DISCONNECT 文 [Interactive SQL]	318
DROP 文	319
DROP AGENT 文	323
DROP CONNECTION 文	324
DROP DATABASE 文	325
DROP EXTERNLOGIN 文	327
DROP LDAP SERVER 文	329
DROP LOGIN POLICY 文	330
DROP LOGICAL SERVER 文	331
DROP LS POLICY 文	332
DROP MULTIPLEX SERVER 文	333
DROP ROLE 文	335
DROP SEQUENCE 文	337
DROP SERVER 文	337
DROP SERVICE 文	338
DROP SPATIAL REFERENCE SYSTEM 文	340
DROP SPATIAL UNIT OF MEASURE 文	340
DROP STATEMENT 文 [ESQL]	341
DROP TEXT CONFIGURATION 文	343
DROP TEXT INDEX 文	344
DROP TRIGGER 文	345
DROP USER 文	346
DROP VARIABLE 文	347
EXECUTE 文 [ESQL]	348
EXECUTE 文 [T-SQL]	351

EXECUTE IMMEDIATE 文 [ESQL] [SP]	352
EXIT 文 [Interactive SQL]	355
FETCH 文 [ESQL] [SP]	357
FOR 文	361
FOR JSON 文	364
FORWARD TO 文	370
FROM 句	372
GET DESCRIPTOR 文 [ESQL]	380
GOTO 文 [T-SQL]	382
GRANT CHANGE PASSWORD 文	383
GRANT CONNECT 文	385
GRANT CREATE 文	388
GRANT オブジェクトレベル権限文	389
GRANT EXECUTE 文	391
GRANT INTEGRATED LOGIN 文	392
GRANT KERBEROS LOGIN 文	393
GRANT ROLE 文	394
GRANT SET USER 文	399
GRANT システム権限文	402
すべてのシステム権限のリスト	403
GRANT USAGE ON SEQUENCE 文	406
IF 文	407
IF 文 [T-SQL]	409
INCLUDE 文 [ESQL]	410
INSERT 文	412
INSTALL JAVA 文	421
IQ UTILITIES 文	424
LEAVE 文	428
LOAD TABLE 文	430
記憶領域サイズ	451
LOCK TABLE 文	452
LOOP 文	455

MESSAGE 文	457
OPEN 文 [ESQL] [SP]	461
OUTPUT 文 [Interactive SQL]	463
PARAMETERS 文 [Interactive SQL]	467
PREPARE 文 [ESQL]	469
PRINT 文 [T-SQL]	472
PUT 文 [ESQL]	474
RAISERROR 文 [T-SQL]	476
READ 文 [Interactive SQL]	478
REFRESH TEXT INDEX 文	480
RELEASE SAVEPOINT 文	483
REMOVE 文	484
RESIGNAL 文	486
RESTORE DATABASE 文	487
RESUME 文	495
RETURN 文	497
REVOKE CHANGE PASSWORD 文	499
REVOKE CONNECT 文	501
REVOKE CREATE 文	502
REVOKE EXECUTE 文	503
REVOKE INTEGRATED LOGIN 文	504
REVOKE KERBEROS LOGIN 文	505
REVOKE オブジェクトレベル権限文	506
REVOKE ROLE 文	508
REVOKE SET USER 文	510
REVOKE システム権限文	512
すべてのシステム権限のリスト	514
REVOKE USAGE ON SEQUENCE 文	517
ROLLBACK 文	518
ROLLBACK TO SAVEPOINT 文	519
ROLLBACK TRANSACTION 文 [T-SQL]	520
SAVEPOINT 文	521

SAVE TRANSACTION 文 [T-SQL]	522
SELECT 文	523
SET 文 [ESQL]	534
SET 文 [T-SQL]	536
SET CONNECTION 文 [ESQL] [Interactive SQL]	539
SET DESCRIPTOR 文 [ESQL]	540
SET OPTION 文	542
SET OPTION 文 [Interactive SQL]	545
SET SQLCA 文 [ESQL]	546
SETUSER 文	548
SIGNAL 文	550
START DATABASE 文 [Interactive SQL]	551
START ENGINE 文 [Interactive SQL]	553
START EXTERNAL ENVIRONMENT 文	554
START JAVA 文	555
STOP DATABASE 文 [Interactive SQL]	556
STOP ENGINE 文 [Interactive SQL]	557
STOP EXTERNAL ENVIRONMENT 文	558
STOP JAVA 文	559
TRIGGER EVENT 文	560
TRUNCATE 文	561
TRUNCATE TEXT INDEX 文	563
UNION 演算	564
UPDATE 文	566
UPDATE (位置付け) 文 [ESQL] [SP]	571
VALIDATE 文	573
VALIDATE LDAP SERVER 文	576
WAITFOR 文	579
WHENEVER 文 [ESQL]	582
WHILE 文 [T-SQL]	583
データベースオプション	585
データベースオプションの概要	585

現在のオプション設定	586
データベースオプションのスコープと継続期 間	587
temporary オプション	588
PUBLIC オプション	589
SECURITY オプション	589
SYSTEM オプション	589
オプション設定の削除	590
オプションの初期設定	591
廃止予定のデータベースオプション	592
一般的なデータベースオプション	592
データ抽出オプション	598
Transact-SQL 互換性オプション	598
Adaptive Server との互換性確保のための Transact-SQL オプションの設定	599
Interactive SQL オプション	600
アルファベット順のオプションリスト	601
AFFINITY_AUTOEXCLUDE_TIMEOUT オプ ション	601
AGGREGATION_PREFERENCE オプション ..	602
ALLOW_NULLS_BY_DEFAULT オプション [TSQL]	603
ALLOW_SNAPSHOT_VERSIONING オプシヨ ン	604
ANSI_CLOSE_CURSORS_ON_ROLLBACK オ プション [TSQL]	604
ANSI_PERMISSIONS オプション [TSQL]	605
ANSINULL オプション [TSQL]	606
ANSI_SUBSTRING オプション [TSQL]	607
ANSI_UPDATE_CONSTRAINTS オプション ...	608
ALLOW_READ_CLIENT_FILE オプション	610
ASE_BINARY_DISPLAY オプション	610

ASE_FUNCTION_BEHAVIOR オプション	611
AUDITING オプション [データベース]	612
BASE_TABLES_IN_RLV_STORE オプション ..	613
BIT_VECTOR_PINNABLE_CACHE_PERCEN T オプション	614
BLOCKING オプション	615
BLOCKING_TIMEOUT オプション	616
BT_PREFETCH_MAX_MISS オプション	616
BT_PREFETCH_SIZE オプション	617
BTREE_PAGE_SPLIT_PAD_PERCENT オプ ション	618
CACHE_AFFINITY_PERCENT オプション	619
CACHE_PARTITIONS オプション	620
CHAINED オプション [TSQL]	621
CHECKPOINT_TIME オプション	622
CIS_ROWSET_SIZE オプション	623
CLOSE_ON_ENDTRANS オプション [TSQL] ..	623
CONTINUE_AFTER_RAISERROR オプション [TSQL]	624
CONVERSION_ERROR オプション [TSQL] ...	625
CONVERSION_MODE オプション	626
CONVERT_VARCHAR_TO_1242 オプション ..	632
COOPERATIVE_COMMIT_TIMEOUT オプ ション	633
COOPERATIVE_COMMITS オプション	634
CREATE_HG_WITH_EXACT_DISTINCTS	635
CREATE_HG_AND_FORCE_PHYSICAL_DEL ETE	636
CURSOR_WINDOW_ROWS オプション	637
DATE_FIRST_DAY_OF_WEEK オプション	638
DATE_FORMAT オプション	639
DATE_ORDER オプション	641

DBCC_LOG_PROGRESS オプション	642
DBCC_PINNABLE_CACHE_PERCENT オプ ション	643
DEBUG_MESSAGES オプション	643
DEDICATED_TASK オプション	644
DEFAULT_DBSPACE オプション	645
DEFAULT_DISK_STRIPING オプション	647
DEFAULT_HAVING_SELECTIVITY_PPM オプ ション	647
DEFAULT_ISQL_ENCODING オプション [Interactive SQL]	648
DEFAULT_KB_PER_STRIPE オプション	649
DEFAULT_LIKE_MATCH_SELECTIVITY_PPM オプション	650
DEFAULT_LIKE_RANGE_SELECTIVITY_PPM オプション	651
DEFAULT_PROXY_TABLE_ROW_COUNT オ プション	652
DEFAULT_TABLE_UDF_ROW_COUNT オプ ション	652
DELAYED_COMMIT_TIMEOUT オプション	653
DELAYED_COMMITS オプション	654
DISABLE_RI_CHECK オプション	654
DIVIDE_BY_ZERO_ERROR オプション [TSQL]	655
DQP_ENABLED オプション	656
DQP_ENABLED_OVER_NETWORK オプシ ョ ン	657
EARLY_PREDICATE_EXECUTION オプション	658
ENABLE_ASYNC_IO オプション	659
ENABLE_LOB_VARIABLES オプション	660

EXTENDED_JOIN_SYNTAX オプション	661
FLOATING_POINT_ACCUMULATOR オプション	661
FORCE_DROP オプション	662
FORCE_NO_SCROLL_CURSORS オプション	663
FORCE_UPDATABLE_CURSORS オプション	664
FP_LOOKUP_SIZE オプション	665
FP_LOOKUP_SIZE_PPM オプション	666
FP_NBIT_AUTOSIZE_LIMIT オプション	667
FP_NBIT_ENFORCE_LIMITS オプション	668
FP_NBIT_IQ15_COMPATIBILITY オプション	670
FP_NBIT_LOOKUP_MB オプション	672
FP_NBIT_ROLLOVER_MAX_MB オプション	673
FP_PREDICATE_WORKUNIT_PAGES オプション	675
FPL_EXPRESSION_MEMORY_KB オプション	675
GARRAY_FILL_FACTOR_PERCENT オプション	676
GARRAY_INSERT_PREFETCH_SIZE オプション	677
GARRAY_PAGE_SPLIT_PAD_PERCENT オプション	678
GARRAY_RO_PREFETCH_SIZE オプション	679
HASH_PINNABLE_CACHE_PERCENT オプション	679
HASH_THRASHING_PERCENT オプション	680
HG_DELETE_METHOD オプション	681
HG_SEARCH_RANGE オプション	682
HTTP_SESSION_TIMEOUT オプション	683

IDENTITY_ENFORCE_UNIQUENESS オプション	684
IDENTITY_INSERT オプション	684
IN_SUBQUERY_PREFERENCE オプション ...	686
INDEX_ADVISOR オプション	687
INDEX_ADVISOR_MAX_ROWS オプション ...	690
INDEX_PREFERENCE オプション	691
INFER_SUBQUERY_PREDICATES オプション	692
IQGOVERN_MAX_PRIORITY オプション	693
IQGOVERN_PRIORITY オプション	694
IQGOVERN_PRIORITY_TIME オプション	694
ISOLATION_LEVEL オプション	695
JAVA_LOCATION オプション	696
JAVA_VM_OPTIONS オプション	696
JOIN_EXPANSION_FACTOR オプション	697
JOIN_OPTIMIZATION オプション	698
JOIN_PREFERENCE オプション	700
JOIN_SIMPLIFICATION_THRESHOLD オプション	702
LF_BITMAP_CACHE_KB オプション	703
LOAD_ZEROLENGTH_ASNULL オプション ...	704
LOG_CONNECT オプション	705
LOG_CURSOR_OPERATIONS オプション	705
LOG_DEADLOCKS オプション	706
LOGIN_MODE オプション	707
LOGIN_PROCEDURE オプション	708
MAIN_RESERVED_DBSPACE_MB オプション	709
MAX_CARTESIAN_RESULT オプション	709
MAX_CLIENT_NUMERIC_PRECISION オプション	710

MAX_CLIENT_NUMERIC_SCALE オプション	711
MAX_CUBE_RESULT オプション	712
MAX_CURSOR_COUNT オプション	713
MAX_HASH_ROWS オプション	714
MAX_IQ_THREADS_PER_CONNECTION オ プション	715
MAX_IQ_THREADS_PER_TEAM オプション	715
MAX_JOIN_ENUMERATION オプション	716
MAX_PARTITIONED_HASH_MB オプション ..	717
MAX_PREFIX_PER_CONTAINS_PHRASE オ プション	718
MAX_QUERY_PARALLELISM オプション	719
MAX_QUERY_TIME オプション	719
MAX_STATEMENT_COUNT オプション	720
MAX_TEMP_SPACE_PER_CONNECTION オ プション	721
MINIMIZE_STORAGE オプション	722
MIN_PASSWORD_LENGTH オプション	724
MIN_ROLE_ADMINS オプション	724
MONITOR_OUTPUT_DIRECTORY オプション	725
MPX_AUTOEXCLUDE_TIMEOUT オプション	726
MPX_HEARTBEAT_FREQUENCY オプション	727
MPX_IDLE_CONNECTION_TIMEOUT オプ ション	727
MPX_LIVENESS_TIMEOUT オプション	727
MPX_MAX_CONNECTION_POOL_SIZE オプ ション	728

MPX_MAX_UNUSED_POOL_SIZE オプション	729
MPX_WORK_UNIT_TIMEOUT オプション	730
NEAREST_CENTURY オプション [TSQL]	730
NOEXEC オプション	731
NON_ANSI_NULL_VARCHAR オプション	732
NON_KEYWORDS オプション [TSQL]	733
NOTIFY_MODULUS オプション	734
ODBC_DISTINGUISH_CHAR_AND_VARCHAR オプション	734
ON_CHARSET_CONVERSION_FAILURE オ プション	735
ON_ERROR オプション [Interactive SQL]	736
ON_TSQL_ERROR オプション [TSQL]	737
OS_FILE_CACHE_BUFFERING オプション	738
OS_FILE_CACHE_BUFFERING_TEMPDB オ プション	739
POST_LOGIN_PROCEDURE オプション	741
PRECISION オプション	742
PREFETCH オプション	743
PREFETCH_BUFFER_LIMIT オプション	744
PREFETCH_BUFFER_PERCENT オプション	744
PREFETCH_GARRAY_PERCENT オプション	745
PREFETCH_SORT_PERCENT オプション	745
PRESERVE_SOURCE_FORMAT オプション [database]	746
QUERY_DETAIL オプション	747
QUERY_NAME オプション	748
QUERY_PLAN オプション	748
QUERY_PLAN_AFTER_RUN オプション	749

QUERY_PLAN_AS_HTML オプション	750
QUERY_PLAN_AS_HTML_DIRECTORY オプ ション	752
QUERY_PLAN_MIN_TIME オプション	754
QUERY_PLAN_TEXT_ACCESS オプション ...	754
QUERY_PLAN_TEXT_CACHING オプション	755
QUERY_ROWS_RETURNED_LIMIT オプシヨ ン	756
QUERY_TEMP_SPACE_LIMIT オプション	757
QUERY_TIMING オプション	758
QUOTED_IDENTIFIER オプション [TSQL]	759
RECOVERY_TIME オプション	760
RESERVED_KEYWORDS オプション	760
RETURN_DATE_TIME_AS_STRING オプシヨ ン	761
REVERT_TO_V15_OPTIMIZER オプション ...	762
ROUND_TO_EVEN オプション	763
ROW_COUNT オプション	764
RV_AUTO_MERGE_EVAL_INTERVAL オプ ション	765
RV_MAX_ACTIVE_SUBFRAGMENT_COUNT オプション	766
RV_MERGE_NODE_MEMSIZE オプション	766
RV_MERGE_TABLE_MEMPERCENT オプ ション	767
RV_MERGE_TABLE_NUMROWS オプション	768
RV_RESERVED_DBSPACE_MB オプション ...	768
SCALE オプション	769
SNAPSHOT_VERSIONING オプション	770

SIGNIFICANTDIGITSFORDOUBLEEQUALITY オプション	771
SORT_COLLATION オプション	772
SORT_PINNABLE_CACHE_PERCENT オプ ション	773
SQL_FLAGGER_ERROR_LEVEL オプション [TSQL]	774
SQL_FLAGGER_WARNING_LEVEL オプシヨ ン [TSQL]	775
STRING_RTRUNCATION オプション [TSQL]	776
SUBQUERY_CACHING_PREFERENCE オプ ション	777
SUBQUERY_FLATTENING_PERCENT オプ ション	778
SUBQUERY_FLATTENING_PREFERENCE オ プション	779
SUBQUERY_PLACEMENT_PREFERENCE オ プション	780
SUPPRESS_TDS_DEBUGGING オプション	781
SWEEPER_THREADS_PERCENT オプション	782
TDS_EMPTY_STRING_IS_NULL オプション [database]	783
TEMP_EXTRACT_APPEND オプション	783
TEMP_EXTRACT_BINARY オプション	784
TEMP_EXTRACT_COLUMN_DELIMITER オ プション	785
TEMP_EXTRACT_DIRECTORY オプション ...	786
TEMP_EXTRACT_ESCAPE_QUOTES オプ ション	787

TEMP_EXTRACT_NAME _n オプション	788
TEMP_EXTRACT_NULL_AS_EMPTY オプ ション	790
TEMP_EXTRACT_NULL_AS_ZERO オプシ ョ	791
TEMP_EXTRACT_QUOTE オプション	792
TEMP_EXTRACT_QUOTES オプション	793
TEMP_EXTRACT_QUOTES_ALL オプシ ョ	794
TEMP_EXTRACT_ROW_DELIMITER オ プシ ョ	795
TEMP_EXTRACT_SIZE _n オプション	796
TEMP_EXTRACT_SWAP オプション	797
TEMP_RESERVED_DBSPACE_MB オ プシ ョ	798
TEMP_SPACE_LIMIT_CHECK オプシ ョ	799
TEXT_DELETE_METHOD オプション	800
TIME_FORMAT オプション	801
TIMESTAMP_FORMAT オプション	802
TOP_NSORT_CUTOFF_PAGES オプシ ョ	803
TRIM_PARTIAL_MBC オプション	804
TRUSTED_CERTIFICATES_FILE オ プシ ョ	805
TSQL_VARIABLES オプション [TSQL]	805
USER_RESOURCE_RESERVATION オ プシ ョ	806
VERIFY_PASSWORD_FUNCTION オ プシ ョ	807
WASH_AREA_BUFFERS_PERCENT オ プシ ョ	809
WAIT_FOR_COMMIT オプション	810
WD_DELETE_METHOD オプション	811

索引813

対象読者

このマニュアルは、SAP Sybase IQ SQL 文およびデータベースオプションに関するリファレンス資料を必要としている SAP® Sybase® IQ ユーザを対象としています。

言語要素、データ型、関数、システムプロシージャ、システムテーブルなど SAP Sybase IQ の他の側面に関する参考資料については、『リファレンス：ビルディングブロック、テーブル、およびプロシージャ』を参照してください。他のマニュアルでは、特定のタスクを実行する方法を詳しく説明します。このリファレンスマニュアルは、使用可能な SQL 構文、パラメータ、オプションなどの情報を参照するためのものです。コマンドラインユーティリティの起動パラメータについては、『ユーティリティガイド』を参照してください。

対象読者

SQL 文

SAP Sybase IQ で利用できる SQL 文について、Embedded SQL または Interactive SQL でのみ使用できるものも含めて説明します。

一般的な SQL 構文要素

ここでは、多くの SQL 構文で使われる言語要素をリストします。

ここで説明する要素の詳細については、『リファレンス：ビルディングブロック、テーブル、およびプロシージャ』の「SQL 言語要素」で、「識別子」、「検索条件」、「式」、および「文字列」を参照してください。

- `column-name` — カラム名を表す識別子。
- `condition` — 評価によって、TRUE、FALSE、UNKNOWN を判断する式。
- `connection-name` — アクティブな接続の名前を表す文字列。
- `data-type` — 記憶データ型。
- `expression` — 式。
- `filename` — ファイル名を指定した文字列。
- `host-variable` — 先頭にコロンがあるホスト変数として宣言される C 言語変数の 1 つ。
- `indicator-variable` — 通常のホスト変数の直後に置かれた `short int` 型の二次的なホスト変数。インジケータ変数の前にはコロンを付けてください。インジケータ変数は、データベースとの NULL 値の受け渡しに使用されます。
- `number` — 後に小数位が続いたり、前に負の記号が置かれたりする、任意のシーケンスに並べられた数字。必要に応じて、数字の後に 'e' を加え指数を続けることができます。たとえば、

```
42
-4.038
.001
3.4e10
1e-10
```

のようになります。

- `owner` — データベースオブジェクトを所有するユーザ ID を表す識別子。
- `role-name` — 外部キーの役割名を表す識別子。
- `savepoint-name` — セーブポイント名を表す識別子。
- `search-condition` — TRUE、FALSE、UNKNOWN を評価する条件。

SQL 文

- **special-value** – 『リファレンス：ビルディングブロック、テーブル、およびプロシージャ』の「SQL 言語要素」の「特別値」に記載されている特別値の1つ。
- **statement-label** – ループまたは複合文のラベルを表す識別子。
- **table-list** – テーブル名のリスト。関連名が含まれることもあります。詳細については、「FROM 句」を参照してください。
- **table-name** – テーブル名を表す識別子。
- **userid** – ユーザ名を表す識別子。ユーザ ID の大文字と小文字は区別されません。また、これはデータベースの CASE RESPECT プロパティの設定に影響されません。
- **variable-name** – 変数名を表す識別子。

参照：

- FROM 句 (372 ページ)

構文の表記規則

ここでは、SQL 構文の説明に使用する表記規則を示します。

- **キーワード** – SQL キーワードはすべて大文字で表記します。ただし、SQL キーワードの大文字と小文字は区別されないため、入力するときはどちらで入力してもかまいません。たとえば、SELECT は Select でも select でも同じです。
- **プレースホルダ** – 適切な識別子または式で置き換えられる項目は、*italics* で表記します。
- **継続** – 省略記号 (...) で始まる行は、直前の行から文が継続していることを示します。
- **オプション部分** – 文のオプション部分は角カッコで囲みます。次に例を示します。

```
RELEASE SAVEPOINT [ savepoint-name ]
```

この例では、*savepoint-name* がオプション部分です。角カッコは入力しないでください。

- **繰り返し項目** – 繰り返し項目のリストは、リストの要素の後ろに省略記号を付けて表します。1つ以上のリスト要素を指定できます。複数の要素を指定する場合、各要素間はカンマで区切る必要があります。次に例を示します。

```
UNIQUE ( column-name [ , ... ] )
```

この例では、カンマで区切って複数回 *column-name* を指定できることを示します。角カッコは入力しないでください。

- **選択肢** – オプションのうち1つを選択する必要がある場合は、その選択肢を大カッコで囲みます。次に例を示します。


```
[ QUOTES { ON | OFF } ]
```

この例では、QUOTES オプションを選択する場合に ON または OFF を指定する必要があることを示しています。中カッコは入力しないでください。

- **1つ以上のオプション** – 複数のオプションを選択する場合は、それらをカンマで区切ります。次に例を示します。

```
{ CONNECT, DBA, RESOURCE }
```

文の適応性インジケータ

一部の文は、どの場合に文を使用できるかを示すインジケータが角カッコで囲まれてタイトルの後ろに付きます。

このインジケータは以下のとおりです。

- [ESQL] – Embedded SQL で使用される文。
- [Interactive SQL] – Interactive SQL (**dbisql**) のみで使用される文。
- [SP] – ストアドプロシージャまたはバッチで使用される文。
- [T-SQL] – SAP Adaptive Server® Enterprise との互換性保持のために実装されている文。Transact-SQL® フォーマットではないストアドプロシージャでは、この文を使用できない場合があります。また、Transact-SQL の互換性が問題にならないかぎり、ISO/ANSI SQL 規格に近い代替りの文が推奨される場合もあります。

カッコが2組ある場合、その文はどちらの環境でも使用できます。たとえば、[ESQL] [SP] は Embedded SQL でもストアドプロシージャでも使用できる文であることを意味します。

SELECT クエリブロック内のロー制限句

FIRST 句、TOP 句、および LIMIT 句を使用すると、WHERE 句を満たすローのサブセットを返すことができます。FIRST 句、TOP 句、および LIMIT 句は、ORDER BY 句を含む SELECT クエリブロック内で使用できます。FIRST、TOP、および LIMIT は文内の最上位クエリブロックでしか使用できません。

FIRST 句、TOP 句、および LIMIT 句はロー制限句で、構文は次のとおりです。

row-limitation-option-1:

```
FIRST | TOP { ALL | limit-expression } [ START AT startat-expression ]
```

row-limitation-option-2:

```
LIMIT { [ offset-expression, ] limit-expression | limit-expression OFFSET offset-expression }
```

limit-expression : *simple-expression*

startat-expression : *simple-expression*

offset-expression : *simple-expression*

simple-expression:

```
integer
| variable
| ( simple-expression )
| ( simple-expression { + | - | * } simple-expression )
```

ロー制限句は、SELECT 句にのみ指定できます。これらの句を指定する場合は、ローの順序を意味のあるものにするために ORDER BY 句も指定する必要があります。

- row-limitation-option-1** – このタイプの句は SELECT クエリブロック内でのみ使用できます。TOP 引数と START AT 引数には、ホスト変数、整数定数、または整数変数を使用した簡単な算術演算を指定できます。TOP 引数は 0 以上の値、START AT 引数は 0 より大きい値に評価される必要があります。startat-expression が指定されていない場合は、デフォルトの 1 になります。

式 $\text{limit-expression} + \text{startat-expression} - 1$ は、 $9223372036854775807 = 2^{64} - 1$ 未満の値に評価される必要があります。TOP の引数が ALL の場合、startat-expression で始まるすべてのローが返されます。

TOP limit-expression START AT startat-expression 句は、LIMIT (startat-expression-1)、limit-expression、または LIMIT limit-expression OFFSET (startat-expression-1) と同等です。

- row-limitation-option-2** – このタイプの句は SELECT クエリブロック内でのみ使用できます。LIMIT 引数と OFFSET 引数には、ホスト変数、整数定数、または整数変数を使用した簡単な算術演算を指定できます。LIMIT 引数と OFFSET 引数は 0 以上の値に評価される必要があります。offset-expression が指定されていない場合、デフォルトは 0 です。式 $\text{limit-expression} + \text{offset-expression}$ は、 $9223372036854775807 = 2^{64} - 1$ 未満の値に評価される必要があります。

ロー制限句 LIMIT offset-expression, limit-expression は LIMIT limit-expression OFFSET offset-expression と同等です。どちらの構成も TOP limit-expression START AT (offset-expression + 1) と同等です。

LIMIT キーワードはデフォルトでは無効になっています。LIMIT キーワードを有効にするには、reserved_keywords オプションを使用します。

参照：

- SELECT 文 (523 ページ)

ALLOCATE DESCRIPTOR 文 [ESQL]

SQL 記述子領域 (SQLDA) に使用する領域を割り付けます。

クイックリンク：

「パラメータ」 (7 ページ)

「例」 (7 ページ)

「使用法」 (8 ページ)

「標準」 (9 ページ)

「パーミッション」 (9 ページ)

構文

```
ALLOCATE DESCRIPTOR descriptor-name  
... [ WITH MAX { integer | host-variable } ]
```

パラメータ

(先頭に戻る) (7 ページ)

- **WITH MAX** - 記述子領域の変数の数を指定できます。デフォルトのサイズは 1 です。

例

(先頭に戻る) (7 ページ)

- **例 1** - 次のサンプルプログラムには、**ALLOCATE DESCRIPTOR** 文の使用例が含まれています。

```
#include <stdio.h>  
#include <stdlib.h>  
#include <string.h>  
  
EXEC SQL INCLUDE SQLCA;  
  
#include <sqldef.h>  
  
EXEC SQL BEGIN DECLARE SECTION;  
int      x;  
short    type;  
int      numcols;  
char     string[100];
```

SQL 文

```
a_sql_statement_number stmt = 0;
EXEC SQL END DECLARE SECTION;

int main(int argc, char * argv[])
{
    struct sqllda *      sqllda1;

    if( !db_init( &sqlca ) ) {
        return 1;
    }

    db_string_connect(&sqlca, "UID=dba;PWD=sql;DBF=d:¥¥IQ-16_0¥
¥sample.db");

    EXEC SQL ALLOCATE DESCRIPTOR sqllda1 WITH MAX 25;

    EXEC SQL PREPARE :stmt FROM
        'select * from Employees';
    EXEC SQL DECLARE curs CURSOR FOR :stmt;
    EXEC SQL OPEN curs;

    EXEC SQL DESCRIBE :stmt into sqllda1;
    EXEC SQL GET DESCRIPTOR sqllda1 :numcols=COUNT;
        // how many columns?
    if( numcols > 25 ) {
        // reallocate if necessary
        EXEC SQL DEALLOCATE DESCRIPTOR sqllda1;
        EXEC SQL ALLOCATE DESCRIPTOR sqllda1
            WITH MAX :numcols;
    }
    type = DT_STRING; // change the type to string
    EXEC SQL SET DESCRIPTOR sqllda1 VALUE 2 TYPE = :type;
    fill_sqllda( sqllda1 ); // allocate space for the variables

    EXEC SQL FETCH ABSOLUTE 1 curs USING DESCRIPTOR sqllda1;
    EXEC SQL GET DESCRIPTOR sqllda1 VALUE 2 :string = DATA;

    printf("name = %s", string );

    EXEC SQL DEALLOCATE DESCRIPTOR sqllda1;
    EXEC SQL CLOSE curs;
    EXEC SQL DROP STATEMENT :stmt;

    db_string_disconnect( &sqlca, "" );
    db_fini( &sqlca );

    return 0;
}
```

使用法

(先頭に戻る) (7 ページ)

この文を使用する際は、事前に C コードで次のように宣言する必要があります。

```
struct sqllda * descriptor_name
```

さらに、フェッチを行ったり、記述子領域内のデータにアクセスしたりする文を実行する前には、`fill_sqlda` を呼び出して、実際のデータ項目に使用する領域を割り付ける必要があります。

標準

(先頭に戻る) (7 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Open Client/Open Server でサポートされています。

パーミッション

(先頭に戻る) (7 ページ)

なし

参照：

- DEALLOCATE DESCRIPTOR 文 [ESQL] (295 ページ)

ALTER AGENT 文

SAP Sybase IQ エージェントの接続情報を変更します。

クイックリンク：

「パラメータ」 (9 ページ)

「例」 (10 ページ)

「使用法」 (10 ページ)

「パーミッション」 (10 ページ)

構文

```
ALTER AGENT FOR MULTIPLEX SERVER server-name
```

```
alter-options
```

```
{ PORT portnum
  | USER username IDENTIFIED BY PASSWORD agentpwd, ... }
}
```

パラメータ

(先頭に戻る) (9 ページ)

SQL 文

- **alter-options** – SCC SAP Sybase IQ エージェントのポート、ユーザおよびパスワードを参照します。

適用対象

マルチプレックスのみ。

例

(先頭に戻る) (9 ページ)

- **例 1** – ユーザ smit のパスワードおよびポート番号の変更により、サーバ mpxdemo_svr2 のエージェントを変更します。

```
ALTER AGENT FOR MULTIPLEX SERVER mpxdemo_svr2 USER smith  
IDENTIFIED BY smith_pwd PORT 1112
```

使用法

(先頭に戻る) (9 ページ)

SYS.ISYSIQMPXSERVERAGENT システムテーブルには、サーバのエージェント接続定義が格納されています。

パーミッション

(先頭に戻る) (9 ページ)

MANAGE MULTIPLEX システム権限が必要です。

ALTER DATABASE 文

以前のバージョンのソフトウェアで作成されたデータベースをアップグレードしたり、Java や jConnect™ for JDBC™ のサポートを追加または削除したり、システムプロシージャ実行の管理を定義したりします。この文は DBISQL Interactive SQL とともに実行します。

クイックリンク：

「パラメータ」 (11 ページ)

「例」 (12 ページ)

「使用法」 (12 ページ)

「標準」 (13 ページ)

「パーミッション」 (13 ページ)

構文

```
ALTER DATABASE UPGRADE
  [ PROCEDURE ON ]
  [ JCONNECT { ON | OFF } ]
  [ RESTART { ON | OFF } ]
  [ SYSTEM PROCEDURE AS DEFINER {ON| OFF} ]
```

パラメータ

(先頭に戻る) (10 ページ)

- **PROCEDURE** – データベース内のすべての dbo 所有プロシージャと sys 所有プロシージャを削除して再作成します。
- **JCONNECT** – SAP Sybase IQ jConnect JDBC ドライバを使用してシステムカタログ情報にアクセスできるようにするには、ON を指定する必要があります。これによって、jConnect システムテーブルおよびプロシージャがインストールされます。jConnect システムオブジェクトを除外するには、OFF を指定します。その場合でも、システムカタログ情報にアクセスしないかぎり、JDBC を使用できます。デフォルトは、jConnect サポートのインクルード (JCONNECT ON) です。
- **RESTART** – ON (デフォルト) が指定されており、AutoStop 接続パラメータが NO に設定されている場合、アップグレード後にデータベースが再起動します。それ以外の場合、アップグレード後にデータベースが停止します。
- **SYSTEM PROCEDURE AS DEFINER** – 権限付きシステムプロシージャが invoker (プロシージャの実行者) と definer (プロシージャの所有者) のいずれの権限で実行されるかを定義します。
 - **OFF** – すべての権限付きシステムプロシージャが、invoker の権限で実行されます。sp_proc_priv() を使用して、システムプロシージャの実行に必要なシステム権限を指定します。
 - **ON (デフォルト)、または指定しない** –
 - 16.0 より前のデータベースにアップグレードする場合は、16.0 より前の権限付きシステムプロシージャは、definer の権限で実行されると 16.0 以降の権限付きシステムプロシージャは、invoker の権限で実行されるです。
 - バージョン 16.0 以降のデータベースにアップグレードする場合、デフォルトはアップグレードするデータベースの動作です。

注意： アップグレード後に実行モデルを変更すると、システムプロシージャに対する EXECUTE 権限を明示的に付与したカスタムストアドプロシージャとアプリケーション上の機能が失われる可能性があります。また、システムプロシージャを実行する機能にも影響する可能性があります。『リファレンス：ビ

ルディングブロック、テーブル、およびプロシージャ』の「システムプロシージャ」を参照してください。

を定義します。

例

(先頭に戻る) (10 ページ)

- **例 1** – jConnect サポートを無効にします。

```
ALTER DATABASE UPGRADE JCONNECT OFF
```

使用法

(先頭に戻る) (10 ページ)

ALTER DATABASE 文は、以前のバージョンのソフトウェアで作成されたデータベースをアップグレードします。これは、メジャーリリースと同様にメンテナンスリリースにも適用されます。

データベースをアップグレードすると、SAP Sybase IQ によって次の変更が行われます。

- 現在のバージョンのシステムテーブルのアップグレード。
- 新規のデータベースオプションの追加。
- 現在のバージョンの新機能の有効化。

また、そのデータベースが現在のバージョンのソフトウェアで作成されている場合は、**ALTER DATABASE UPGRADE** を使用して jConnect の機能を容易に追加できます。

注意：

- アップグレード前のバックアップの推奨事項については、『インストールおよび設定ガイド』を参照してください。
- **ALTER DATABASE UPGRADE** を実行する前に、サーバを起動するときにユーザ接続を制限する方法で起動したことを確認してください。手順とその他のアップグレードに関する注意事項については、使用するプラットフォームの『移行』ガイドを参照してください。
- **iqunload** ユーティリティを使用して、15.0 よりも前のバージョンで作成されたデータベースをアップグレードします。プラットフォームの『移行』ガイドを参照してください。

ALTER DATABASE UPGRADE の使用後は、データベースを停止します。

関連する動作：

- オートコミット

標準

(先頭に戻る) (10 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Adaptive Server ではサポートされていません。

パーミッション

(先頭に戻る) (10 ページ)

ALTER DATABASE システム権限が必要です。

参照：

- CREATE DATABASE 文 (137 ページ)

ALTER DBSPACE 文

既存の DB 領域に対し、読み書きのモードの変更、サイズの変更、または領域の拡張を行います。

クイックリンク：

「パラメータ」 (14 ページ)

「例」 (16 ページ)

「使用法」 (17 ページ)

「標準」 (18 ページ)

「パーミッション」 (18 ページ)

構文

```
ALTER DBSPACE dbspace-name
{ ADD new-file-spec [, new-file-spec ... ]
| DROP FILE logical-file-name [, FILE logical-file-name ... ]
| RENAME TO newname | RENAME 'new-file-pathname'
| READONLY | READWRITE
| ONLINE | OFFLINE
| STRIPING{ ON | OFF }
| STRIPESIZEKB size-in-KB
ALTER FILE file-name
{ READONLY | [ FORCE ] READWRITE }
| SIZE file-size [ KB | MB | GB | TB ]
| ADD file-size [ KB | MB | GB | TB | PAGES ] }
```

```

RENAME PATH 'new-file-pathname'
RENAME TO newname

new-file-spec - (back to Syntax)
  FILE logical-file-name 'file-path' iq-file-opts

iq-file-opts - (back to new-file-spec)
  [ [ SIZE ] file-size ]
  ...[ KB | MB | GB | TB ] ]
  [ RESERVE reserve-size [ KB | MB | GB | TB ] ]

```

パラメータ

(先頭に戻る) (13 ページ)

- **ADD** – 指定した DB 領域に 1 つまたは複数のファイルを追加します。各ファイルに対して、DB ファイル名と物理ファイルパスは必要かつユニークでなければなりません。ファイルは IQ メイン、IQ 共有テンポラリ、IQ テンポラリ、またはキャッシュの各 DB 領域に追加できます。ファイルを読み込み専用の DB 領域に追加できますが、DB 領域は読み込み専用のままになります。ファイルをマルチプレックス共有テンポラリ DB 領域に追加できますが、読み込み専用モードの場合のみです (ADD FILE のデフォルト)。

カタログ DB 領域には 1 つのファイルしか含むことができないため、ADD FILE をカタログ DB 領域で使用できない場合があります。

RLV DB 領域の場合は、シンプレックスサーバのみで ADD FILE を使用します。マルチプレックス RLV DB 領域にはファイルを追加できません。

キャッシュ DB 領域の場合は、マルチプレックスサーバまたはシンプレックスサーバで ADD FILE を使用します。

ALTER FILE 句で使用する場合、ファイルのサイズを、ページ、キロバイト (KB)、メガバイト (MB)、ギガバイト (GB)、テラバイト (TB) の単位で拡張します。デフォルトは MB です。ファイルのサイズは、フリーリスト (アロケーションマップ) に十分な余裕があり、DB 領域に十分な領域が確保されていなければ追加できません。

- **DROP FILE** – 指定したファイルを IQ DB 領域から削除します。ファイルは空である必要があります。最後のファイルを指定の DB 領域から削除することはできません。DB 領域に 1 つのファイルしか含まれていない場合は、DROP DBSPACE を代用します。
- **RENAME TO** – DROP FILE 句とともに使用すると、単一ファイルを含む DB 領域のパス名を変更します。RENAME PATH 句と意味的に同じです。DB 領域に複数のファイルが含まれている場合はエラーが返されます。

IQ_SYSTEM_MAIN、IQ_SYSTEM_MSG、IQ_SYSTEM_TEMP、
IQ_SHARED_TEMP、または SYSTEM の名前は変更できません。

ALTER FILE 句とともに使用すると、指定されたファイルの論理名を新しい名前に変更します。新しい名前はデータベース内でユニークにします。

- **READONLY** – DROP 句とともに使用すると、IQ_SYSTEM_MAIN、IQ_SYSTEM_TEMP、IQ_SYSTEM_MSG、IQ_SHARED_TEMP、および SYSTEM 以外の DB 領域を読み込み専用に変更します。DB 領域に現在割り当てられたオブジェクトへの DML 変更を禁止します。キャッシュ DB 領域および IQ メインストア内の DB 領域に対してのみ使用できます。

ALTER FILE 句とともに使用すると、指定したファイルを読み込み専用に変更します。このファイルは、IQ メイン DB 領域に関連付けられている必要があります。IQ_SHARED_TEMP のファイルは READONLY ステータスに変更することはできません。

- **READWRITE** – DROP FILE 句とともに使用すると、DB 領域を読み書き用に変更します。DB 領域はオンラインである必要があります。キャッシュ DB 領域および IQ メインストア内の DB 領域に対してのみ使用できます。

ALTER FILE 句とともに使用すると、指定したキャッシュ DB 領域、IQ メインまたはテンポラリストアの DB 領域を読み書き用に変更します。このファイルは、キャッシュ DB 領域、IQ メインまたはテンポラリの DB 領域に関連付けられている必要があります。

- **ONLINE** – オフライン DB 領域と関連するすべてのファイルをオンラインにします。キャッシュ DB 領域内および IQ メインストア内の DB 領域に対してのみ使用できます。
- **OFFLINE** – オンライン読み込み専用 DB 領域と関連するすべてのファイルをオフラインにします (DB 領域が読み書き用である、すでにオフラインになっている、あるいはキャッシュ DB 領域または IQ メインストア内にはない場合は、エラーが返されます)。キャッシュ DB 領域内または IQ メインストア内の DB 領域に対してのみ使用できます。
- **STRIPING** – DB 領域のディスクストライピングを指定どおりに変更します。ディスクストライピングがオンに設定されている場合、データは DB 領域内の各ファイルからラウンドロビン方式で割り付けられます。たとえば、最初に書き込みがあったデータベースページが最初のファイルへ、2 番目に書き込みがあったページが指定の DB 領域内の次のファイルへ、というようになります。読み込み専用の DB 領域はスキップされます。

- **STRIPESIZEKB** – ディスクストライピングアルゴリズムが指定した DB 領域の次のストライプに移動する前に、各ファイルに書き込むデータ量をキロバイト (KB) で指定します。
- **FORCE READWRITE** – ALTER FILE 句とともに使用すると、セカンダリノードに既知のファイルステータスの問題がある場合でも、指定した共有テンポラリストアの DB ファイルのステータスを読み書き用に変更します。ファイルは、IQ メイン、共有テンポラリ、またはテンポラリの DB 領域に関連付けられている可能性があります。IQ_SYSTEM_MAIN とユーザメインの新しい DB ファイルは読み書き用に作成されるので、この句は、共有テンポラリ DB 領域にのみ影響します。
- **SIZE** – ファイルの新しいサイズを、キロバイト (KB)、メガバイト (MB)、ギガバイト (GB)、テラバイト (TB) の単位で指定します。デフォルトはメガバイトです。DB 領域のサイズは、フリーリスト (アロケーションマップ) に十分な余裕があり、DB 領域に十分な領域が確保されていないと増加できません。DB 領域のサイズを減少できるのは、切り取られる部分が未使用である場合だけです。
- **RENAME PATH** – ALTER FILE 句とともに使用すると、指定したファイルに関連付けられたファイルのパス名を変更します。この句はファイルを古いパスの代わりに新しいファイルパスに関連付けるだけであり、オペレーティングシステムのファイル名を実際に変更するわけではありません。ファイル名を変更するには、オペレーティングシステム経由で実行する必要があります。ファイルのパス名を変更するには、DB 領域がオフラインである必要があります。新しいパスは、DB 領域をオンラインで変更するか、データベースを再起動する場合に使用されます。

新しいパスにアクセスできない場合は、データベースを起動できないため、IQ_SYSTEM_MAIN 内でファイルのパス名を変更する必要はありません。

IQ_SYSTEM_MAIN 内のファイルのパス名を変更する必要がある場合は、ファイルを読み込み専用を設定し、ファイルを空にして削除してから、ファイルに新しいパス名を付けて追加し直します。DB ファイルへの物理ファイルパスは、一重引用符で囲みます。

例

(先頭に戻る) (13 ページ)

- **例 1** – DspHist という名前の DB 領域のモードを、READONLY に変更します。

```
ALTER DBSPACE DspHist READONLY
```
- **例 2** – サイズが 500MB のファイル FileHist3 を追加して、DB 領域 DspHist に 500MB を追加します。

```
ALTER DBSPACE DspHist
ALTER FILE FileHist3 ADD 500MB
```

- **例 3** – UNIX システムで、2つの 500MB ファイルを DB 領域 DspHist に追加します。

```
ALTER DBSPACE DspHist ADD
FILE FileHist3 '/History1/data/file3' SIZE 500MB,
FILE FileHist4 '/History1/data/file4' SIZE 500
```

- **例 4** – DB 領域 IQ_SYSTEM_TEMP のサイズを、2GB だけ増加します。

```
ALTER DBSPACE IQ_SYSTEM_TEMP ADD 2 GB
```

- **例 5** – 2つのファイルを DB 領域 DspHist から削除します。ファイルはどちらも空である必要があります。

```
ALTER DBSPACE DspHist
DROP FILE FileHist2, FILE FileHist4
```

- **例 6** – DB 領域 IQ_SYSTEM_MAIN のサイズを、1000 ページだけ増加します (ADD 句のデフォルトはページです)。

```
ALTER DBSPACE IQ_SYSTEM_MAIN ADD 1000
```

- **例 7** – ファイルをキャッシュ DB 領域 myDAS に追加します。

```
ALTER DBSPACE myDAS ADD FILE iqdas2 'sampledb.iqcache' size 1024
```

- **例 8** – DB ファイル iqdas2 をキャッシュ DB 領域 myDAS から削除します。

```
ALTER DBSPACE myDAS DROP FILE iqdas2
```

- **例 9** – キャッシュ DB 領域 myDAS を無効にします。

```
ALTER DBSPACE myDAS OFFLINE
```

- **例 10** – myDAS キャッシュ DB 領域の DB ファイル iqdas2 を読み込み専用にします。

```
ALTER DBSPACE myDAS ALTER FILE iqdas2 READONLY
```

使用法

(先頭に戻る) (13 ページ)

ALTER DBSPACE は、読み書きのモードの変更、オンライン/オフライン状態の変更、ファイルサイズの変更、DB 領域名の変更、ファイルの論理名またはファイルパスの変更、または DB 領域ストライピングパラメータの設定を行います。既存の DB 領域の詳細については、**sp_iqdbspace** プロシージャ、**sp_iqdbspaceinfo** プロシージャ、**sp_iqfile** プロシージャ、**sp_iqdbspaceobjectinfo**、および **sp_iqobjectinfo** を実行してください。DB 領域名と DB ファイル名では、大文字と小文字は常に区別されません。CASE RESPECT を指定してデータベースが作成され、大文字と小文字が区別されるファイルがオペレーティングシステムでサポートされている場

SQL 文

合、物理ファイルパスの大文字と小文字は区別されます。そうでない場合、ファイルパスの大文字と小文字は区別されません。

DB 領域と DB ファイルの名前は、引用符で囲まないか、二重引用符で囲みます。

Windows でパスを指定する場合、円記号 (¥) の後に n または x がある場合は円記号を 2 つ重ねます。こうすることで、SQL の文字列の規則に従って、改行文字 (¥n) または 16 進数字 (¥x) として解釈されるのを回避できます。円記号は常にエスケープした方が安全です。

関連する動作：

- オートコミット
- 自動チェックポイント
- モードを READONLY に変更すると、DB 領域上のデータベースの内部構造が読み書き用であるいずれかの DB 領域へ直ちに移動されます。

標準

(先頭に戻る) (13 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Adaptive Server ではサポートされていません。

パーミッション

(先頭に戻る) (13 ページ)

MANAGE ANY DBSPACE システム権限が必要です。

参照：

- CREATE DATABASE 文 (137 ページ)
- CREATE DBSPACE 文 (150 ページ)
- DROP 文 (319 ページ)

ALTER DOMAIN 文

ユーザ定義のドメインまたはデータ型の名前を変更します。Java データ型の名前は変更しません。

クイックリンク：

「パラメータ」 (19 ページ)

「例」 (19 ページ)

「使用法」 (19 ページ)

「パーミッション」 (19 ページ)

構文

```
ALTER { DOMAIN | DATATYPE } user-type  
    RENAME new-name
```

パラメータ

(先頭に戻る) (18 ページ)

- **new-name** – 新しいドメイン名を表す識別子。
- **user-type** – 名前が変更されるユーザ定義のドメインのデータ型。

例

(先頭に戻る) (18 ページ)

- **例 1** – Address ドメイン名を MailingAddress に変更します。

```
ALTER DOMAIN Address RENAME MailingAddress
```

使用法

(先頭に戻る) (18 ページ)

ALTER DOMAIN 文は、SYSUSERTYPE システムテーブルのユーザ定義のドメインまたはデータ型の名前を更新します。

そのユーザ定義のドメインまたはデータ型を参照するすべてのプロシージャ、ビュー、またはイベントを再作成する必要があります。そうしないと、それらは以前の名前を参照します。

関連する動作：

- オートコミット

パーミッション

(先頭に戻る) (18 ページ)

ドメインを作成したデータベースユーザにするか、ALTER DATATYPE または ALTER ANY OBJECT システム権限が必要です。

参照：

- CREATE DOMAIN 文 (154 ページ)

ALTER EVENT 文

イベント、または事前定義のアクションを自動化するイベント関連ハンドラの定義を変更します。また、スケジュールされたアクションの定義も変更します。

クイックリンク：

「パラメータ」 (21 ページ)

「例」 (21 ページ)

「使用法」 (21 ページ)

「パーミッション」 (22 ページ)

構文

```
ALTER EVENT event-name
  [ DELETE TYPE | TYPE event-type ]
  { WHERE { trigger-condition | NULL }
  | { ADD | [ MODIFY ] | DELETE } SCHEDULE schedule-spec }
  [ ENABLE | DISABLE ]
  [ [ MODIFY ] HANDLER compound-statement | DELETE HANDLER ]
```

event-type - (構文に戻る)

```
BackupEnd
| "Connect"
| ConnectFailed
| DatabaseStart
| DBDiskSpace
| "Disconnect"
| GlobalAutoincrement
| GrowDB
| GrowLog
| GrowTemp
| IQMainDBSpaceFree
| IQTempDBSpaceFree
| LogDiskSpace
| "RAISERROR"
| ServerIdle
| TempDiskSpace
```

trigger-condition - (構文に戻る)

```
event_condition( condition-name )
{ =
| <
| >
| !=
| <=
| >= } value
```

schedule-spec - (構文に戻る)


```
[ schedule-name ]
{ START TIME start-time | BETWEEN start-time AND end-time }
[ EVERY period { HOURS | MINUTES | SECONDS } ]
[ ON { ( day-of-week, ... ) | ( day-of-month, ... ) } ]
[ START DATE start-date ]
```

パラメータ

(先頭に戻る) (20 ページ)

- **DELETE TYPE** – イベントとイベントタイプとの関連付けを削除します。
- **ADD | MODIFY | DELETE SCHEDULE** – スケジュールの定義を変更します。1 つの ALTER EVENT 文で変更できるスケジュールは 1 つだけです。
- **WHERE** – イベントを起動する条件を定義します。WHERE NULL オプションは条件を削除します。

注意： その他のパラメータの説明については、「CREATE EVENT 文」を参照してください。

例

(先頭に戻る) (20 ページ)

- **例 1** – システムテーブル SYSEVENT に対してクエリを実行して、イベント名をリストします。

```
SELECT event_id, event_name FROM SYS.SYSEVENT
```

- **例 2** – システムテーブル SYSSCHEDULE に対してクエリを実行して、スケジュール名をリストします。

```
SELECT event_id, sched_name FROM SYS.SYSSCHEDULE
```

使用法

(先頭に戻る) (20 ページ)

ALTER EVENT では、CREATE EVENT で作成されたイベント定義を変更できません。以下の用法が考えられます。

- 開発期間にイベントハンドラを変更する。
- 開発段階では、イベントハンドラをトリガ条件やスケジュールなしで定義およびテストし、イベントハンドラが完成した後で、ALTER EVENT を使用して実行の条件を追加する。
- イベントを無効にすることによって、イベントハンドラを一時的に無効にする。

SQL 文

ALTER EVENT を使用してイベントを変更するときは、イベント名と任意でスケジュール名を指定します。

イベントにはユニークなイベント ID が存在します。イベントと関連するスケジュールの対応付けには、SYSEVENT と SYSSCHEDULE の event_id カラムを使用します。

関連する動作：

- オートコミット

パーミッション

(先頭に戻る) (20 ページ)

次のいずれかが必要です。

- MANAGE ANY EVENT システム権限
- ALTER ANY OBJECT システム権限

参照：

- BEGIN ... END 文 (108 ページ)
- CREATE EVENT 文 (157 ページ)

ALTER FUNCTION 文

既存の関数を変更します。変更した関数全体を ALTER FUNCTION 文にインクルードします。

クイックリンク：

「パラメータ」 (23 ページ)

「使用法」 (23 ページ)

「標準」 (23 ページ)

「パーミッション」 (23 ページ)

構文

構文 1

```
ALTER FUNCTION [ owner.]function-name function-definition
```

function-definition :

```
CREATE FUNCTION syntax
```

構文 2

```
ALTER FUNCTION [ owner.]function-name
SET HIDDEN
```

構文 3

```
ALTER FUNCTION [ owner.]function-name
RECOMPILE
```

パラメータ

(先頭に戻る) (22 ページ)

- **SET HIDDEN** – 関連する関数の定義を暗号化し、解読できないようにします。この関数はアンロードして、他のデータベースに再ロードできます。

警告！ SET HIDDEN 句の設定は、元に戻すことができません。元のソースを再度必要とする場合は、データベースの外部に保持しておく必要があります。

- **RECOMPILE** – ユーザ定義の関数を再コンパイルします。関数を再コンパイルすると、カタログに格納された定義が再解析され、構文が検証されます。保持されている関数のソースは、再コンパイルしても変わりません。関数を再コンパイルすると、SET HIDDEN 句でスクランブルをかけた定義は、スクランブルがかかった状態が維持され、解読できません。

使用法

(先頭に戻る) (22 ページ)

- **構文 1** – 最初の単語を除き、**CREATE FUNCTION** 文の構文とまったく同じです。**CREATE FUNCTION** 文のいずれかのバージョンを変更できます。

関数に対する既存のパーミッションはそのまま維持されます。このため、パーミッションの再割り当ては必要ありません。**DROP FUNCTION** と **CREATE FUNCTION** を実行した場合は、実行パーミッションを再割り当てする必要があります。

関連する動作：

- オートコミット

標準

(先頭に戻る) (22 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。

パーミッション

(先頭に戻る) (22 ページ)

SQL 文

Watcom SQL または Transact-SQL 関数を変更します。以下のいずれかが必要です。

- ALTER ANY PROCEDURE システム権限
- ALTER ANY OBJECT システム権限
- その関数を所有している

外部 C/C++ スカラまたは集合、あるいは、外部 Java 関数を変更します。次のいずれかが必要です。

- CREATE EXTERNAL REFERENCE システム権限が必要です。
- 次のいずれかも必要です。
 - ALTER ANY PROCEDURE システム権限
 - ALTER ANY OBJECT システム権限
 - その関数を所有している

参照：

- ALTER PROCEDURE 文 (48 ページ)
- CREATE FUNCTION 文 (169 ページ)
- DROP 文 (319 ページ)

ALTER INDEX 文

ベーステーブルまたはグローバルテンポラリテーブルのインデックスの名前、およびインデックスと外部キーがユーザによって明示的に作成された場合の外部キーのロール名を変更したり、カタログ保存テーブル上でインデックスのクラスタ化された内容を変更したりします。キー制約を適用するために作成したインデックスの名前は変更できないという点に注意してください。

クイックリンク：

「パラメータ」 (25 ページ)

「例」 (25 ページ)

「使用法」 (26 ページ)

「標準」 (26 ページ)

「パーミッション」 (26 ページ)

構文

```
ALTER { INDEX index-name
      | [ INDEX ] FOREIGN KEY role-name
      | [ INDEX ] PRIMARY KEY
```

```

| ON [owner.]table-name { rename-clause | move-clause | cluster-clause }

rename-clause - (構文に戻る)
  RENAME TO | AS new-name

move-clause - (構文に戻る)
  MOVE TO dbspace-name

cluster-clause - (構文に戻る)
  CLUSTERED | NONCLUSTERED

```

パラメータ

(先頭に戻る) (24 ページ)

- **ON** – 名前を変更するインデックスや外部キーを含むテーブルの名前を指定します。
- **RENAME** – インデックスまたは外部キーロールの新しい名前を指定します。
- **MOVE** – 指定したインデックス、一意性制約、外部キー、またはプライマリキーを指定した DB 領域に移動します。一意性制約キーまたは外部キーでは、一意なインデックス名を指定する必要があります。
- **cluster-clause** – インデックスを **CLUSTERED** と **NONCLUSTERED** のどちらに変更するかを指定します。カタログストアテーブルのみに適用され、特定のテーブル上で 1 つのインデックスのみ、クラスタ化できます。

例

(先頭に戻る) (24 ページ)

- **例 1** – プライマリキー (c5 の HG) を DB 領域 Dsp4 から Dsp8 に移動します。

```

CREATE TABLE foo (
  c1 INT IN Dsp1,
  c2 VARCHAR(20),
  c3 CLOB IN Dsp2,
  c4 DATE,
  c5 BIGINT,
  PRIMARY KEY (c5) IN Dsp4) IN Dsp3);
CREATE DATE INDEX c4_date ON foo(c4) IN Dsp5;
ALTER INDEX PRIMARY KEY ON foo MOVE TO Dsp8;

```

- **例 2** – DATE インデックスを Dsp5 から Dsp9 に移動します。

```
ALTER INDEX c4_date ON foo MOVE TO Dsp9
```

- **例 3** – テーブル `jal.mytable` のインデックス `COL1_HG_OLD` の名前を `COL1_HG_NEW` に変更します。

```
ALTER INDEX COL1_HG_OLD ON jal.mytable  
RENAME AS COL1_HG_NEW
```

- **例 4** - テーブル dba.Employees の外部キーのロール名 ky_dept_id を emp_dept_id に変更します。

```
ALTER INDEX FOREIGN KEY ky_dept_id  
ON dba.Employees  
RENAME TO emp_dept_id
```

使用法

(先頭に戻る) (24 ページ)

新しい DB 領域に対する CREATE 権限があり、かつテーブル所有者であるか、MANAGE ANY DBSPACE システム権限がある必要があります。

注意: ローカルテンポラリテーブルのインデックスを変更しようとする、「インデックスが見つかりません。」というエラーが返されます。デフォルトインデックス (FP) などユーザ作成でないインデックスを変更しようとする、「インデックスを変更できません。所有者のタイプが USER のベーステーブルまたはグローバルテンポラリテーブルのインデックスのみを変更できます。」というエラーが返されます。

関連する動作:

- オートコミット。Interactive SQL の、[結果] ウィンドウ枠の [結果] タブがクリアされます。現在接続しているすべてのカーソルを閉じます。

標準

(先頭に戻る) (24 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - Adaptive Server ではサポートされていません。

パーミッション

(先頭に戻る) (24 ページ)

move-clause マテリアライズドビューの場合は、次のいずれかが必要です。

- MANAGE ANY DBSPACE システム権限
- ALTER ANY INDEX システム権限
- ALTER ANY OBJECT システム権限
- マテリアライズドビューを所有し、さらに次のいずれかがある。
 - CREATE ANY OBJECT システム権限

- ターゲット DB 領域に対する CREATE 権限

move-clause その他のインデックスの場合は、次のいずれかが必要です。

- MANAGE ANY DBSPACE システム権限
- ALTER ANY INDEX システム権限
- ALTER ANY OBJECT システム権限
- 基本となるテーブルを所有しているか、次のいずれかとともに REFERENCES 権限がある
 - CREATE ANY OBJECT システム権限
 - ターゲット DB 領域に対する CREATE 権限

マテリアライズドビューの **cluster-clause** は以下のいずれかを必要とします。

- ALTER ANY INDEX システム権限
- ALTER ANY OBJECT システム権限
- マテリアライズドビューの所有者である。

他のすべてのインデックスの **cluster-clause** は以下のいずれかを必要とします。

- ALTER ANY INDEX システム権限
- ALTER ANY OBJECT システム権限
- そのテーブルに対する REFERENCES 権限
- テーブルの所有者である。

それ以外のすべての句は次のいずれかを必要とします。

- ALTER ANY INDEX システム権限
- ALTER ANY OBJECT システム権限
- そのテーブルに対する REFERENCES 権限
- 基本となるテーブルの所有者である。

参照：

- ALTER TABLE 文 (67 ページ)
- CREATE INDEX 文 (179 ページ)
- CREATE TABLE 文 (257 ページ)

ALTER LDAP SERVER 文

LDAP サーバ設定オブジェクトへの変更は以降の接続に適用されます。変更が適用されたときにすでに開始していた接続に対しては、変更がすぐに反映されることはありません。

クイックリンク：

「パラメータ」 (28 ページ)

「例」 (30 ページ)

「使用法」 (30 ページ)

「標準」 (30 ページ)

「パーミッション」 (30 ページ)

構文

```
ALTER LDAP SERVER ldapua-server-name
  { ldapua-server-attrs
    | [ WITH ( SUSPEND | ACTIVATE | REFRESH ) ] }
```

ldapua-server-attrs - (構文に戻る)

```
SEARCH DN
  URL { 'URL_string' | NULL }
  | ACCESS ACCOUNT { 'DN_string' | NULL }
  | IDENTIFIED BY ( 'password' | NULL )
  | IDENTIFIED BY ENCRYPTED { encrypted-password | NULL }
  | AUTHENTICATION URL { 'URL_string' | NULL }
  | CONNECTION TIMEOUT timeout_value
  | CONNECTION RETRIES retry_value
  | TLS { ON | OFF }
```

パラメータ

(先頭に戻る) (28 ページ)

- **URL** – 指定されたユーザ ID のホスト (名前または IP アドレスで指定)、ポート番号、および DN ルックアップで実行される検索を指定します。この値は、ISYSLDAPSERVER システムテーブルに格納される前に、LDAP URL 構文が正しいかどうかを検証されます。この文字列の最大サイズは 1024 バイトです。
- **ACCESS ACCOUNT** – SAP Sybase IQ 内のユーザではなく、SAP Sybase IQ が使用するために LDAP サーバで作成されたユーザ。このユーザの識別名 (DN) は、LDAP サーバへの接続に使用されます。このユーザは、SEARCH DN URL で指定された場所でユーザ ID によって DN を検索するためのパーミッションを、

LDAP サーバ内に保持しています。この文字列の最大サイズは 1024 バイトです。

- **IDENTIFIED BY – ACCESS ACCOUNT** ユーザに関連付けられたパスワードを指定します。このパスワードは、対称暗号化を使用してディスクに保存されます。パスワードを解除して何も設定しない場合は、値 NULL を指定します。クリアテキストのパスワードの最大サイズは 255 バイトです。
- **IDENTIFIED BY ENCRYPTED – ACCESS ACCOUNT** に指定されている識別名に関連付けられた暗号化形式のパスワードを設定します。バイナリ値は暗号化されたパスワードであるため、ディスクにそのまま保存されます。パスワードを解除して何も設定しない場合は、値 NULL を指定します。バイナリの最大サイズは 289 バイトです。暗号化キーは有効な varbinary 値である必要があります。暗号化キーは、引用符で囲まないでください。
- **AUTHENTICATION URL** – ユーザの認証に使用する LDAP サーバのホスト (名前または IP アドレスで指定) とポート番号を指定します。これは、URL_string として定義された値で、ISYSLDAPSERVER に格納される前に、LDAP URL 構文が正しいかどうかを検証されます。事前の DN 検索によって得られたユーザの DN とユーザパスワードによって、新しい接続が認証 URL にバインドされます。LDAP サーバへの正常な接続は、接続ユーザの ID の証明とみなされます。この文字列の最大サイズは 1024 バイトです。
- **CONNECTION TIMEOUT** – DN 検索と認証の両方に使用する SAP Sybase IQ から LDAP サーバへの接続のタイムアウトを指定します。この値はミリ秒で指定します。デフォルト値は 10 秒です。
- **CONNECTION RETRIES** – DN 検索と認証の両方に使用する SAP Sybase IQ から LDAP サーバへの接続の再試行回数を指定します。有効な値の範囲は 1 ~ 60 で、デフォルト値は 3 です。
- **TLS** – DN 検索と認証の両方に使用する LDAP サーバへの接続に、TLS とセキュア LDAP プロトコルのいずれを使用するかを定義します。ON に設定すると、TLS プロトコルが使用され、URL は "ldap://" で始まります。OFF に設定すると (または指定しないと)、セキュア LDAP プロトコルが使用され、URL は "ldaps://" で始まります。TLS プロトコルを使用する場合は、LDAP サーバによって使用される証明書に署名する認証局 (CA) の証明書が含まれているファイル名を使用して、データベースセキュリティオプション TRUSTED_CERTIFICATES_FILE を指定します。
- **WITH ACTIVATE** – LDAP サーバ設定オブジェクトを有効にして、作成時にすぐに使用できるようにします。これによって、1 つの文で LDAP ユーザ認証の

定義と有効化を行うことができます。WITH ACTIVATE を使用すると、LDAP サーバ設定オブジェクトのステータスは READY に変わります。

例

(先頭に戻る) (28 ページ)

- **例 1** – apps_primary という名前の LDAP サーバ設定オブジェクトを中断します。

```
ALTER LDAP SERVER apps_primary SUSPEND
```

- **例 2** – apps_primary という名前の LDAP サーバ設定オブジェクトがホスト fairfax 上の別の URL を認証に使用するように変更し、ポート番号を 1066 に設定し、接続試行回数を 10 に設定して、最後に LDAP サーバ設定オブジェクトを有効化します。

```
ALTER LDAP SERVER apps_primary  
AUTHENTICATION URL 'ldap://my_LDAPserver:1066/'  
CONNECTION RETRIES 10  
WITH ACTIVATE
```

使用法

(先頭に戻る) (28 ページ)

属性の LDAP サーバ設定オブジェクト値のリセットに加えて、**ALTER LDAP SERVER** 文により、管理者はサーバのステータスと動作を手動で調整できます。これは、LDAP サーバ設定オブジェクトをメンテナンスモードにしたり、メンテナンスモードからサービスモードに戻すことによって行います。

標準

(先頭に戻る) (28 ページ)

ANSI SQL – 準拠レベル：Transact-SQL® 拡張。

パーミッション

(先頭に戻る) (28 ページ)

MANAGE ANY LDAP SERVER システム権限が必要です。

ALTER LOGICAL SERVER 文

データベース内の既存のユーザ定義による論理サーバの設定を変更します。次の文では、論理サーバで共有される物理ノード間で、一貫した共有システムテンポラリストア設定が強制的に適用されます。

クイックリンク：

「パラメータ」 (31 ページ)

「例」 (31 ページ)

「使用法」 (32 ページ)

「パーミッション」 (32 ページ)

構文

```
ALTER LOGICAL SERVER logical-server-name
    { alter-ls-clause } [ WITH STOP SERVER ]

alter-ls-clause - (back to Syntax)
    { ADD MEMBERSHIP '(' { ls-member, ... } ')' |
      DROP MEMBERSHIP '(' { ls-member, ... } ')' |
      POLICY policy-name }

ls-member - (back to alter-ls-clause)
    FOR LOGICAL COORDINATOR | mpx-server-name
```

パラメータ

(先頭に戻る) (31 ページ)

- **logical-server-name** – 既存のユーザ定義論理サーバの名前を指します。
- **WITH STOP SERVER – TEMP_DATA_IN_SHARED_TEMP** データベースオプションが直接的または間接的に変更された場合に、論理サーバ内のすべてのサーバを自動的に停止します。

適用対象

マルチプレックスのみ。

例

(先頭に戻る) (31 ページ)

- **例 1** – マルチプレックスノード *n1* と *n2* を論理サーバ *ls1* に追加して、ユーザ定義の論理サーバを変更します。

```
ALTER LOGICAL SERVER ls1 ADD MEMBERSHIP (n1, n2)
```

- **例 2** – コーディネータの論理メンバーシップを追加し、現在のコーディネータノード n1 の名前付きメンバーシップを論理サーバ ls1 から削除します。

```
ALTER LOGICAL SERVER ls1 ADD MEMBERSHIP (FOR LOGICAL COORDINATOR)  
ALTER LOGICAL SERVER ls1 DROP MEMBERSHIP (n1)
```

- **例 3** – 論理サーバ ls2 の論理サーバポリシーを、ポリシー lsp1 に変更します。

```
ALTER LOGICAL SERVER ls2 POLICY lsp1
```

使用法

(先頭に戻る) (31 ページ)

SYS.ISYSIQLSMEMBER システムテーブルには、論理サーバメンバーシップの定義が格納されています。

論理サーバに追加、または論理サーバから削除されたメンバーノードは、そのノードで **ALTER LOGICAL SERVER** に対応する TLV ログが適用されるまで、論理サーバ接続の受け入れ、または拒否をしません。論理サーバの既存の接続は、ノードが論理サーバから削除されてもそのノードで稼動し続けますが、これらの接続の分散処理は停止します。

次の場合には、エラーが返されます。

- ADD MEMBERSHIP 句で指定された ls-member のいずれかが、すでに論理サーバのメンバーである場合。
- DROP MEMBERSHIP 句で指定された ls-member のいずれかが、論理サーバの既存のメンバーでない場合。
- 論理サーバのメンバーシップ変更により、あるノードが単一のログインポリシーに割り当てられた複数の論理サーバに属することになる場合。1つのログインポリシー内で論理サーバのメンバーシップの重複は許可されません。

パーミッション

(先頭に戻る) (31 ページ)

MANAGE MULTIPLEX システム権限が必要です。

ALTER LOGIN POLICY 文

既存のログインポリシーを変更、または論理サーバアクセスを設定します。

クイックリンク：

「パラメータ」 (34 ページ)

「例」 (34 ページ)

「使用法」 (34 ページ)

「パーミッション」 (35 ページ)

構文

構文 1

```
ALTER LOGIN POLICY policy-name
  { { ADD | DROP | SET } LOGICAL SERVER ls-assignment-list
    [ LOGICAL SERVER ls-override-list ] }
```

ls-assignment-list - (back to Syntax 1)

```
{ { ls-name, ... }
  | ALL
  | COORDINATOR
  | SERVER
  | NONE
  | DEFAULT }
```

ls-override-list - (back to Syntax 1)

```
{ ls-name, ... }
```

ls-name - (back to *ls-assignment-list*) or (back to *ls-override-list*)

```
{ OPEN | user-defined-ls-name }
```

構文 2

```
ALTER LOGIN POLICY policy-name policy-option
```

policy-option - (back to Syntax 2)

```
policy-option-name = policy-option-value
```

policy-option-name - (back to *policy-option*)

```
AUTO_UNLOCK_TIME
| CHANGE_PASSWORD_DUAL_CONTROL
| DEFAULT_LOGICAL_SERVER
| LOCKED
| MAX_CONNECTIONS
| MAX_DAYS_SINCE_LOGIN
| MAX_FAILED_LOGIN_ATTEMPTS
| MAX_NON_DBA_CONNECTIONS
| PASSWORD_EXPIRY_ON_NEXT_LOGIN
| PASSWORD_GRACE_TIME
| PASSWORD_LIFE_TIME
| ROOT_AUTO_UNLOCK_TIME
| LDAP_PRIMARY_SERVER
| LDAP_SECONDARY_SERVER
| LDAP_AUTO_FAILBACK_PERIOD
| LDAP_FAILOVER_TO_STD
| LDAP_REFRESH_DN
```

policy-option-value - (back to *policy-option*)

```
{ UNLIMITED | DEFAULT | value }
```

パラメータ

(先頭に戻る) (32 ページ)

- **policy-name** – ログインポリシーの名前。ルートを指定してルートログインポリシーを修正します。
- **policy-option-name** – ポリシーオプションの名前。各オプションの詳細については、「ログインポリシーオプション」と「LDAP ログインポリシーオプション」を参照してください。
- **policy-option-value** – ログインポリシーオプションに割り当てられる値。UNLIMITED を指定すると、制限は使用されません。DEFAULT を指定すると、デフォルトの制限が使用されます。各オプションでサポートされている値については、「ログインポリシーオプション」と「LDAP ログインポリシーオプション」を参照してください。 .

適用対象

シングルプレックスとマルチプレックス。

例

(先頭に戻る) (32 ページ)

- **例 1** – 「論理サーバへのアクセス許可設定」と「マルチプレックスログインポリシーの設定」を参照してください。
- **例 2** – Test1 ログインポリシーで password_life_time 値を UNLIMITED に設定し、max_failed_login_attempts 値を 5 に設定します。

```
ALTER LOGIN POLICY Test1
password_life_time=UNLIMITED
max_failed_login_attempts=5;
```

使用法

(先頭に戻る) (32 ページ)

ポリシーオプションを指定しない場合は、ルートログインポリシーからこのログインポリシーの値が取得されます。新しいポリシーは、MAX_NON_DBA_CONNECTIONS および ROOT_AUTO_UNLOCK_TIME ポリシーオプションを継承しません。

新しいデータベースのすべてに、ルートログインポリシーが含まれています。ルートログインポリシーの値を変更することはできますが、ポリシーは削除できません。

パーミッション

(先頭に戻る) (32 ページ)

MANAGE ANY LOGIN POLICY システム権限が必要です。

ログインポリシーオプション

ルートログインポリシーとユーザ定義ログインポリシーで使用可能なオプションを次に示します。

オプション	説明
AUTO_UNLOCK_TIME	<p>MANAGE ANY USER システム権限が付与されていないアカウントがロックされてから自動的にロック解除されるまでの時間。このオプションは、ルートログインポリシーを含む任意のログインポリシーで定義できる。</p> <ul style="list-style-type: none"> • 値 - 0 ~ UNLIMITED • デフォルト - UNLIMITED • 適用対象 - MANAGE ANY USER システム権限が付与されていないすべてのユーザ。
CHANGE_PASSWORD_DUAL_CONTROL	<p>別のユーザのパスワードを変更するために、CHANGE PASSWORD システム権限が付与されている 2 人のユーザからの入力を要求する。</p> <ul style="list-style-type: none"> • 値 - ON、OFF • デフォルト - OFF • 適用対象 - すべてのユーザ。

オプション	説明
DEFAULT_LOGICAL_SERVER	<p>接続文字列で論理サーバが指定されていない場合、ユーザはユーザのログインポリシーで指定されている DEFAULT_LOGICAL_SERVER オプションに接続する。</p> <ul style="list-style-type: none"> • 値 – <ul style="list-style-type: none"> • 既存のユーザ定義論理サーバの名前。 • ALL – すべての論理サーバへのアクセスを許可する。 • AUTO – ルートログインポリシーのデフォルト論理サーバの値。 • COORDINATOR – 現在のコーディネータノード。 • NONE – あらゆるマルチプレックスサーバへのアクセスを拒否する。 • OPEN – 単独またはユーザ定義論理サーバの名前とともに使用する。どのユーザ定義論理サーバのメンバーでもないすべてのマルチプレックスノードへのアクセスを許可する。 • SERVER – SERVER 論理サーバのセマンティックに従って、すべてのマルチプレックスノードへのアクセスを許可する。 • デフォルト – AUTO • 適用対象 – すべてのユーザ。MANAGE MULTIPLEX システム権限が必要。
LOCKED	<p>ON に設定すると、ユーザは新しい接続を確立できない。この設定は一時的にログインポリシーユーザへのアクセスを拒否する。このオプションは、論理サーバに設定されたログインポリシーの上書きはできない。</p> <ul style="list-style-type: none"> • 値 – ON、OFF • デフォルト – OFF • 適用対象 – MANAGE ANY USER システム権限を持つユーザを除くすべてのユーザ。

オプション	説明
MAX_CONNECTIONS	<p>1 ユーザに許可される最大同時接続数。このオプションは、論理サーバごとの設定を指定できる。</p> <ul style="list-style-type: none"> • 値 - 0 ~ 2147483647 • デフォルト - UNLIMITED • 適用対象 - SERVER OPERATOR または DROP CONNECTION システム権限を持つユーザを除くすべてのユーザ。
MAX_DAYS_SINCE_LOGIN	<p>同一ユーザによる連続する 2 回のログインの間に許容される最大経過日数。</p> <ul style="list-style-type: none"> • 値 - 0 ~ 2147483647 • デフォルト - UNLIMITED • 適用対象 - MANAGE ANY USER システム権限を持つユーザを除くすべてのユーザ。
MAX_FAILED_LOGIN_ATTEMPTS	<p>前回のユーザアカウントへのログイン成功以降、アカウントがロックされるまでのログイン失敗の最大回数。</p> <ul style="list-style-type: none"> • 値 - 0 ~ 2147483647 • デフォルト - UNLIMITED • 適用対象 - すべてのユーザ。
MAX_NON_DBA_CONNECTIONS	<p>SERVER OPERATOR または DROP CONNECTION システム権限を持たないユーザが確立できる同時接続の最大数。このオプションは、ルートログインポリシーでのみサポートされる。</p> <ul style="list-style-type: none"> • 値 - 0 ~ 2147483647 • デフォルト - UNLIMITED • 適用対象 - SERVER OPERATOR または DROP CONNECTION システム権限を持つユーザを除くすべてのユーザ。

オプション	説明
PASSWORD_EXPIRY_ON_NEXT_LOGIN	<p>ON に設定すると、次のログイン時にユーザのパスワードの有効期限が切れる。</p> <ul style="list-style-type: none"> • 値 - ON、OFF • デフォルト - OFF • 適用対象 - すべてのユーザ。 <hr/> <p>注意： この機能は現在、SAP Control Center へのログイン時に実装されなくなっています。ユーザは、パスワードの変更を要求されません。ただし、SAP Control Center 外から(たとえば Interactive SQL を使用して) SAP Sybase IQ にログインする際には要求されます。</p>
PASSWORD_GRACE_TIME	<p>パスワードの有効期限が切れるまでの日数(ログインが可能だが、デフォルトの post_login プロシージャによって警告が発行される期間)。</p> <ul style="list-style-type: none"> • 値 - 0 ~ 2147483647 • デフォルト - 0 • 適用対象 - すべてのユーザ。
PASSWORD_LIFE_TIME	<p>パスワードの変更が必要となるまでの最大日数。</p> <ul style="list-style-type: none"> • 値 - 0 ~ 2147483647 • デフォルト - UNLIMITED • 適用対象 - すべてのユーザ。
ROOT_AUTO_UNLOCK_TIME	<p>MANAGE ANY USER システム権限が付与されているアカウントがロックされてから自動的にロック解除されるまでの時間。このオプションは、ルートログインポリシーでのみ定義できる。</p> <ul style="list-style-type: none"> • 値 - 0 ~ UNLIMITED • デフォルト - 15 • 適用対象 - MANAGE ANY USER システム権限が付与されているすべてのユーザ。

LDAP ログインポリシーオプション

LDAP ユーザ認証で使用可能なログインポリシーオプションを示します。

オプション	説明
LDAP_PRIMARY_SERVER	<p>プライマリ LDAP サーバの名前を指定する。</p> <ul style="list-style-type: none"> • 値 - 該当なし • デフォルト - なし • 適用対象 - すべてのユーザ。
LDAP_SECONDARY_SERVER	<p>セカンダリ LDAP サーバの名前を指定する。</p> <ul style="list-style-type: none"> • 値 - 該当なし • デフォルト - なし • 適用対象 - すべてのユーザ。
LDAP_AUTO_FAILBACK_PERIOD	<p>プライマリサーバへの自動フェールバックが試行されるまでの時間 (分単位) を指定する。</p> <ul style="list-style-type: none"> • 値 - 0 ~ 2147483647 • デフォルト - 15 分 • 適用対象 - すべてのユーザ。
LDAP_FAILOVER_TO_STD	<p>システムリソース、ネットワークの停止、接続のタイムアウト、または同様のシステム障害が原因で LDAP サーバによる認証に失敗した場合に、標準認証による認証を許可する。ただし、LDAP サーバから返された実際の認証の失敗を標準認証にフェールバックすることは許可しない。</p> <ul style="list-style-type: none"> • 値 - ON、OFF • デフォルト - ON • 適用対象 - すべてのユーザ。

オプション	説明
LDAP_REFRESH_DN	<p>ISYSLOGINPOLICYOPTION システムテーブル内の ldap_refresh_dn の値を協定世界時 (UTC) の現在時刻で更新する。</p> <p>ISYSLOGINPOLICYOPTION の ldap_refresh_dn の値が ISYSUSER の user_dn の値より新しい場合、LDAP によるユーザ認証のたびに新しいユーザ DN の検索が行われる。その場合、新しいユーザ DN で user_dn の値が更新され、現在時刻で user_dn_changed_at の値が再更新される。</p> <ul style="list-style-type: none"> • 値 - NOW • ROOT ポリシーの初期値 - NULL • ユーザ定義ログインポリシーの初期値 - 現在時刻を UTC で格納 • 適用対象 - すべてのユーザ。

マルチプレックスログインポリシーの設定

マルチプレックスサーバのログインポリシーを設定します。

例

この例では、論理サーバのログインポリシー設定が上書きされ、論理サーバ ls1 の最大接続数が増加します。

```
ALTER LOGIN POLICY lp1 max_connections=20 LOGICAL SERVER ls1;
```

使用法

マルチプレックスにのみ適用されます。

任意のマルチプレックスサーバ上で実行するログイン管理コマンドは、マルチプレックス内のすべてのサーバに自動的に伝達されます。最高のパフォーマンスを実現するには、これらのコマンドまたは DDL をコーディネータで実行します。

論理サーバレベルで上書きすると、特定のログインポリシーオプションが、論理サーバごとに設定が異なることとなります。

SYS.ISYSIQLSLOGINPOLICYOPTION には、論理サーバ上書きのためのログインポリシーオプション値が格納されています。ISYSIQLSLOGINPOLICYOPTION には、ログインポリシーオプションの論理サーバの上書きのそれぞれに対応するローが存在します。

論理サーバへのアクセス許可設定

論理サーバアクセスを設定します。

例 1

ルートログインポリシーが論理サーバの `ls4` と `ls5` へのアクセスを許可し、ログインポリシー `lp1` が論理サーバの割り当てなしで存在するとします。次の文は、ログインポリシー `lp1` に、論理サーバ `ls4` と `ls5` へのアクセス許可も実質的に割り当てます。

論理サーバ `ls1` をログインポリシー `lp1` に割り当てます。

```
ALTER LOGIN POLICY lp1 ADD LOGICAL SERVER ls1
```

例 2

次の文は、ログインポリシー `lp1` から論理サーバの `ls2` と `ls3` へのアクセスを許可します。

```
ALTER LOGIN POLICY lp1 ADD LOGICAL SERVER ls2, ls3
```

例 3

ログインポリシー `lp1` を変更して、`ls3` と `ls4` にのみにアクセスを許可します。

```
ALTER LOGIN POLICY lp1 ADD LOGICAL SERVER ls4
```

```
ALTER LOGIN POLICY lp1 DROP LOGICAL SERVER ls1, ls2
```

または

```
ALTER LOGIN POLICY lp1 SET LOGICAL SERVER ls3, ls4
```

例 4

ログインポリシー `lp1` を変更して、すべての論理サーバへのアクセスを拒否します。

```
ALTER LOGIN POLICY lp1 SET LOGICAL SERVER NONE
```

例 5

ログインポリシー `lp1` の現在の論理サーバ割り当てを削除し、ルートログインポリシーの論理サーバ割り当てを継承できるようにします。

```
ALTER LOGIN POLICY lp1 SET LOGICAL SERVER DEFAULT
```

使用法

`ADD` 句、`DROP` 句、または `SET` 句を使用すると、次のようにログインポリシーの論理サーバ割り当てを設定できます。

- **ADD** – 新しい論理サーバ割り当てをログインポリシーに追加します。
- **DROP** – ログインポリシーから既存の論理サーバ割り当てを削除します。
- **SET** – 特定のログインポリシーのすべての論理サーバ割り当てを新しい一連の論理サーバに置き換えます。

ADD 句、DROP 句、または SET 句のいずれか 1 つのみを使用します。SERVER、NONE、DEFAULT は、SET 句でのみ使用します。個別の論理サーバ名は、ls-assignment list または ls-override list ごとに 1 回のみ指定します。

次の場合には、エラーが返されます。

- ADD 句で指定された論理サーバが、すでにログインポリシーに割り当てられている場合。
- DROP 句で指定された論理サーバが、ログインポリシーに現在割り当てられていない場合。
- 論理サーバ割り当ての変更により、割り当てられている論理サーバ間でメンバーシップの重複が発生する場合。

SYS.ISSYSIQLOGINPOLICYLSINFO には、論理サーバ割り当ての情報が格納されています。ISSYSIQLOGINPOLICYLSINFO には、ログインポリシーオプションの論理サーバの上書きのそれぞれに対応するローが存在します。

ALTER LS POLICY 文

ルート論理サーバポリシーまたはユーザが作成した論理サーバポリシーのオプション値の一部またはすべてを変更します。次の文では、論理サーバで共有される物理ノード間で、一貫した共有システムテナンタリストア設定が強制的に適用されます。

クイックリンク：

「パラメータ」 (43 ページ)

「例」 (43 ページ)

「使用法」 (43 ページ)

「パーミッション」 (45 ページ)

構文

```
ALTER LS POLICY ls-policy-name ls-option-value-list
  [ WITH STOP SERVER ]

ls-option-value-list - (back to Syntax)
  { ls-option-name = ls-policy-option-value } ...
```

ls-option-name - (back to ls-option-value-list)

```

ALLOW_COORDINATOR_AS_MEMBER
| DQP_ENABLED
| ENABLE_AUTOMATIC_FAILOVER
| LOGIN_REDIRECTION
| REDIRECTION_WAITERS_THRESHOLD
| TEMP_DATA_IN_SHARED_TEMP

```

パラメータ

(先頭に戻る) (42 ページ)

- **ls-policy-name** – 論理サーバポリシーの名前。ルートを指定してルート論理サーバポリシーを変更します。
- **ls-option-value-list** – 論理サーバポリシーオプションの名前。オプションのリストについては、「使用法 (43 ページ)」を参照してください。
- **ls-policy-option-value** – 指定されていないオプションの値はすべて、ルート論理サーバポリシーから継承されます。「使用法 (43 ページ)」を参照してください。
- **WITH STOP SERVER** – TEMP_DATA_IN_SHARED_TEMP オプションが直接的または間接的に変更された場合、論理サーバ内のすべてのサーバを自動的に停止します。

適用対象

マルチプレックスのみ。

例

(先頭に戻る) (42 ページ)

- **例 1** – 論理サーバポリシーを変更します。

```

ALTER LS POLICY root
ALLOW_COORDINATOR_AS_MEMBER=ON

```

- **例 2** – 論理サーバポリシーを変更し、オプション値が変更された場合にサーバが自動的に停止するようにします。

```

ALTER LS POLICY root
TEMP_DATA_IN_SHARED_TEMP=ON WITH STOP SERVER

```

使用法

(先頭に戻る) (42 ページ)

より小さな IQ_SYSTEM_TEMP DB 領域が必要な場合は、TEMP_DATA_IN_SHARED_TEMP を ON に設定します。こうすることによって、

テンポラリデータが IQ_SYSTEM_TEMP ではなく、IQ_SHARED_TEMP に書き込まれます。ただし、分散クエリ処理環境では、DQP_ENABLED と TEMP_DATA_IN_SHARED_TEMP の両方を ON に設定すると、IQ_SHARED_TEMP 内の追加データで SAN が飽和する可能性があります。これは、IQ_SHARED_TEMP に対する追加の I/O 処理が DQP のパフォーマンスに悪影響を及ぼす可能性があるためです。

オプション	説明
ALLOW_COORDINATOR_AS_MEMBER	<p>ROOT 論理サーバポリシーに対してのみ設定可能。ON (デフォルト) の場合は、コーディネータを任意のユーザ定義論理サーバのメンバーにできる。OFF の場合は、コーディネータを任意のユーザ定義論理サーバのメンバーとして使用できない。</p> <ul style="list-style-type: none"> • 値 - ON、OFF • デフォルト - ON
DQP_ENABLED	<p>0 に設定すると、クエリ処理が分散されない。1 (デフォルト) に設定した場合、書き込み可能な共有テンポラリファイルが存在するかぎり、クエリ処理が分散される。2 に設定した場合、クエリ処理がネットワーク上で分散され、共有テンポラリストアは使用されない。</p> <ul style="list-style-type: none"> • 値 - 0、1、2 • デフォルト - 1
ENABLE_AUTOMATIC_FAILOVER	<p>ROOT 論理サーバポリシーに対してのみ設定可能。ON は、指定のログインポリシーが適用される論理サーバの自動フェールオーバーを有効化。OFF (デフォルト) は、論理サーバレベルの自動フェールオーバーを無効にし、手動フェールオーバーを可能にする。デフォルト値の設定に戻すには DEFAULT を指定する。</p> <ul style="list-style-type: none"> • 値 - ON、OFF、DEFAULT • デフォルト - OFF
LOGIN_REDIRECTION	<p>ON は、指定のログインポリシーが適用される論理サーバのログインリダイレクションを有効化。OFF (デフォルト) は、論理サーバレベルのログインリダイレクションを無効にし、外部接続管理を可能にする。</p> <ul style="list-style-type: none"> • 値 - ON、OFF • デフォルト - OFF

オプション	説明
REDIRECTION_WAITERS_THRESHOLD	<p>SAP Sybase IQ がこの論理サーバへの接続を別のサーバにリダイレクトする前に、キューに入れることが可能な接続数を指定する。任意の整数値に設定可能。デフォルトは 5。</p> <ul style="list-style-type: none"> • 値 - 整数 • デフォルト - 5
TEMP_DATA_IN_SHARED_TEMP	<p>ON の場合、共有テンポラリストアに 1 つ以上の読み書きファイルが追加されていると、すべてのテンポラリテーブルデータと対象スクラッチデータが共有テンポラリストアに書き込まれる。このオプションを設定した後、または、読み書きファイルを共有テンポラリストアに追加した後は、すべてのマルチプレックスノードの再起動が必要 (共有テンポラリストアに読み書きファイルがない場合、または、ノードを再起動しなかった場合は、データが IQ_SYSTEM_TEMP に書き込まれる)。OFF (デフォルト) の場合は、すべてのテンポラリテーブルデータとスクラッチデータがローカルテンポラリストアに書き込まれる。</p> <ul style="list-style-type: none"> • 値 - ON、OFF • デフォルト - OFF

パーミッション

(先頭に戻る) (42 ページ)

MANAGE MULTIPLEX システム権限が必要です。

ALTER MULTIPLEX RENAME 文

マルチプレックスの名前を変更し、SYS.ISYSIQINFO システムテーブルにマルチプレックスの名前を格納します。

クイックリンク：

「使用法」 (46 ページ)

「パーミッション」 (46 ページ)

構文

```
ALTER MULTIPLEX RENAME multiplex-name
```

適用対象

マルチプレックスのみ。

使用法

(先頭に戻る) (45 ページ)

作成されたマルチプレックスには、コーディネータの名前が付けられます。この文は自動的にコミットされます。

パーミッション

(先頭に戻る) (45 ページ)

MANAGE MULTIPLEX システム権限が必要です。

ALTER MULTIPLEX SERVER 文

特定のサーバの名前、カタログファイルパス、役割、またはステータスを変更します。

クイックリンク：

「パラメータ」 (46 ページ)

「例」 (47 ページ)

「パーミッション」 (47 ページ)

構文

構文 1

```
ALTER MULTIPLEX SERVER server-name server-option
```

server-option - (back to Syntax 1)

```
{ RENAME new-server-name  
| DATABASE 'dbfile'  
| ROLE { WRITER | READER | COORDINATOR }  
| STATUS { INCLUDED | EXCLUDED }  
| ASSIGN AS FAILOVER SERVER  
| host-port-list }
```

host-port-list (back to server-option)

```
{ HOST ' hostname ' PORT port number ...}  
{ PRIVATE HOST ' hostname ' PORT port number ...}
```

構文 2

```
ALTER MULTIPLEX SERVER server-name PRIVATE NULL
```

パラメータ

(先頭に戻る) (46 ページ)

- **RENAME** – 特定のサーバの名前を変更します。サーバは自動的に停止し、次の再起動時に新しい名前が必要となります。
- **DATABASE** – 特定のサーバのカタログファイルパスを変更します。サーバは自動的に停止し、次の再起動時に新しいカタログパスが必要となります。ユーザはカタログファイルを移動する必要があります。
- **ROLE** – 特定のサーバのロールを変更します。ユーザは、コーディネータのロールの変更や、コーディネータへのロールの変更はできません。ライターノードのロールがリーダーに変更された場合、サーバは停止します。
- **STATUS** – 特定のサーバのステータスを変更します。フェールオーバーノードは、そのノードが除外される最後のノードである場合を除き、除外することはできません。除外後に、サーバは自動的に停止します。ノードの追加後は、そのノードを同期し、再起動します。
- **ASSIGN** – 特定のサーバを新しいフェールオーバーサーバとして指定します。除外済みのステータスにあるノードは使用できません。ASSIGN AS FAILOVER 句は、他の **ALTER MULTIPLEX SERVER** 句とともに使用できないスタンドアロン句です。

コーディネータが実行中である必要がありますが、**ALTER MULTIPLEX SERVER** コマンドはマルチプレックス内のいずれのサーバからでも実行できます (すべての DDL 文をコーディネータで実行します)。名前付きサーバは、ロールをリーダーからライターに変更する場合を除き、すべての場合に自動的に停止します。

- **host-port-list** – サーバを除外する前に、そのサーバを停止します。除外されるサーバは、停止されていない場合、除外された後に自動的に停止します。また、除外されたサーバをマルチプレックスに再び参加させるには、**ALTER MULTIPLEX SERVER server-name STATUS INCLUDED** を実行し、同期を行う必要があります。

適用対象

マルチプレックスのみ。

例

(先頭に戻る) (46 ページ)

- **例 1** – セカンダリサーバ `mpx_writer1` を除外します。

```
ALTER MULTIPLEX SERVER mpx_writer1 STATUS EXCLUDED
```

パーミッション

(先頭に戻る) (46 ページ)

MANAGE MULTIPLEX システム権限が必要です。

ALTER PROCEDURE 文

既存のプロシージャを修正したものに置き換えます。このとき、修正したプロシージャ全体を **ALTER PROCEDURE** 文にインクルードして、プロシージャに対するユーザパーミッションを再割り当てします。

クイックリンク：

「パラメータ」 (49 ページ)

「使用法」 (49 ページ)

「標準」 (50 ページ)

「パーミッション」 (50 ページ)

構文

構文 1

```
ALTER PROCEDURE [ owner.]procedure-name procedure-definition
```

構文 2

```
ALTER PROCEDURE [ owner.]procedure-name
  REPLICATE { ON | OFF }
```

構文 3

```
ALTER PROCEDURE [ owner.]procedure-name
  SET HIDDEN
```

構文 4

```
ALTER PROCEDURE [ owner.]procedure-name
  RECOMPILE
```

構文 5

```
ALTER PROCEDURE
  [ owner.]procedure-name ( [ parameter, ...] )
  [ RESULT (result-column, ...)]
  EXTERNAL NAME 'external-call' [ LANGUAGE JAVA [ environment-name ] ]
```

external-call - (back to Syntax 5)
 [column-name:]function-name@library; ...

environment-name - (back to Syntax 5)
 DISALLOW | ALLOW SERVER SIDE REQUESTS

パラメータ

(先頭に戻る) (48 ページ)

- **procedure-definition** – 名前の後に **CREATE PROCEDURE** 構文。
- **REPLICATE** – SAP Sybase Replication Server を使用してプロシージャを他のサイトに再配置する必要がある場合は、**REPLICATE ON** 句を使用します。
- **SET HIDDEN** – 関連するプロシージャの定義を難読化し、解読できないようにします。このプロシージャはアンロードして、他のデータベースに再ロードできます。

注意： この設定は、元に戻せません。元のプロシージャ定義をデータベースの外部に保持することをおすすめします。

- **RECOMPILE** – ストアドプロシージャを再コンパイルします。ストアドプロシージャを再コンパイルすると、カタログに格納された定義が再解析され、構文が検証されます。
プロシージャの定義は、再コンパイルしても変わりません。**SET HIDDEN** 句で隠された定義を持つプロシージャは再コンパイルできますが、これらの定義は隠されたままになります。
- **RESULT** – 結果セットを生成するものの **RESULT** 句を含まないプロシージャの場合は、データベースサーバがプロシージャの結果セットの特性を判別しようとし、カタログに情報を格納します。これは、プロシージャの作成後に、プロシージャの参照先テーブルがカラムの追加、削除、または名前変更を行うように変更されている場合に役立つことがあります。
- **environment-name** – デフォルトは **DISALLOW** です。**ALLOW** はサーバ側接続が許可されることを示します。

注意：

- **ALLOW** は、必要な場合以外は指定しないでください。**ALLOW** 句を使用すると、特定の種類の SAP Sybase IQ テーブルジョインの速度が低下します。
 - **UDF** を使用するとき **ALLOW SERVER SIDE REQUESTS** 句と **DISALLOW SERVER SIDE REQUESTS** 句の両方を同じクエリの中で指定しないでください。
-

使用法

(先頭に戻る) (48 ページ)

ALTER PROCEDURE 文には、新しいプロシージャ全体を含める必要があります。**PROC** を **PROCEDURE** の同意語として使用することができます。Watcom-SQL ダイアレクトと Transact-SQL® ダイアレクトのプロシージャは、いずれも **ALTER**

PROCEDURE を使用して変更できます。プロシージャの既存のパーミッションは変更されません。**DROP PROCEDURE** に続けて **CREATE PROCEDURE** を実行した場合は、実行パーミッションが再割り当てされます。

構文 1 と構文 2 を組み合わせることはできません。

テーブル UDF に **ALTER PROCEDURE** 文を使用する場合、**CREATE PROCEDURE** 文 (外部プロシージャ) と同じ制限が適用されます。

標準

(先頭に戻る) (48 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 — SAP Adaptive Server® Enterprise ではサポートされていません。

パーミッション

(先頭に戻る) (48 ページ)

Watcom-SQL または Transcat-SQL プロシージャの変更 - 次のいずれかが必要です。

- ALTER ANY PROCEDURE システム権限
- ALTER ANY OBJECT システム権限
- プロシージャを所有している

外部 C/C++ または外部環境プロシージャの変更 - CREATE EXTERNAL REFERENCE システム権限が必要です。次のいずれかも必要です。

- ALTER ANY PROCEDURE システム権限
- ALTER ANY OBJECT システム権限
- プロシージャを所有している

参照：

- CREATE PROCEDURE 文 (210 ページ)

ALTER ROLE 文

互換ロールをユーザ定義システムロールに移行してから、その互換ロールを自動的に削除します。

注意： ALTER ROLE 文を使用して SYS_AUTH_SA_ROLE または SYS_AUTH_SSO_ROLE を移行することはできません。これらのロールは、SYS_AUTH_DBA_ROLE が移行されると、自動的に移行されます。

クイックリンク：

「パラメータ」 (51 ページ)

「例」 (51 ページ)

「使用法」 (52 ページ)

「標準」 (52 ページ)

「パーミッション」 (52 ページ)

構文

構文 1 – SYS_AUTH_DBA_ROLE を移行する場合

```
ALTER ROLE predefined_sys_role_name  
MIGRATE TO new_role_name [, new_sa_role_name, new_sso_role_name]
```

構文 2 – 他のすべての互換ロールを移行する場合

```
ALTER ROLE predefined_sys_role_name  
MIGRATE TO new_role_name
```

パラメータ

(先頭に戻る) (50 ページ)

- **predefined_sys_role_name** – データベースにまだ存在する (まだ削除されていない) 互換ロールの名前。
- **new_role_name** – 新しいロールの名前として、プレフィクス SYS_ で始まる名前またはサフィクス _ROLE で終わる名前は使用できません。
- **new_sa_role_name** – SYS_AUTH_DBA_ROLE を移行する場合のみ指定する必要があります。SYS_AUTH_SA_ROLE の基礎となるシステム権限を移行する新しいロールはデータベースに既存であってははいけません。また、新しいロールの名前として、プレフィクス SYS_ で始まる名前またはサフィクス _ROLE で終わる名前は使用できません。
- **new_sso_role_name** – SYS_AUTH_DBA_ROLE を移行する場合のみ指定する必要があります。SYS_AUTH_SSO_ROLE の基礎となるシステム権限を移行する新しいロールはデータベースに既存であってははいけません。また、新しいロールの名前として、プレフィクス SYS_ で始まる名前またはサフィクス _ROLE で終わる名前は使用できません。

例

(先頭に戻る) (50 ページ)

- **例 1** – SYS_AUTH_DBA_ROLE を新しいロール Custom_DBA、Custom_SA、および Custom_SSO にそれぞれ移行します。さらに自動的に、SYS_AUTH_DBA_ROLE に付与されているすべてのユーザ、基礎となるシステム権限、およびロールを適用可能な新しいロールに移行します。最後に、SYS_AUTH_DBA_ROLE、SYS_AUTH_SA_ROLE、および SYS_AUTH_SSO_ROLE を削除します。

```
ALTER ROLE SYS_AUTH_DBA_ROLE  
MIGRATE TO Custom_DBA, Custom_SA, Custom_SSO
```

- **例 2** – SYS_AUTH_OPERATOR_ROLE ロールを新しいロール Operator_role に移行します。さらに自動的に、SYS_AUTH_OPERATOR_ROLE に付与されているすべてのユーザ、基礎となるシステム権限、およびロールを新しいロールに移行し、SYS_AUTH_OPERATOR_ROLE を削除します。

```
ALTER ROLE SYS_AUTH_OPERATOR_ROLE  
MIGRATE TO Operator_role
```

使用法

(先頭に戻る) (50 ページ)

移行プロセス中は、次の処理が実行されます。

- 新しいユーザ定義ロールが作成されます。
- 移行中の事前定義済みのロールに現在付与されているすべてのシステム権限が、自動的に新しいユーザ定義ロールに付与されます。
- 移行中の事前定義済みのロールに現在付与されているすべてのユーザとロールが、自動的に新しいユーザ定義ロールに付与されます。
- 互換ロールは削除されます。

移行プロセス中はロール管理者が指定されないので、グローバルロール管理者のみが新しいロールを管理できます。CREATE ROLE 文を使用して、適切な管理権限を持つロール管理者をロールに追加します。

標準

(先頭に戻る) (50 ページ)

ANSI SQL – 準拠レベル： Transact-SQL 拡張。

パーミッション

(先頭に戻る) (50 ページ)

MANAGE ROLES システム権限が管理権限付きで付与されている必要があります。

ALTER SEQUENCE 文

シーケンスを変更します。この文は、SAP Sybase IQ のカタログストアテーブルのみに適用されます。

構文

```
ALTER SEQUENCE [ owner. ] sequence-name  
[ RESTART WITH signed-integer ]  
[ INCREMENT BY signed-integer ]  
[ MINVALUE signed-integer | NO MINVALUE ]  
[ MAXVALUE signed-integer | NO MAXVALUE ]  
[ CACHE integer | NO CACHE ]  
[ CYCLE | NO CYCLE ]
```

パラメータ

RESTART WITH 句 – 指定したシーケンスを指定した値で再起動します。

INCREMENT BY 句 – 最後に割り当てられた値から次のシーケンス値までの増分を定義します。デフォルトは 1 です。負の値を指定すると、降順のシーケンスが生成されます。INCREMENT BY 値が 0 の場合は、エラーが返されます。

MINVALUE 句 – シーケンスで生成される最小値を定義します。デフォルトは 1 です。MINVALUE が $(2^{63}-1)$ より大きいか $-(2^{63}-1)$ より小さい場合は、エラーが返されます。また、MINVALUE が MAXVALUE より大きい場合も、エラーが返されます。

MAXVALUE 句 – シーケンスで生成される最大値を定義します。デフォルトは -1 です。MAXVALUE が $2^{63}-1$ より大きいか $-(2^{63}-1)$ より小さい場合は、エラーが返されます。

CACHE 句 – より速くアクセスできるようにメモリに保持される、事前に割り付けられたシーケンス値の数を指定します。キャッシュが不足すると、シーケンスキャッシュが再移植され、対応するエントリがトランザクションログに書き込まれます。キャッシュの現在の値は、チェックポイントの時点で ISYSSEQUENCE システムテーブルに転送されます。デフォルトは 100 です。

CYCLE 句 – 最大値または最小値に達した後に、値の生成を継続するかどうかを指定します。

備考

指定したシーケンスが見つからない場合は、エラーメッセージが返されます。

権限

シーケンスの所有者であるか、または次のいずれかの権限を持っていることが必要です。

ALTER ANY SEQUENCE システム権限

ALTER ANY OBJECT システム権限

関連する動作

なし

標準と互換性

- **SQL/2008** – ALTER SEQUENCE 文は、SQL/2008 標準のオプションの SQL 言語機能 T176 の一部です。CACHE 句はベンダー拡張です。

例

次の例は、シーケンス Test に新しい最大値を設定します。

```
ALTER SEQUENCE Test
MAXVALUE 1500;
```

ALTER SERVER 文

リモートサーバの属性を変更します。**ALTER SERVER** による変更は、次にリモートサーバに接続するまで有効になりません。

クイックリンク：

「パラメータ」 (55 ページ)

「例」 (56 ページ)

「使用法」 (56 ページ)

「標準」 (57 ページ)

「パーミッション」 (57 ページ)

構文

```
ALTER SERVER server-name
[ CLASS 'server-class' ]
[ USING 'connection-info' ]
[ CAPABILITY 'cap-name' { ON | OFF } ]
[ CONNECTION CLOSE [ CURRENT | ALL | connection-id ] ]
```

server-class - (構文に戻る)
{ ASAJDBC

```
| ASEJDBC
| SAODBC
| ASEODBC
| DB2ODBC
| MSSODBC
| ORAODBC
| ODBC }
```

connection-info - (構文に戻る)

```
{ machine-name:port-number [ /dbname ] | data-source-name }
```

パラメータ

(先頭に戻る) (54 ページ)

- **cap-name** – サーバ機能の名前
- **CLASS** – サーバクラスを変更します。
- **USING** – JDBC ベースのサーバクラスを使用する場合、USING 句は *hostname:port-number [/dbname]* となります。
 - **hostname** – リモートサーバを実行するマシン。
 - **portnumber** – リモートサーバが受信する TCP/IP ポート番号。SAP Sybase IQ および SAP Sybase SQL Anywhere® のデフォルトのポート番号は 2638 です。
 - **dbname** – SQL Anywhere リモートサーバでは、*dbname* を指定しない場合、デフォルトのデータベースが使用されます。Adaptive Server の場合は、デフォルトが master データベースです。*dbname* を使用しない場合は、他の方法 (**FORWARD TO** 文など) を使って別のデータベースを指定します。

ODBC ベースのサーバクラスを使用する場合、USING 句は *data-source-name* になります。これは ODBC データソース名です。

- **CAPABILITY** – サーバの機能の ON と OFF を切り替えます。サーバの機能は、システムテーブル SYSCAPABILITY に格納されています。サーバ機能の名前は、システムテーブル SYSCAPABILITYNAME に格納されています。サーバへの最初の接続が確立されるまで、SYSCAPABILITY テーブルには、リモートサーバのエントリは含まれません。最初の接続時に、SAP Sybase IQ はサーバに対して機能に関する問い合わせを行って、SYSCAPABILITY に結果を格納します。後続の接続では、このテーブルからサーバの機能が取得されます。
通常、サーバの機能を変更する必要はありません。クラス ODBC の汎用サーバの機能を変更しなければならない場合があります。
- **CONNECTION CLOSE** – ユーザがリモートサーバへの接続を確立した場合、ユーザがローカルデータベースから接続を切断しないかぎり、リモート接続は閉じられません。CONNECTION CLOSE 句を使用すると、リモートサーバへの

SQL 文

接続を明示的に閉じることができます。この句は、リモート接続が非アクティブまたは不要になった場合に役立ちます。

次の SQL 文は互いに同じであり、リモートサーバへの現在の接続を閉じるために使用されます。

```
ALTER SERVER server-name CONNECTION CLOSE
ALTER SERVER server-name CONNECTION CLOSE CURRENT
```

この構文を使用すると、リモートサーバへの ODBC 接続と JDBC 接続の両方を閉じることができます。これらの文の実行に SERVER OPERATOR システム権限は不要です。

また、接続 ID を指定して特定のリモート ODBC 接続を切断することも、ALL キーワードを指定してすべてのリモート ODBC 接続を切断することもできます。接続 ID または ALL キーワードを指定して JDBC 接続を閉じようとする、エラーが発生します。*connection-id* で指定された接続が現在のローカル接続ではない場合、接続を閉じるにはユーザに SERVER OPERATOR システム権限が必要です。

例

(先頭に戻る) (54 ページ)

- **例 1** – 名前が `ase_prod` である Adaptive Server サーバのサーバクラスを変更して、SAP Sybase IQ への接続が ODBC ベースになるようにします。データソース名は `ase_prod` です。

```
ALTER SERVER ase_prod
CLASS 'ASEODBC'
USING 'ase_prod'
```

- **例 2** – サーバ `infodc` の機能を変更します。

```
ALTER SERVER infodc
CAPABILITY 'insert select' OFF
```

- **例 3** – リモートサーバ `rem_test` への接続をすべて閉じます。

```
ALTER SERVER rem_test
CONNECTION CLOSE ALL
```

- **例 4** – 接続 ID 142536 を使用するリモートサーバ `rem_test` への接続を閉じます。

```
ALTER SERVER rem_test
CONNECTION CLOSE 142536
```

使用法

(先頭に戻る) (54 ページ)

関連する動作：

- オートコミット

標準

(先頭に戻る) (54 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Open Client/Open Server でサポートされています。

パーミッション

(先頭に戻る) (54 ページ)

SERVER OPERATOR システム権限が必要です。

参照：

- CREATE SERVER 文 (241 ページ)
- DROP SERVER 文 (337 ページ)

ALTER SERVICE 文

データベースサーバを Web サーバとして機能させます。

クイックリンク：

「パラメータ」 (58 ページ)

「例」 (60 ページ)

「使用法」 (60 ページ)

「標準」 (60 ページ)

「パーミッション」 (60 ページ)

構文

```
ALTER SERVICE service-name
  [ TYPE service-type-string ]
  [ attributes ]
  [ AS statement' ]
```

service-type-string - (構文に戻る)

```
{ 'RAW '
| 'HTML '
| 'XML '
| 'SOAP '
```

```
| 'DISH ' }
```

attributes - (構文に戻る)

```
[ AUTHORIZATION { ON | OFF } ]
[ SECURE { ON | OFF } ]
[ USER { user-name | NULL } ]
[ URL [ PATH/ ] { ON | OFF | ELEMENTS } ]
[ USING { SOAP-prefix | NULL } ]
```

パラメータ

(先頭に戻る) (57 ページ)

- **service-type-string** – Web サービス名には、英数字または "/"、"-","_",".", "!", "~", "*", "", "(", ")" を自由に並べて指定できます。ただし、最初の文字をスラッシュ (/) にしないでください。また、名前にスラッシュを2つ以上続けて使用することはできません。
- **AUTHORIZATION** – ユーザがサービスに接続する際、ユーザ名とパスワードの指定が必要かどうかを定義します。デフォルト値は ON です。
 - 認証が OFF の場合、AS 句が必要となり、USER 句によって1人のユーザが識別されます。要求はすべて、そのユーザのアカウントとパーミッションを使用して実行されます。
 - ON の場合は、すべてのユーザにユーザ名とパスワードの入力が必要になります。必要に応じて、USER 句を使用しユーザ名またはロール名を指定することにより、ユーザに対しサービスの使用許可を制限することもできます。ユーザ名が NULL の場合、既知のユーザはすべてサービスにアクセスできます。

運用システムは、認証を ON にして実行します。ユーザをロールに追加することにより、サービスを使用するためのパーミッションを付与します。

- **SECURE** – 安全でない接続を受け入れるかどうかを指定します。ON は、HTTPS 接続のみ受け入れることを意味します。HTTP ポートで受信したサービス要求は、HTTPS ポートへ自動的にリダイレクトされます。OFF に設定すると、HTTP と HTTPS の接続をどちらも受け入れます。デフォルト値は OFF です。
- **USER** – AUTHORIZATION が OFF の場合、このパラメータは必須となり、すべてのサービス要求の実行に使用するユーザ ID が指定されます。AUTHORIZATION が ON (デフォルト) であれば、この句はオプションとなり、サービスへのアクセスが許可されたユーザまたはロールが識別されます。デフォルト値は NULL で、すべてのユーザにアクセス許可が与えられます。
- **URL** – URI パスを受け入れるかどうか、また、受け入れる場合はどのように処理されるかを定義します。OFF の場合、URI 要求のサービス名の後に何も指定

することができません。ON の場合は、URI の残りの部分が、*url* という名前の変数の値として解釈されます。ELEMENTS は、URI パスの残りの部分がスラッシュ文字で区切られて、最大 10 個の要素のリストで構成されることを指定します。値は、*url* の末尾に 1 から 10 までの数字を付けた名前を持つ変数に割り当てられます。たとえば、3 つ目までの変数の名前は *url1*、*url2*、*url3* です。指定された値が 10 に満たなければ、残りの変数は NULL に設定されます。サービス名が / の文字で終わっている場合、URL は OFF に設定する必要があります。デフォルト値は OFF です。

- **USING** – DISH サービスにだけ適用されます。パラメータには名前のプレフィックスを指定します。このプレフィックスで始まる名前の SOAP サービスだけが処理されます。
- **service-type-string** – サービスのタイプを識別します。タイプはリストされたサービスタイプのいずれかにする必要があります。デフォルト値はありません。
 - **RAW** – 結果セットをそのままクライアントに送信します。プロシージャ内で必要なタグを明示的に生成することで、フォーマットされたドキュメントを作成できます。
 - **HTML** – 文またはプロシージャの結果セットをテーブルを格納する HTML ドキュメントにフォーマットします。
 - **XML** – 結果セットを XML フォーマットとして扱います。XML フォーマットではない場合は、自動的に XML RAW フォーマットに変換されます。
 - **SOAP** – 結果セットを Simple Object Access Protocol (SOAP) 応答としてフォーマットします。有効な SOAP 要求を使用する必要があります。SOAP 標準の詳細については、www.w3.org/TR/SOAP を参照してください。
 - **DISH** – SOAP ハンドラ、つまり DISH を指定します。このサービスは、1 つ以上の SOAP サービスのプロキシとして機能します。使用中は、複数の SOAP サービスへのアクセスを保持し、アクセス可能にするコンテナとして機能します。DISH に含まれる SOAP サービスのそれぞれに、Web Services Description Language (WSDL) ファイルが自動的に生成されます。内部の SOAP サービスは、共通のプレフィックスで識別されます。この識別子は、**USING** 句で指定する必要があります。
- **statement** – **statement** が NULL の場合、実行する文を URI に指定する必要があります。それ以外の場合は、指定された SQL 文だけがそのサービス内で実行可能になります。**statement** は SOAP サービスには必須ですが、DISH サービスでは無視されます。デフォルト値は NULL です。

SQL 文

運用システム内で実行されるすべてのサービスで `statement` を定義する必要があります。AUTHORIZATION が ON である場合のみ、`statement` を NULL にできません。

例

(先頭に戻る) (57 ページ)

- **例 1** – Web サーバをすばやく設定するには、**-xs** スイッチを指定してデータベースサーバを起動し、次の文を実行します。

```
CREATE SERVICE tables TYPE 'HTML'  
ALTER SERVICE tables  
AUTHORIZATION OFF  
USER DBA  
AS SELECT * FROM SYS.ISYSTABAfter executing these statements, use  
any Web browser to open the URL http://localhost/tables.
```

使用法

(先頭に戻る) (57 ページ)

Web サービスの名前を変更することはできません。

標準

(先頭に戻る) (57 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Adaptive Server ではサポートされていません。

パーミッション

(先頭に戻る) (57 ページ)

MANAGE ANY WEB SERVICE システム権限が必要です。

参照：

- CREATE SERVICE 文 (244 ページ)
- DROP SERVICE 文 (338 ページ)

ALTER SPATIAL REFERENCE SYSTEM 文

既存の空間参照系の設定を変更します。

クイックリンク：

「パラメータ」 (61 ページ)

「例」 (66 ページ)

「使用法」 (66 ページ)

「標準」 (67 ページ)

「パーミッション」 (67 ページ)

構文

ALTER SPATIAL REFERENCE SYSTEM

```
srs-name
[ srs-attribute [ srs-attribute ... ] ]
```

srs-attribute - (構文に戻る)

```
SRID srs-id
| DEFINITION { definition-string | NULL }
| ORGANIZATION { organization-name IDENTIFIED BY organization-srs-id
| NULL }
| TRANSFORM DEFINITION { transform-definition-string | NULL }
| LINEAR UNIT OF MEASURE linear-unit-name
| ANGULAR UNIT OF MEASURE { angular-unit-name | NULL }
| TYPE { ROUND EARTH | PLANAR }
| COORDINATE coordinate-name { UNBOUNDED | BETWEEN low-number
AND high-number }
| ELLIPSOID SEMI MAJOR AXIS semi-major-axis-length { SEMI MINOR AXIS
semi-minor-axis-length
| INVERSE FLATTENING inverse-flattening-ratio }
| TOLERANCE { tolerance-distance | DEFAULT }
| SNAP TO GRID { grid-size | DEFAULT }
| AXIS ORDER axis-order
| POLYGON FORMAT polygon-format
| STORAGE FORMAT storage-format
```

grid-size - (back to srs-attribute)

DOUBLE : usually between 0 and 1

axis-order - (back to srs-attribute)

{ '*x/y/z/m*' | '*long/lat/z/m*' | '*lat/long/z/m*' }

polygon-format - (back to srs-attribute)

{ '**CounterClockWise**' | '**Clockwise**' | '**EvenOdd**' }

storage-format - (back to srs-attribute)

{ '**Internal**' | '**Original**' | '**Mixed**' }

パラメータ

(先頭に戻る) (60 ページ)

- **IDENTIFIED BY** - 空間参照系の SRID 番号。

- **DEFINITION** – 座標系のデフォルトを設定または上書きします。DEFINITION 句以外の句で属性が設定されている場合は、DEFINITION 句の設定内容に関係なく、他の句で指定された値が優先されます。

definition-string は、SQL/MM と OGC で定義されるような、空間参照系の Well Known Text 構文の文字列です。たとえば、次のクエリは WGS 84 の定義を返します。

```
SELECT ST_SpatialRefSys::ST_FormatWKT( definition )
FROM ST_SPATIAL_REFERENCE_SYSTEMS
WHERE srs_id=4326;
```

Interactive SQL で戻り値をダブルクリックすると、読みやすい形で値が表示されます。

DEFINITION 句が指定されている場合は、*definition-string* が解析され、属性のデフォルト値の選択に使用されます。たとえば、*definition-string* に、*organization-name* と *organization-srs-id* を定義する AUTHORITY 要素が含まれる場合があります。

definition-string のパラメータ値は、SQL 文の句で明示的に設定した値で上書きされます。たとえば、ORGANIZATION 句が指定されていると、*definition-string* の ORGANIZATION の値が上書きされます。

- **ORGANIZATION** – 新しい空間参照系の基となる空間参照系を作成した組織に関する情報。
- **TRANSFORM DEFINITION** – 空間参照系に使用する変換の説明。現時点では、PROJ.4 変換のみがサポートされています。変換定義は、空間参照系間でデータを変換するときに ST_Transform メソッドで使用されます。一部の变換は、*transform-definition-string* が定義されていなくても使用できます。
- **LINEAR UNIT OF MEASURE** – 空間参照系の線形測定単位。指定する値は、ST_UNITS_OF_MEASURE システムビューで定義された線形測定単位と一致している必要があります。

この句が指定されておらず、DEFINITION 句で定義されていない場合、デフォルトは METRE です。事前に定義された測定単位をデータベースに追加するには、sa_install_feature システムプロシージャを使用します。

カスタム測定単位をデータベースに追加するには、CREATE SPATIAL UNIT OF MEASURE 文を使用します。

注意： METRE と METER はいずれも有効なスペルですが、SQL/MM 標準に準拠している METRE を使用することをおすすめします。

- **ANGULAR UNIT OF MEASURE** – 空間参照系の角度測定単位。指定する値は、`ST_UNITS_OF_MEASURE` システムテーブルで定義された角度測定単位と一致している必要があります。

この句が指定されておらず、`DEFINITION` 句で定義されていない場合、デフォルトは、地理的空間参照系の場合は `DEGREE`、非地理的空間参照系の場合は `NULL` です。

角度測定単位は、地理的空間参照系の場合は `NULL` 以外、非地理的空間参照系の場合は `NULL` にしてください。

角度測定単位は、地理的空間参照系の場合は `NULL` 以外、非地理的空間参照系の場合は `NULL` にしてください。事前に定義された測定単位をデータベースに追加するには、`sa_install_feature` システムプロシージャを使用します。

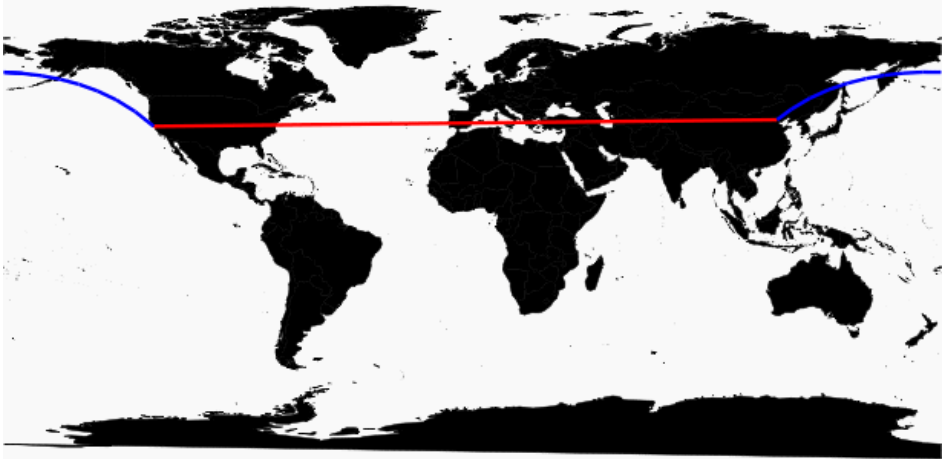
カスタム測定単位をデータベースに追加するには、`CREATE SPATIAL UNIT OF MEASURE` 文を使用します。

- **TYPE** – `SRS` で点と点を結ぶ線を解釈する方法を制御します。地理的空間参照系では、`TYPE` 句で `ROUND EARTH` (デフォルト) または `PLANAR` のいずれかを指定できます。`ROUND EARTH` モデルでは、点と点を結ぶ線が大楕円弧として解釈されます。地表面の2点を指定し、この2点および地球の中心と交差するように平面を選択したとします。この平面は地球と交差し、2点を結ぶ線はこの交差に沿った最短距離になります。

2点がまったく反対側にある場合、2点および地球の中心と交差する単一でユニークな平面は存在しません。このような正反対の点を結ぶ線セグメントは無効であり、`ROUND EARTH` モデルでエラーになります。

`ROUND EARTH` モデルでは、地球が回転楕円体と見なされ、地球の湾曲に沿った線が選択されます。場合によっては、2点間の線が x = 経度、 y = 緯度の正距円筒図法における直線として解釈される平面モデルの使用が必要となる場合があります。

次の例では、青い線は `ROUND EARTH` モデルで使用される線の解釈を示し、赤い線は対応する `PLANAR` モデルを示します。



PLANAR モデルは、他の製品で使用される解釈と一致させるために使用できます。また、ROUND EARTH モデルでサポートされていないメソッド (ST_Area、ST_ConvexHull など) には制限があり、一部のメソッドは部分的にのみサポートされている (ST_Distance はポイントジオメトリでのみサポートされる) ため、PLANAR モデルも便利です。円ストリングに基づいたジオメトリは、ROUND EARTH 空間参照系ではサポートされていません。

非地理的 SRS の場合には、タイプを PLANAR にしてください (これが、TYPE 句が指定されておらず、DEFINITION 句が指定されていないか非地理的定義を使用している場合のデフォルトです)。

- **COORDINATE** – 空間参照系の次元の境界。coordinate-name は、空間参照系で使用される座標系の名前です。非地理的座標系の場合は、coordinate-name に x、y、または m を使用できます。地理的座標系の場合は、coordinate-name に LATITUDE、LONGITUDE、z、または m を使用できます。

次元で境界を配置しない場合は、UNBOUNDED を指定します。境界の上限と下限を設定するには、BETWEEN 句を使用します。

X 座標と Y 座標には、関連する境界が必要です。地理的座標系の場合は、これらの設定を COORDINATE 句で上書きしないかぎり、デフォルトにより経度座標は -180 ~ 180 度の間、緯度座標は -90 ~ 90 の間になります。非地理的座標系の場合は、X 座標と Y 座標の両方について CREATE 文で境界を指定してください。

LATITUDE と LONGITUDE は、地理的座標系に使用されます。LATITUDE と LONGITUDE の境界は、指定がなければデフォルトにより地球全体になります。

- **ELLIPSOID** – タイプ ROUND EARTH の空間参照系の楕円として、地球を表現するとき使用する値。DEFINITION 句が指定されている場合は、この句で楕円定義を指定できます。ELLIPSOID 句が指定されていると、このデフォルトの楕円が上書きされます。

地球の自転が引き起こす扁平化により地球の中心から北極または南極までの距離が赤道までの距離より短くなるため、地球は完全な球体ではありません。このため、地球は、半長径 (中心から赤道までの距離) と半短径 (中心から極までの距離) に異なる値を使用する楕円としてモデル化されます。半長径と逆扁平率を使用して楕円を定義するのが最も一般的ですが、代わりに半短径を使用して指定することもできます (たとえば、完全な球形を使用して地球を近似させる場合には、この方法を使用します)。半長径と半短径は空間参照系の線形単位で定義され、逆扁平率 (1/f) は次の式で求められる比率です。

$$1/f = (\text{semi-major-axis}) / (\text{semi-major-axis} - \text{semi-minor-axis})$$

product-name では、地理的空間参照系で距離を計算するとき楕円定義が使用されます。

- **SNAP TO GRID** – 平らな地球 (平面) の空間参照系では、SNAP TO GRID 句を使用して、SAP Sybase IQ で計算に使用されるグリッドサイズを定義します。デフォルトでは、X と Y の空間の境界におけるすべての点で 12 有効桁数を格納できるように SAP Sybase IQ でグリッドサイズが選択されます。たとえば、空間参照系の境界が X は -180 ~ 180 の間、Y は -90 ~ 90 の間の場合、グリッドサイズ 0.000000001 (1E-9) が選択されます。
- **TOLERANCE** – 平らな地球 (平面) の空間参照系では、TOLERANCE 句を使用して、点を比較するとき使用する精度を指定します。2 点間の距離が tolerance-distance より小さい場合、その 2 点は同じと見なされます。tolerance-distance を指定することにより、入力データまたは制限された内部精度の不正確さに対する許容度を制御できます。デフォルトでは、tolerance-distance は grid-size と等しくなるように設定されます。

0 に設定すると、2 点が完全に等しい場合にのみ同じであると見なされます。

曲面の空間参照系では、TOLERANCE を 0 に設定してください。

- **POLYGON FORMAT** – SAP Sybase IQ では、まず、多角形を構成するリングの方向によって多角形が解釈されます。定義された点の順序でリングを移動した場合、多角形の内側がリングの左側になります。平面と曲面の空間参照系では、同じルールが適用されます。

SAP Sybase IQ で使用される解釈は一般的ですが、汎用的な解釈ではありません。正反対の方向を使用する製品もあれば、多角形の解釈にリングの方向を使用しない製品もあります。POLYGON FORMAT 句を使用して、必要に応じて入

力データと一致する多角形の解釈を選択できます。次の値がサポートされません。

- **CounterClockwise** – 入力 は SAP Sybase IQ の内部解釈に従います。リングの方向に従って、多角形の内側が左側になります。
- **Clockwise** – 入力 は SAP Sybase IQ の内部解釈と反対になります。リングの方向に従って、多角形の内側が右側になります。
- **EvenOdd** – (デフォルト) リングの方向は無視され、多角形の内部は代わりにリングのネストによって決定されます。外部のリングは最大のリングとなり、内部のリングはこのリングの内側の小さなリングとなります。斜線は、リング内の点から、すべてのリングを交差して外側に向かってトレースされます。交差するリングの数が偶数の場合は、外部リングです。奇数の場合は、内部リングです。
- **STORAGE FORMAT** – 空間データをデータベースにロードするときの格納内容を制御します。考えられる値は、次のとおりです。
 - **Internal** – SAP Sybase IQ は正規化された表現のみを格納します。元の入力特性を再現する必要がない場合に指定します。これは、平面 (TYPE PLANAR) の空間参照系のデフォルトです。
 - **Original** – SAP Sybase IQ は元の表現のみを格納します。元の入力特性を再現できますが、格納された値のすべての操作に対して正規化の手順を繰り返す必要があり、データの操作が遅くなる可能性があります。
 - **Mixed** – SAP Sybase IQ は内部バージョンを格納します。元のバージョンと異なる場合、SAP Sybase SQL Anywhere® は元のバージョンも格納します。両方のバージョンを格納すると、元の表現特性を再現でき、格納された値の操作に対して正規化の手順を繰り返す必要はありません。ただし、ジオメトリごとに 2 つの表現が格納される可能性があるため、記憶領域の要件が大幅に増加します。Mixed は、曲面の空間参照系 (TYPE ROUND EARTH) のデフォルトフォーマットです。

例

(先頭に戻る) (60 ページ)

- **例** – mySpatialRef という架空の空間参照系の多角形フォーマットを EvenOdd に変更します。

```
ALTER SPATIAL REFERENCE SYSTEM mySpatialRef
POLYGON FORMAT 'EvenOdd';
```

使用法

(先頭に戻る) (60 ページ)

空間参照系を参照している既存のデータがある場合、その空間参照系を変更することはできません。たとえば、ST_Point(SRID=8743) として宣言したカラムがある

場合、SRID 8743 を持つ空間参照系は変更できません。これは、格納フォーマットなどの空間参照系の多くの属性が、データの格納フォーマットに影響を及ぼすためです。SRID を参照するデータがある場合は、新しい空間参照系を作成して、データを新しい SRID に変換します。

標準

(先頭に戻る) (60 ページ)

ANSI SQL – 準拠レベル： Transact-SQL 拡張。

パーミッション

(先頭に戻る) (60 ページ)

次のいずれかが必要です。

- その空間参照系を所有している
- 空間参照系に対する ALTER 権限
- MANAGE ANY SPATIAL OBJECT システム権限
- ALTER ANY OBJECT システム権限

ALTER TABLE 文

テーブルの定義を変更します。

クイックリンク：

「パラメータ」 (70 ページ)

「例」 (78 ページ)

「使用法」 (81 ページ)

「標準」 (82 ページ)

「パーミッション」 (82 ページ)

構文

構文 1 - Alter Owner

```
ALTER TABLE table_name ALTER OWNER TO new_owner
  [ { PRESERVE | DROP } PERMISSIONS ]
  [ { PRESERVE | DROP } FOREIGN KEYS ]
```

構文 2

```
ALTER TABLE [ owner. ] table-name
  [ { ENABLE | DISABLE } RLV STORE
```

```

{ alter-clause, ... }

alter-clause - (back to Syntax 2)
  ADD create-clause
  | ALTER column-name column-alteration
  | ALTER [ CONSTRAINT constraint-name ] CHECK ( condition )
  | DROP drop-object
  | RENAME rename-object
  | move-clause
  | SPLIT PARTITION range-partition-name
      INTO ( range-partition-decl, range-partition-decl )
  | MERGE PARTITION partition-name-1 INTO partition-name-2
  | UNPARTITION
  | PARTITION BY
      range-partitioning-scheme
      | hash-partitioning-scheme
      | composite-partitioning-scheme

create-clause - (back to alter-clause)
  column-name column-definition [ column-constraint ]
  | table-constraint
  | [ PARTITION BY ] range-partitioning-scheme

column definition - (back to create-clause)
  column-name data-type [ NOT NULL | NULL ]
  [ IN dbspace-name ]
  [ DEFAULT default-value | IDENTITY ]

column-constraint - (back to create-clause)
  [ CONSTRAINT constraint-name ]
  { UNIQUE
    | PRIMARY KEY
    | REFERENCES table-name [ ( column-name ) ] [ actions ]
    | CHECK ( condition )
    | IQ UNIQUE ( integer )
  }

table-constraint - (back to create-clause)
  [ CONSTRAINT constraint-name ]
  { UNIQUE ( column-name [ , ... ] )
    | PRIMARY KEY ( column-name [ , ... ] )
    | foreign-key-constraint
    | CHECK ( condition )
  }

foreign-key-constraint - (back to table-constraint)
  FOREIGN KEY [ role-name ] [ ( column-name [ , ... ] ) ]
  ... REFERENCES table-name [ ( column-name [ , ... ] ) ]
  ... [ actions ]

actions - (back to foreign-key-constraint)
  [ ON { UPDATE | DELETE } { RESTRICT } ]

column-alteration - (back to alter-clause)
  { column-data-type | alterable-column-attribute } [ alterable-column-attribute ... ]

```



```

| ADD [ constraint-name ] CHECK ( condition )
| DROP { DEFAULT | CHECK | CONSTRAINT constraint-name }

alterable-column-attribute - (back to column-alteration)
[ NOT ] NULL
| DEFAULT default-value
| [ CONSTRAINT constraint-name ] CHECK { NULL |( condition )
}

default-value - (back to alterable-column-attribute)
CURRENT { DATABASE | DATE | REMOTE USER | TIME | TIMESTAMP | USER |
PUBLISHER )
| string
| global variable
| [ - ] number
| ( constant-expression )
| built-in-function ( constant-expression )
| AUTOINCREMENT
| NULL
| TIMESTAMP
| LAST USER
| USER

drop-object - (back to alter-clause)
{ column-name
| CHECK constraint-name
| CONSTRAINT
| UNIQUE ( index-columns-list )
| PRIMARY KEY
| FOREIGN KEY fkey-name
| [ PARTITION ] range-partition-name
}

rename-object - (back to alter-clause)
new-table-name
| column-name TO new-column-name
| CONSTRAINT constraint-name TO new-constraint-name
| [ PARTITION ] range-partition-name TO new-range-partition-name

move-clause - (back to alter-clause)
{ ALTER column-name
MOVE
{ PARTITION ( range-partition-name TO new-dbspace-name)
| TO new-dbspace-name }
}
| MOVE PARTITION range-partition-name TO new-dbspace-name
| MOVE TO new-dbspace-name
| MOVE TABLE METADATA TO new-dbspace-name
}

range-partitioning-scheme - (back to alter-clause)
RANGE( partition-key )
( range-partition-decl [, range-partition-decl ... ] )

partition-key - (back to range-partitioning-scheme)

```

```

column-name

range-partition-decl - (back to alter-clause) or (back to range-
partitioning-scheme)
  range-partition-name VALUES <= ( {constant | MAX } ) [ IN dbspace-
name ]

hash-partitioning-scheme - (back to alter-clause) or (back to composite-
partitioning-scheme)
  HASH ( partition-key, ... ] )

composite-partitioning-scheme - (back to alter-clause)
  hash-partitioning-scheme SUBPARTITION range-partitioning-scheme

```

パラメータ

(先頭に戻る) (67 ページ)

- **{ ENABLE | DISABLE } RLV STORE** - リアルタイムのインメモリ更新用に、このテーブルを RLV ストアに登録します。IQ テンポラリテーブルまたはマルチプレックス環境ではサポートされていません。この値は、データベースオプション **BASE_TABLES_IN_RLV** の値よりも優先されます。
- **ADD column-definition [column-constraint]** - テーブルに新しいカラムを追加します。

NOT NULL を指定するには、テーブルが空であることが必要です。IDENTITY カラムまたは DEFAULT AUTOINCREMENT カラムの追加時に、テーブルにデータが含まれていてもかまいません。カラムにデフォルトの IDENTITY 値が指定されていれば、新しいカラムのすべてのローに連続する値が入力されます。また、1つのカラムキーに、FOREIGN 制約をカラム制約として追加できます。IDENTITY/DEFAULT AUTOINCREMENT カラムの値は、テーブル内の各ローをユニークに識別します。

IDENTITY/DEFAULT AUTOINCREMENT カラムには、挿入や更新の際に自動的に生成される連続した数値が格納されます。DEFAULT AUTOINCREMENT カラムは、IDENTITY カラムとも呼ばれます。IDENTITY/DEFAULT AUTOINCREMENT を使用するカラムは、整数データ型のいずれか、または位取りが0の真数値型であることが必要です。カラム制約と IDENTITY/DEFAULT AUTOINCREMENT カラムの詳細については、「CREATE TABLE 文」を参照してください。

IQ UNIQUE 制約 - カラムの予期されるカーディナリティを定義し、カラムをフラット FP または NBit FP のどちらとしてロードするかを決定します。IQ UNIQUE(*n*) の値を明示的に 0 に設定すると、カラムはフラット FP としてロードされます。IQ UNIQUE 制約のないカラムは、FP_NBIT_AUTOSIZE_LIMIT、

FP_NBIT_LOOKUP_MB、および FP_NBIT_ROLLOVER_MAX_MB オプションで定義された上限まで暗黙的に NBit としてロードされます。

IQ UNIQUE を FP_NBIT_AUTOSIZE_LIMIT 未満の n 値とともに使用する必要はありません。自動サイズ機能によって、カーディナリティが低いか中程度のカラムはすべて NBit としてサイズ決定されます。カラムをフラット FP としてロードする場合や、重複しない値の数が FP_NBIT_AUTOSIZE_LIMIT を超えるときにカラムを NBit としてロードする場合は、IQ UNIQUE を使用します。

注意：

- 高い IQ UNIQUE 値を指定する際はメモリ使用率を考慮します。マシンリソースに制限がある場合、FP_NBIT_ENFORCE_LIMITS='OFF' (デフォルト) でロードしないでください。
SAP Sybase IQ 16.0 以前は、16777216 を超える IQ UNIQUE n 値はフラット FP にロールオーバーされていました。16.0 では、より大きい IQ UNIQUE 値がトークン化でサポートされていますが、カーディナリティとカラム幅に応じて、重大なメモリリソース要件が発生する場合があります。
 - BIT、BLOB、および CLOB の各データ型は NBit ディクショナリ圧縮をサポートしません。FP_NBIT_IQ15_COMPATIBILITY='OFF' である場合、これらのデータ型を含む CREATE TABLE 文または ALTER TABLE 文でゼロ以外の IQ UNIQUE カラムを指定すると、エラーが返されます。
-
- **ALTER *column-name* *column-alteration*** – カラム定義を次のように変更します。
 - **SET DEFAULT *default-value*** - テーブルの既存のカラムのデフォルト値を変更します。この作業では MODIFY 句も使用できますが、ALTER は ISO/ANSI SQL に準拠しているのに対して MODIFY は準拠していません。デフォルト値を変更しても、テーブルの既存の値は変更されません。
 - **DROP DEFAULT** - テーブルの既存のカラムのデフォルト値を削除します。この作業では MODIFY 句も使用できますが、ALTER は ISO/ANSI SQL に準拠しているのに対して MODIFY は準拠していません。デフォルトを削除しても、テーブルの既存の値は変更されません。
 - **ADD** - 名前付き制約または CHECK 条件をカラムに追加します。新しい制約または条件は、それを定義した後でテーブルに対して実行される処理のみに適用されます。テーブルの既存の値は、新しい制約や条件を満たすかどうかの検証を受けません。
 - **CONSTRAINT *column-constraint-name*** - オプションのカラム制約名を指定すると、後で、カラム制約全体を修正するのではなく、制約を個別に修正または削除できます。
 - **[CONSTRAINT *constraint-name*] CHECK (*condition*)** - この句はカラムに検査制約を追加するときに使用します。

- **SET COMPUTE** (*expression*) - 計算カラムに関連付けられた式を変更します。この文を実行すると、カラムの値が再計算されます。新しい式が無効な場合、この文は失敗します。
- **DROP COMPUTE** - 計算カラムから非計算カラムに変更します。この文はテーブル内の既存の値を変更しません。
- **ADD table-constraint** - テーブルに制約を追加します。

シングルカラムまたはマルチカラムのキーに、外部キー制約をテーブル制約として追加することもできます。PRIMARY KEY を指定する場合、テーブルには CREATE TABLE 文または別の ALTER TABLE 文で作成したプライマリキーがあってはなりません。テーブル制約の詳細については、「CREATE TABLE 文」を参照してください。

注意： テーブルまたはカラムの制約は MODIFY (変更) できません。制約を変更するには、古い制約を DELETE (削除) し、新しい制約を ADD (追加) します。

- **DROP drop-object** - テーブルオブジェクトを削除します。
 - **DROP column-name** - カラムをテーブルから削除します。カラムがマルチカラムインデックス、一意性制約、外部キー、またはプライマリキーに含まれている場合は、インデックス、制約またはキーを削除してからカラムを削除してください。このようにするとカラムを参照する検査制約は削除されません。IDENTITY/DEFAULT AUTOINCREMENT カラムを削除できるのは、テーブルの IDENTITY_INSERT が OFF に設定され、かつテーブルがローカルテンポラリテーブルでない場合だけです。
 - **DROP CHECK** - テーブルのすべての検査制約を削除します。テーブル検査制約とカラム検査制約の両方が対象となります。
 - **DROP CONSTRAINT constraint-name** - テーブルまたは指定したカラムの名前付き制約を削除します。
 - **DROP UNIQUE** (*column-name, ...*) - 指定したカラムの一意性制約を削除します。一意性制約を参照する外部キー (プライマリキーではなく) も削除します。関連する外部キー制約がある場合は、エラーが報告されます。ALTER TABLE を使用して、プライマリキーを参照するすべての外部キーを削除した後でなければ、プライマリキー制約を削除することはできません。
 - **DROP PRIMARY KEY** - プライマリキーを削除します。このテーブルのプライマリキーを参照するすべての外部キーも削除します。関連する外部キー制約がある場合は、エラーが報告されます。プライマリキーに強制力がない場合、そのプライマリキーに強制力のない外部キー制約が存在すると、DELETE はエラーを返します。
 - **DROP FOREIGN KEY role-name** - 特定のロール名を持つ、該当テーブルの外部キー制約を削除します。その外部キー制約に対して自動的に作成された、

ユニークでない HG インデックスは削除されません。HG インデックスは、DROP INDEX 文を使用して明示的に削除してください。

- **DROP [PARTITION]** - 指定したパーティションを削除します。パーティション P1 内のローとパーティション定義が削除されます。最後のパーティションは削除できません。これは、分割されたテーブルが非分割テーブルに変換されるためです (分割されたテーブルをマージするには、UNPARTITION 句を代わりに使用します)。次に例を示します。

```
CREATE TABLE foo (c1 INT, c2 INT)
PARTITION BY RANGE (c1)
(P1 VALUES <= (100) IN dbbsp1,
 P2 VALUES <= (200) IN dbbsp2,
 P3 VALUES <= (MAX) IN dbbsp3
 ) IN dbbsp4);
LOAD TABLE ...
ALTER TABLE DROP PARTITION P1;
```

- **RENAME *rename-object*** - テーブル内のオブジェクトの名前を変更します。
 - **RENAME *new-table-name*** - テーブルの名前を *new-table-name* に変更します。古いテーブル名を使用しているアプリケーションは、修正する必要があります。また、古いテーブル名と同じ名前が自動的に割り当てられた外部キーの名前は、変更しません。
 - **RENAME *column-name* TO *new-column-name*** - カラムの名前を *new-column-name* に変更します。古いカラム名を使用しているアプリケーションがある場合は、修正が必要になります。
 - **RENAME [PARTITION]** - 既存のパーティション名を変更します。
 - **RENAME *constraint-name* TO *new-constraint-name*** - 制約の名前を *new-constraint-name* に変更します。古い制約名を使用しているアプリケーションがある場合は、修正が必要になります。
- **MOVE clause** - テーブルオブジェクトを移動します。各テーブルオブジェクトは 1 つの DB 領域にのみ置くことができます。どのタイプの ALTER MOVE も、移動中はテーブルへの変更をすべてブロックします。

注意： オブジェクトをキャッシュ DB 領域に移動することはできません。

- **MOVE TO** - テーブルが新しい DB 領域にマッピングされると、そのテーブルと同じ DB 領域に存在するカラム、インデックス、一意性制約、プライマリキー、外部キー、メタデータなどのすべてのテーブルオブジェクトを新しい DB 領域に移動します。ALTER Column MOVE TO 句は、分割テーブルに対しては要求できません。

BIT データ型のカラムは DB 領域に明示的に配置することはできません。以下は BIT データ型に対してサポートされていません。

```
ALTER TABLE t2 alter c1_bit MOVE TO iq_main;
```

- **MOVE TABLE METADATA** - テーブルのメタデータを新しい DB 領域に移動します。分割されたテーブルでは、MOVE TABLE METADATA はパーティション間で共有されるメタデータも移動します。
- **MOVE PARTITION** - 指定したパーティションを新しい DB 領域に移動します。
- **PARTITION BY** - 大きなテーブルを、より小さく管理しやすいストレージオブジェクトに分割します。各パーティションは親テーブルと同じ論理属性を共有しますが、別々の DB 領域に配置して個別に管理できます。SAP Sybase IQ は、次のような複数のテーブル分割スキームをサポートしています。
 - ハッシュパーティション
 - 範囲パーティション
 - 複合パーティション

partition-key は、テーブル分割キーが格納されている 1 つまたは複数のカラムです。分割キーには、NULL 値 および DEFAULT 値を含めることができますが、次のカラムを含めることはできません。

- LOB (BLOB または CLOB) カラム
- BINARY または VARBINARY カラム
- 長さが 255 バイトを超える CHAR または VARCHAR カラム
- BIT カラム
- FLOAT/DOUBLE/REAL カラム
- **PARTITION BY RANGE** - 分割カラム内の値の範囲によってローを分割します。範囲分割は、単一の分割キーカラムおよび最大 1024 パーティションまでに制限されています。range-partitioning-scheme 内の partition-key は、テーブル分割キーが格納されているカラムです。

```
range-partition-decl:
  partition-name VALUES <= ( {constant-expr | MAX } [ ,
  { constant-expr | MAX } ]... )
  [ IN dbspace-name ]
```

partition-name は、テーブルローが格納される新しいパーティションの名前です。パーティション名は、テーブル上にあるパーティションセット内でユニークである必要があります。パーティション名は必須です。

- **VALUE** - 各パーティションの包括的な上限を (昇順に) 指定します。ユーザは、各ローが 1 つのパーティションのみに分配されるように、各範囲分割の分割基準を指定する必要があります。NULL は分割カラムに使用でき、NULL を分割キー値に含んだローは最初のテーブル分割に属します。ただし、NULL をバインド値に指定することはできません。

最初のパーティションには、下限 (MIN 値) は設定されていません。分割キーの最初のカラムにある NULL セルのローは、最初のパーティションに

移動します。最後のパーティションでは、包括的な上限または MAX を指定できます。最後のパーティションの上限値が MAX でない場合は、最後のパーティションの上限値よりも大きい分割キーの値を含んだローをロードまたは挿入すると、エラーが生成されます。

- **MAX** – 無制限の上限を示し、最後のパーティションに対してのみ指定できます。
- **IN-partition-decl** でパーティションのローが存在する DB 領域を指定します。

次の制限を設定すると、範囲分割されたテーブルの分割キーとバインド値がその影響を受けます。

- 未分割のテーブルを範囲分割できるのは、すべての既存のローが最初のパーティションに属する場合のみです。
- パーティションバインドは定数式でなく、定数として指定する必要があります。
- パーティションバインドは、パーティションの作成順に応じて、昇順で指定する必要があります。つまり、2 番目のパーティションの上限は最初のパーティションよりも高く指定する必要がある、というようになります。さらに、パーティションバインドの値は、対応する分割キーカラムのデータ型と互換性がなければなりません。たとえば、VARCHAR は CHAR と互換性があります。
- バインド値に対応する分割キーのカラムとは異なるデータ型が指定されていると、SAP Sybase IQ はバインド値を分割キーのカラムのデータ型に変換します。ただし、次の場合は例外となります。
- 明示的な変換は使用できません。この例では、INT から VARCHAR に明示的に変換しようとしてエラーが生成されます。

```
CREATE TABLE Employees(emp_name VARCHAR(20))
PARTITION BY RANGE(emp_name)
(p1 VALUES <=(CAST (1 AS VARCHAR(20))),
p2 VALUES <= (CAST (10 AS VARCHAR(20)))
```

- データロスにつながる暗黙的な変換は使用できません。この例では、パーティションバインドは分割キー型と互換性がありません。丸めを前提で処理を行うとデータロスにつながる可能性があり、エラーが生成されます。

```
CREATE TABLE emp_id (id INT) PARTITION BY RANGE(id) (p1 VALUES
<= (10.5), p2 VALUES <= (100.5))
```

- この例では、パーティションバインドと分割キーのデータ型の間には互換性があります。バインド値は FLOAT 値に直接変換されます。丸め処理は必要なく、変換はサポートされています。

```
CREATE TABLE id_emp (id FLOAT)
PARTITION BY RANGE(id) (p1 VALUES <= (10),
p2 VALUES <= (100))
```

- 非バイナリデータ型からバイナリデータ型に変換することはできません。たとえば、次の変換は実行できず、エラーが返されます。

```
CREATE TABLE newemp (name BINARY)
PARTITION BY RANGE(name)
```

```
(p1 VALUES <= ("Maarten"),
p2 VALUES <= ("Zymmerman"))
```

- NULL を範囲分割テーブルで境界として使用することはできません。
- 分割キーの最初のカラムのセル値が NULL と評価された場合、ローは最初のパーティションに挿入されます。SAP Sybase IQ は、1つのカラムの分割キーのみをサポートしているため、分割キー内に NULL が含まれていると、ローは最初のパーティションに分配されます。
- **PARTITION BY HASH** – 内部ハッシュ関数によって処理された分割キーの値に基づいて、データをパーティションにマップします。ハッシュ分割キーは最大 8 カラムで、組み合わせた宣言カラム幅が 5300 バイト以下に制限されています。ハッシュパーティションの場合、テーブル作成者は分割キーカラムのみを決定します。パーティションの数と位置は内部的に決定されます。

hash-partitioning 宣言内の partition-key は、1つのカラムまたはカラムのグループです。その複合値によってデータの各ローが格納されるパーティションが決まります。

```
hash-partitioning-scheme:
HASH ( partition-key [ , partition-key, ... ] )
```

- **制限事項** –
 - ハッシュ分割できるのはベーステーブルのみです。グローバルテンポラリテーブルやローカルテンポラリテーブルを分割しようとすると、エラーが発生します。
 - 未分割のテーブルをハッシュ分割できるのは、そのテーブルが空の場合のみです。
 - ハッシュパーティションの追加、削除、マージ、分割はできません。
 - カラムをハッシュ分割キーから追加または削除することはできません。
- **PARTITION BY HASH RANGE** – ハッシュ分割されたテーブルを範囲によってさらに分割します。hash-range-partitioning-scheme 宣言内の SUBPARTITION BY RANGE 句は、新しい範囲サブパーティションを既存のハッシュ範囲分割テーブルに追加します。

```
hash-range-partitioning-scheme:
PARTITION BY HASH ( partition-key [ , partition-key, ... ] )
  [ SUBPARTITION BY RANGE ( range-partition-decl [ , range-
partition-decl ... ] ) ]
```

ハッシュパーティションはデータの論理的な配分および配置方法を指定するのに対して、範囲サブパーティションはデータの物理的な配置方法を指定します。新しい範囲サブパーティションは、既存のハッシュ範囲分割テーブルと同じハッシュ分割キーを持つハッシュによって論理的に分割されます。範囲サブパーティションは 1つのカラムに制限されています。

- **制限事項** –

- ハッシュ分割できるのはベーステーブルのみです。グローバルテンポラリテーブルやローカルテンポラリテーブルを分割しようとすると、エラーが発生します。
- ハッシュ分割されたテーブルを範囲によってさらに分割できるのは、そのテーブルが空の場合のみです。
- ハッシュパーティションの追加、削除、マージ、分割はできません。
- カラムをハッシュ分割キーから追加または削除することはできません。

注意： 範囲パーティションと複合分割スキームは、ハッシュ範囲パーティションと同様、個別にライセンスが必要な VLDB Management オプションを必要とします。

- **MERGE PARTITION** – *partition-name-1* を *partition-name-2* にマージします。2つのパーティションが隣接しており、データが同じ DB 領域にある場合は、パーティションをマージできます。低いパーティションの値を持つパーティションを高いパーティションの値を持つ隣接パーティションにマージする場合にのみ、パーティションをマージできます。サーバでは、パーティションのマージ先の DB 領域に対する CREATE 権限がチェックされないことに注意してください。隣接するパーティションを作成する方法の例については、「CREATE TABLE 文」の例を参照してください。
- **RENAME PARTITION** – 既存の PARTITION の名前を変更します。
- **UNPARTITION** – 分割されたテーブルからパーティションを削除します。各カラムは 1 つの DB 領域に配置されます。サーバでは、すべてのパーティションのデータの移動先となる DB 領域に対する CREATE 権限がチェックされないことに注意してください。ALTER TABLE UNPARTITION は、データベースのアクティビティをすべてブロックします。
- **ALTER OWNER** – テーブルの所有者を変更します。ALTER OWNER 句を ALTER TABLE 文の他の [alter-clause] 句と組み合わせて使用することはできません。
- **[PRESERVE | DROP] PERMISSIONS** — 新しい所有者に古い所有者と同じ権限を付与しない場合は、DROP privileges 句 (デフォルト) を指定して、明示的に付与されていたテーブルへのユーザアクセスを許可する権限をすべて取り消すことができます。そのテーブルの所有者の権限が暗黙的に与えられたものである場合、その権限が新しい所有者に付与され、前の所有者から削除されます。
- **[PRESERVE | DROP] FOREIGN KEYS** - 新しい所有者が参照先テーブル内のデータにアクセスできないようにするには、DROP FOREIGN KEYS 句 (デフォルト) を使用して、テーブル内のすべての外部キーと、そのテーブルを参照しているすべての外部キーを削除します。新しい所有者がすべての参

照元テーブルを所有していないかぎり、PRESERVE FOREIGN KEYS 句と DROP PERMISSIONS 句を組み合わせて使用すると失敗します。

ALTER TABLE ALTER OWNER 文は次の場合に失敗します。

- 元のテーブルと同じ名前を持つ別のテーブルが存在し、新しいユーザがそのテーブルを所有している場合。
- PRESERVE FOREIGN KEYS 句と PRESERVE PERMISSIONS 句の両方が指定され、新しいテーブル所有者以外のユーザが所有する外部キーが存在し、その外部キーが暗黙的に付与された権限 (テーブルの所有者に与えられた権限など) に依存するテーブルを参照している場合。この失敗を回避するには、参照元テーブルの元の所有者に対して SELECT 権限を明示的に付与するか、該当する外部キーを削除します。
- PRESERVE FOREIGN KEYS が指定され、PRESERVE PERMISSIONS が指定されていないときに、新しいテーブル所有者以外のユーザが所有する外部キーが存在し、その外部キーがテーブルを参照している場合。この失敗を回避するには、該当する外部キーを削除します。
- PRESERVE FOREIGN KEYS 句が指定され、暗黙的に付与された権限 (テーブルの所有者に与えられた権限など) に依存する外部キーがテーブルに含まれている場合。この失敗を回避するには、新しい所有者に対して参照先テーブルの SELECT 権限を明示的に付与するか、該当する外部キーを削除します。
- テーブル内にシーケンスを参照するデフォルト値を持つカラムがあり、シーケンスジェネレータの USAGE 権限が暗黙的に付与された権限 (シーケンスの所有者に与えられた権限など) に依存する場合。この失敗を回避するには、テーブルの新しい所有者に対してシーケンスジェネレータの USAGE 権限を明示的に付与します。
- 元のテーブルに依存する有効化されたマテリアライズドビューが存在する場合。

例

(先頭に戻る) (67 ページ)

- **例 1** – 従業員の勤務先の事務所を示す新しいカラムを、Employees テーブルに追加します。

```
ALTER TABLE Employees  
ADD office CHAR(20)
```

- **例 2** – office カラムを Employees テーブルから削除します。

```
ALTER TABLE Employees  
DROP office
```

- **例3**—Customers テーブルにカラムを追加して、各顧客の販売担当を割り当てます。

```
ALTER TABLE Customers
ADD SalesContact INTEGER
REFERENCES Employees (EmployeeID)
```

- **例4**—新しいカラム CustomerNum を Customers テーブルに追加して、デフォルト値の 88 を割り当てます。

```
ALTER TABLE Customers
ADD CustomerNum INTEGER DEFAULT 88
```

- **例5**—c2、c4、および c5 の **FP** インデックスを DB 領域 Dsp3 から Dsp6 に移動します。c1 の **FP** インデックスは Dsp1 に残ります。c3 の **FP** インデックスは Dsp2 に残ります。c5 のプライマリキーは Dsp4 に残ります。**DATE** インデックス c4_date は Dsp5 に残ります。

```
CREATE TABLE foo (
    c1 INT IN Dsp1,
    c2 VARCHAR(20),
    c3 CLOB IN Dsp2,
    c4 DATE,
    c5 BIGINT,
    PRIMARY KEY (c5) IN Dsp4) IN Dsp3);

CREATE DATE INDEX c4_date ON foo(c4) IN Dsp5;
ALTER TABLE foo
MOVE TO Dsp6;
```

- **例6**—**FP** インデックス c1 のみを DB 領域 Dsp1 から Dsp7 に移動します。

```
ALTER TABLE foo ALTER c1 MOVE TO Dsp7
```

- **例7**—多数の **ALTER TABLE** 句を使用して、パーティションの移動、分割、名前変更、およびマージを行います。

分割されたテーブルを作成します。

```
CREATE TABLE bar (
    c1 INT,
    c2 DATE,
    c3 VARCHAR(10))
PARTITION BY RANGE(c2)
(p1 VALUES <= ('2005-12-31') IN dbsp1,
 p2 VALUES <= ('2006-12-31') IN dbsp2,
 p3 VALUES <= ('2007-12-31') IN dbsp3,
 p4 VALUES <= ('2008-12-31') IN dbsp4);
INSERT INTO bar VALUES(3, '2007-01-01', 'banana nut');
INSERT INTO bar VALUES(4, '2007-09-09', 'grape jam');
INSERT INTO bar VALUES(5, '2008-05-05', 'apple cake');
```

パーティション p2 を dbsp5 に移動します。

```
ALTER TABLE bar MOVE PARTITION p2 TO DBSP5;
```

パーティション p4 を 2 つのパーティションに分割します。

```
ALTER TABLE bar SPLIT PARTITION p4 INTO
  (P41 VALUES <= ('2008-06-30') IN dbbsp4,
   P42 VALUES <= ('2008-12-31') IN dbbsp4);
```

次の **SPLIT PARTITION** では、データを移動する必要があるため、エラーが報告されます。既存のローが分割後にすべて同じパーティションにあるとはかぎりません。

```
ALTER TABLE bar SPLIT PARTITION p3 INTO
  (P31 VALUES <= ('2007-06-30') IN dbbsp3,
   P32 VALUES <= ('2007-12-31') IN dbbsp3);
```

次のエラーが報告されます。

```
No data move is allowed, cannot split partition p3.
```

次の **SPLIT PARTITION** では、パーティションの境界値が変更されるため、エラーが報告されます。

```
ALTER TABLE bar SPLIT PARTITION p2 INTO
  (p21 VALUES <= ('2006-06-30') IN dbbsp2,
   P22 VALUES <= ('2006-12-01') IN dbbsp2);
```

次のエラーが報告されます。

```
Boundary value for the partition p2 cannot be changed.
```

パーティション p3 を p2 にマージします。高い境界値から低い境界値のパーティションへのマージは使用できないため、エラーが報告されます。

```
ALTER TABLE bar MERGE PARTITION p3 into p2;
```

次のエラーが報告されます。

```
Partition 'p2' is not adjacent to or before partition 'p3'.
```

パーティション p2 を p3 にマージします。

```
ALTER TABLE bar MERGE PARTITION p2 INTO P3;
```

パーティション p1 の名前を p1_new に変更します。

```
ALTER TABLE bar RENAME PARTITION p1 TO p1_new;
```

テーブル bar の分割を解除します。

```
ALTER TABLE bar UNPARTITION;
```

テーブル bar を分割します。このコマンドでは、すべてのローが最初のパーティションに含まれている必要があるため、エラーが報告されます。

```
ALTER TABLE bar PARTITION BY RANGE (c2)
  (p1 VALUES <= ('2005-12-31') IN dbbsp1,
   P2 VALUES <= ('2006-12-31') IN DBSP2,
```

```
P3 VALUES <= ('2007-12-31') IN dbsp3,
P4 VALUES <= ('2008-12-31') IN dbsp4);
```

次のエラーが報告されます。

```
All rows must be in the first partition.
```

テーブル bar を分割します。

```
ALTER TABLE bar PARTITION BY RANGE (c2)
  (p1 VALUES <= ('2008-12-31') IN dbsp1,
  P2 VALUES <= ('2009-12-31') IN dbsp2,
  P3 VALUES <= ('2010-12-31') IN dbsp3,
  P4 VALUES <= ('2011-12-31') IN dbsp4);
```

- **例 8** – テーブル tab1 を変更して、インメモリリアルタイム更新の対象として RLV ストアに登録されないようにします。

```
ALTER TABLE tab1 DISABLE RLV STORE
```

使用法

(先頭に戻る) (67 ページ)

ALTER TABLE 文は、以前作成したテーブルのテーブル属性(カラム定義、制約)を変更します。構文では複数の ALTER 句を使用できますが、1つの ALTER TABLE 文の中では1つの table-constraint または column-constraint しか追加、修正、削除できないことに注意してください。ALTER TABLE 文は、他の接続で現在使用中のテーブルに影響を及ぼす場合は実行できません。ALTER TABLE は処理に時間がかかり、この文の処理中は、同じテーブルを参照する要求がサーバで処理されません。

注意： ローカルテンポラリテーブルは変更できませんが、グローバルテンポラリテーブルは、テーブルを使用する接続が1つだけの場合には変更できます。

SAP Sybase IQ は、REFERENCES 制約と CHECK 制約を適用します。ALTER TABLE 文で追加されるテーブルまたはカラムの検査制約は、追加される新しいカラムのいずれかに対して定義された場合にのみ、ALTER TABLE 操作の中で評価されます。CHECK 制約の詳細については、「CREATE TABLE 文」を参照してください。

ビュー定義に **SELECT *** を使用し、その **SELECT *** で参照されるテーブルを変更する場合は、**ALTER VIEW <viewname> RECOMPILE** を実行してビュー定義を訂正し、ビューをクエリしたときに予期しない結果が返されるのを防ぐ必要があります。

関連する動作:

SQL 文

- オートコミット。ALTER と DROP オプションは現在の接続に対するすべてのカーソルをクローズします。Interactive SQL のデータウィンドウもクリアされます。
- ALTER TABLE 操作の開示時にチェックポイントを実行します。
- カラムまたはテーブルを変更すると、その変更したカラムを参照するストアードプロシージャ、ビューなどは機能しなくなります。

標準

(先頭に戻る) (67 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - 一部の句は SAP Adaptive Server® Enterprise でサポートされています。

パーミッション

(先頭に戻る) (67 ページ)

構文 1

次のいずれかが必要です。

- ALTER ANY TABLE システム権限
- ALTER ANY OBJECT システム権限
- そのテーブルに対する ALTER 権限
- テーブルを所有していること

構文 2

構文 1 に必要なシステム権限は、使用する句によって異なります。

句	必要な権限
ADD	<p>次のいずれかが必要。</p> <ul style="list-style-type: none"> • ALTER ANY TABLE システム権限 • ALTER ANY OBJECT システム権限 • 基礎となるテーブルに対する ALTER 権限 • 基本となるテーブルを所有していること <p>UNIQUE、PRIMARY KEY、FOREIGN KEY、または IQ UNIQUE カラム制約には、上記とともに基本となるテーブルの REFERENCES 権限が必要。</p> <p>FOREIGN KEY テーブル制約には、上記とともに次のいずれかが必要。</p> <ul style="list-style-type: none"> • CREATE ANY INDEX システム権限 • CREATE ANY OBJECT システム権限 • ベーステーブルに対する REFERENCES 権限 <p>PARTITION BY RANGE には、上記とともに次のいずれかが必要。</p> <ul style="list-style-type: none"> • CREATE ANY OBJECT システム権限 • パーティションが作成される DB 領域に対する CREATE 権限
ALTER	<p>次のいずれかが必要。</p> <ul style="list-style-type: none"> • ALTER ANY TABLE システム権限 • ALTER ANY OBJECT システム権限 • そのテーブルに対する ALTER 権限 • テーブルを所有していること <p>プライマリキーまたは一意性制約を変更するには、そのテーブルに対する REFERENCES 権限も必要。</p>

句	必要な権限
DROP	<p>制約を持たないカラムを削除するには、次のいずれかが必要。</p> <ul style="list-style-type: none"> • ALTER ANY OBJECT システム権限 • ALTER ANY TABLE システム権限 • 基礎となるテーブルに対する ALTER 権限 • 基本となるテーブルを所有していること <p>ALTER 権限を使用する場合、制約を持つカラムまたはテーブルを削除するには、上記とともに REFERENCES 権限が必要。</p> <p>自分が所有するテーブルのパーティションを削除する場合は、何も必要ない。</p> <p>他のユーザが所有するテーブルのパーティションを削除する場合は、次のいずれかが必要。</p> <ul style="list-style-type: none"> • ALTER ANY TABLE システム権限 • ALTER ANY OBJECT システム権限 • そのテーブルに対する ALTER 権限
RENAME	<p>次のいずれかが必要。</p> <ul style="list-style-type: none"> • ALTER ANY TABLE システム権限 • ALTER ANY OBJECT システム権限 • そのテーブルに対する ALTER 権限 • テーブルを所有していること
MOVE	<p>次のいずれかが必要。</p> <ul style="list-style-type: none"> • ALTER ANY TABLE システム権限 • ALTER ANY OBJECT システム権限 • システム権限 • 基礎となるテーブルに対する ALTER 権限 • 基本となるテーブルを所有していること <p>次のいずれかも必要。</p> <ul style="list-style-type: none"> • CREATE ANY OBJECT システム権限 • パーティションの移動先となる DB 領域に対する CREATE 権限

句	必要な権限
SPLIT PARTITION	<p>自分が所有するテーブルのパーティションの場合は、何も必要ない。 他のユーザが所有するテーブルのパーティションの場合は、次のいずれかが必要。</p> <ul style="list-style-type: none"> • SELECT ANY TABLE システム権限 • テーブルに対する SELECT 権限 <p>次のいずれかも必要。</p> <ul style="list-style-type: none"> • ALTER ANY TABLE システム権限 • ALTER ANY OBJECT システム権限 • そのテーブルに対する ALTER 権限
MERGE PARTITION、 UNPARTITION	<p>自分が所有するテーブルの場合は、何も必要ない。 他のユーザが所有するテーブルの場合は、次のいずれかが必要。</p> <ul style="list-style-type: none"> • ALTER ANY TABLE システム権限 • ALTER ANY OBJECT システム権限 • そのテーブルに対する ALTER 権限
PARTITION BY	<p>次のいずれかが必要。</p> <ul style="list-style-type: none"> • CREATE ANY OBJECT システム権限 • パーティションが作成される DB 領域に対する CREATE 権限 <p>次のいずれかも必要。</p> <ul style="list-style-type: none"> • ALTER ANY TABLE システム権限 • ALTER ANY OBJECT システム権限 • そのテーブルに対する ALTER 権限 • テーブルを所有していること
/DISABLE RLV STORE	<p>次のいずれかが必要。</p> <ul style="list-style-type: none"> • ALTER ANY TABLE システム権限 • ALTER ANY OBJECT システム権限

参照：

- CREATE TABLE 文 (257 ページ)
- DROP 文 (319 ページ)
- IDENTITY_INSERT オプション (684 ページ)
- FP_NBIT_AUTOSIZE_LIMIT オプション (667 ページ)
- FP_NBIT_ENFORCE_LIMITS オプション (668 ページ)

- FP_NBIT_LOOKUP_MB オプション (672 ページ)
- FP_NBIT_ROLLOVER_MAX_MB オプション (673 ページ)

ALTER TEXT INDEX 文

TEXT インデックスの名前の変更、移動、または定義の変更を行います。

注意： この文は、非構造化データ分析 (IQ_UDA) ライセンスを必要とします。

クイックリンク：

「パラメータ」 (86 ページ)

「例」 (86 ページ)

「使用法」 (87 ページ)

「パーミッション」 (87 ページ)

構文

```
ALTER TEXT INDEX [owner.] text-index-name
  ON [owner.] table-name
  alter-clause
```

alter-clause - (構文に戻る)

rename-object | **move-object**

rename-object - (back to alter-clause)

RENAME { **AS** | **TO** } *new-name*

move-object - (back to alter-clause)

MOVE TO *dbspace-name*

パラメータ

(先頭に戻る) (86 ページ)

- **RENAME** – TEXT インデックスの名前を変更します。
- **MOVE** – 指定した DB 領域に TEXT インデックスを移動します。

例

(先頭に戻る) (86 ページ)

- **例** – TEXT インデックス `MyTextIndex` を作成し、`IMMEDIATE REFRESH` として定義します。TEXT インデックスの名前を `Text_index_daily` に変更し、この TEXT インデックスを `tispace` という名前の DB 領域に移動します。

```
CREATE TEXT INDEX MyTextIndex ON Customers ( CompanyName )
IMMEDIATE REFRESH;
ALTER TEXT INDEX MyTextIndex ON Customers RENAME AS
Text_index_daily;
ALTER TEXT INDEX Text_Index_Daily ON Customers MOVE TO tspace;
```

使用法

(先頭に戻る) (86 ページ)

関連する動作：

- オートコミット

パーミッション

(先頭に戻る) (86 ページ)

move-object 句 - 次のいずれかが必要です。

- ALTER ANY INDEX システム権限
- ALTER ANY OBJECT システム権限
- 基本となるテーブルに対する REFERENCES 権限
- 基本となるテーブルを所有している

rename-object 句 - 次のいずれかが必要です。

- ALTER ANY INDEX システム権限
- ALTER ANY OBJECT システム権限
- MANAGE ANY DBSPACE システム権限
- 次のいずれかが必要です。
 - インデックスの基本となるテーブルを所有している
 - そのテーブルに対する REFERENCES 権限に加え、さらに次のいずれかが必要です。
 - CREATE ANY OBJECT システム権限
 - ターゲット DB 領域に対する CREATE 権限

ALTER TEXT CONFIGURATION 文

テキスト設定オブジェクトを変更します。

注意： この文は、非構造化データ分析 (IQ_UDA) ライセンスを必要とします。

クイックリンク：

「パラメータ」 (88 ページ)

「例」 (89 ページ)

「使用法」 (90 ページ)

「パーミッション」 (90 ページ)

構文

```
ALTER TEXT CONFIGURATION [ owner.] config-name
  STOPLIST stoptlist
  | DROP STOPLIST
  | { MINIMUM | MAXIMUM } TERM LENGTH integer
  | TERM BREAKER
  | { GENERIC
    | [ EXTERNAL NAME library-and-entry-point-name-string ]
    | NGRAM }
  | PREFILTER EXTERNAL NAME library-and-entry-point-name-string
  | DROP PREFILTER
```

パラメータ

(先頭に戻る) (87 ページ)

- **stoptlist** – TEXT インデックスの構築時に無視する単語のリストを作成したり、置き換えたりする場合に使用する文字列式。このリストで指定されている単語は、クエリでも無視されます。ストップリストの単語はスペースで区切ります。

ストップリストの単語にスペースを含めることはできません。また、英数字以外の文字を使用しないでください。英数字以外の文字はスペースとして解釈され、単語が複数の単語に分割されます。たとえば、"and/or" は、2つの単語 "and" と "or" として解釈されます。ストップリストの単語の最大数は 7999 です。

- **DROP STOPLIST** – テキスト設定オブジェクトのストップリストを削除する場合に使用します。
- **MINIMUM TERM LENGTH** – TEXT インデックスに含める単語の最小長 (文字数) を指定します。NGRAM TEXT インデックスを使用する場合、MINIMUM TERM LENGTH 句に指定された値は無視されます。

この設定より短い単語は、TEXT インデックスの構築時または更新時に無視されます。このオプションの値は 0 より大きくする必要があります。このオプションを MAXIMUM TERM LENGTH より大きい値に設定すると、MAXIMUM TERM LENGTH の値が自動的に調整され、MINIMUM TERM LENGTH の新しい値と同じになります。

- **MAXIMUM TERM LENGTH** – GENERIC TEXT インデックスを使用する場合、TEXT インデックスに含める単語の最大長 (文字数) を指定します。この設定より長い単語は、TEXT インデックスの構築時または更新時に無視されます。

MAXIMUM TERM LENGTH の値は 60 以下にする必要があります。このオプションを MINIMUM TERM LENGTH より小さい値に設定すると、MINIMUM TERM LENGTH の値が自動的に調整され、MAXIMUM TERM LENGTH の新しい値と同じになります。

- **TERM BREAKER** – カラム値を単語に分割するために使用するアルゴリズムの名前を指定します。IN SYSTEM テーブルでの選択肢は、GENERIC (デフォルト) または NGRAM です。GENERIC アルゴリズムは、英数字以外の文字で区切られた 1 つまたは複数の英数字の文字列を単語として扱います。

NGRAM アルゴリズムは文字列を N-gram に分割します。N-gram は、ある文字列中の n 文字分の部分文字列です。NGRAM 単語分割は、ファジー (近似) 一致、または単語の区切りにスペースや英数字以外の文字を使用しないドキュメントに必要です。NGRAM は IN SYSTEM テーブルでサポートされます。

NGRAM 単語分割は TEXT インデックスに構築されます。そのため、テキスト設定オブジェクトの設定を使用して、NGRAM と GENERIC のどちらの TEXT インデックスを使用するかを定義します。

TERM BREAKER には、EXTERNAL NAME とライブラリのエントリポイントを使用して、外部の単語分割ライブラリの指定を含めることができます。

- **library-and-entry-point-name-string** – *[operating-system:]function-name@library*
- **PREFILTER EXTERNAL NAME** – 外部ベンダによって提供される外部のプレフィルタライブラリのエントリポイント (entry_point) とライブラリ名を指定します。
- **DROP PREFILTER** – 外部のプレフィルタを削除し、ISYSTETEXTCONFIG テーブルのプレフィルタカラムに NULL を設定します。

例

(先頭に戻る) (87 ページ)

- **例 1** – テキスト設定オブジェクト maxTerm16 を作成し、単語の最大長を 16 に変更します。

```
CREATE TEXT CONFIGURATION maxTerm16 FROM default_char;
ALTER TEXT CONFIGURATION maxTerm16 MAXIMUM TERM LENGTH 16;
```

- **例 2** – ストップリストの単語を maxTerm16 設定オブジェクトに追加します。

```
ALTER TEXT CONFIGURATION maxTerm16
STOPLIST 'because about therefore only';
```

- **例3**–外部ライブラリ mytermbreaker.dll のエントリポイント my_term_breaker を使用してテキストを分割するように、テキスト設定オブジェクト my_text_config を更新します。

```
CREATE TEXT CONFIGURATION my_text_config FROM default_char;
ALTER TEXT CONFIGURATION my_text_config
TERM BREAKER GENERIC EXTERNAL NAME
'platform:my_term_breaker@mytermbreaker';
```

- **例4**–外部ライブラリ myprefilter.dll のエントリポイント my_prefilter を使用してドキュメントのプレフィルタを実行するように、テキスト設定オブジェクト my_text_config を更新します。

```
ALTER TEXT CONFIGURATION my_text_config
PREFILTER EXTERNAL NAME 'platform:my_prefilter@myprefilter';
```

使用法

(先頭に戻る) (87 ページ)

TEXT インデックスは、テキスト設定オブジェクトに依存しています。SAP Sybase IQ の TEXT インデックスでは即時更新が使用され、トランケートすることはできません。テキスト設定オブジェクトを変更する際は、その前にインデックスを削除する必要があります。テキスト設定オブジェクトの設定を表示するには、SYSTEXTCONFIG システムビューに対してクエリを実行します。

関連する動作：

- オートコミット。

パーミッション

(先頭に戻る) (87 ページ)

TERM BREAKER 句または **PREFILTER EXTERNAL NAME** 句 - CREATE ANY EXTERNAL REFERENCE システム権限に加え、次のいずれかが必要です。

- ALTER ANY TEXT CONFIGURATION システム権限
- ALTER ANY OBJECT システム権限
- テキスト設定オブジェクトを所有している

その他のすべての句 - ユーザがテキスト設定オブジェクトの所有者であるかどうかに関係なく、ALTER ANY TEXT CONFIGURATION システム権限が必要です。

ALTER TRIGGER 文

トリガの定義を修正したものに置き換えます。新しいトリガの定義全体を ALTER TRIGGER 文にインクルードします。

構文 1 - トリガ定義の変更

```
ALTER TRIGGER trigger-name trigger-definition
```

```
trigger-definition: CREATE TRIGGER 構文
```

構文 2 - トリガ定義の難読化

```
ALTER TRIGGER trigger-name ON [owner.] table-name SET HIDDEN
```

備考

- **構文 1** - ALTER TRIGGER 文の構文は、最初の 1 語を除き、CREATE TRIGGER 文の構文とまったく同じです。

Transact-SQL 形式と Watcom-SQL 形式のどちらの CREATE TRIGGER 構文も使用できます。

- **構文 2** - SET HIDDEN を使用して、関連するトリガの定義を難読化し、解読できないようにします。このトリガはアンロードして、他のデータベースに再ロードできます。

注意： SET HIDDEN 操作は元に戻せません。

権限

基本となるテーブルの所有者であるか、または次のいずれかの権限を持っていることが必要です。

基本となるテーブルに対する ALTER 権限と CREATE ANY OBJECT システム権限

ALTER ANY TRIGGER システム権限

ALTER ANY OBJECT システム権限

別のユーザが所有するビューのトリガを変更するには、ALTER ANY TRIGGER および ALTER ANY VIEW システム権限のどちらかを持っているか、ALTER ANY OBJECT システム権限を持っている必要があります。

関連する動作

オートコミット。

標準と互換性

- **SQL/2008** – ベンダー拡張。

ALTER USER 文

ユーザ設定を変更します。

クイックリンク：

「パラメータ」 (92 ページ)

「例」 (93 ページ)

「使用法」 (94 ページ)

「標準」 (95 ページ)

「パーミッション」 (95 ページ)

構文

構文 1 - データベースユーザの定義を変更します

```
ALTER USER user-name  
  | [ IDENTIFIED BY password ]  
  | [ LOGIN POLICY policy-name ]  
  | [ FORCE PASSWORD CHANGE { ON | OFF } ]
```

構文 2 - LDAP ユーザの識別名 (DN) をリフレッシュします

```
ALTER USER user-name  
  REFRESH DN
```

構文 3 - ユーザのログインポリシーの元の値へ復元します

```
ALTER USER user-name  
  RESET LOGIN POLICY
```

構文 4 – ユーザのログインポリシーで **CHANGE_PASSWORD_DUAL_CONTROL** が有効になっている場合の、ユーザパスワードを変更します

```
ALTER USER user-name  
  IDENTIFIED [ FIRST | LAST ] BY password_part
```

パラメータ

(先頭に戻る) (92 ページ)

- **user-name** – ユーザの名前。

- **IDENTIFIED BY** – ユーザのパスワード。ユーザのログインポリシーで `CHANGE_PASSWORD_DUAL_CONTROL` オプションが有効になっている場合、この句はサポートされません (ERROR)。
- **IDENTIFIED[FIRST | LAST] BY** – ターゲットユーザのログインポリシーで `CHANGE_PASSWORD_DUAL_CONTROL` オプションが有効になっている場合に必須の句。FIRST | LAST キーワードで、定義する二重パスワードの部分を指定します。
- **policy-name** – ユーザを割り当てるログインポリシーの名前。ログインポリシーを指定しないと、変更は行われません。LOGIN POLICY 句が指定されていない場合、変更は行われません。
- **FORCE PASSWORD CHANGE** – ログイン時にユーザが新しいパスワードを指定する必要があるかどうかを制御します。この設定は、ユーザのログインポリシーの `PASSWORD_EXPIRY_ON_NEXT_LOGIN` オプションの設定を上書きします。

注意： この機能は現在、SAP Control Center へのログイン時に実装されなくなっています。ユーザは、パスワードの変更を要求されません。ただし、SAP Control Center 外から (たとえば Interactive SQL を使用して) SAP Sybase IQ にログインする際には要求されます。

- **RESET LOGIN POLICY** – ユーザのログインの設定をログインポリシー内の元の値に戻します。通常は、これによって、ユーザのログイン失敗回数が最大値を超えたか、前回のログイン以降に経過した日数が最大値を超えたために暗黙的に設定されたロックがすべてクリアされます。ログインポリシーをリセットすると、`MAX_FAILED_LOGIN_ATTEMPTS` や `MAX_DAYS_SINCE_LOGIN` などのログインポリシーオプションの上限を超えたためにロックされていたアカウントに、ユーザがアクセスできるようになります。
- **REFRESH DN** – LDAP 認証時に使用される、ユーザの保存済みの DN およびタイムスタンプをクリアします。

例

(先頭に戻る) (92 ページ)

- **例 1** – ユーザ `SQLTester` を変更します。パスワードは、`welcome` に設定されます。SQLTester ユーザは `Test1` ログインポリシーに割り当てられ、パスワードの有効期限は次回ログイン時に切れません。

```
ALTER USER SQLTester
IDENTIFIED BY welcome
LOGIN POLICY Test1
FORCE PASSWORD CHANGE OFF
```

- **例 2** – LDAP 認証で使用される、ユーザ Mary の識別名 (DN) およびタイムスタンプをクリアします。

```
ALTER USER Mary REFRESH DN
```

- **例 3** – user3 のパスワードを PassPart1PassPart2 に設定します。ここでは、user1 と user2 が CHANGE PASSWORD システム権限を持っており、user3 のログインポリシーで change_password_dual_control オプションが有効になっている (ON) ことを前提にしています。

user1 は次のように入力します。

```
ALTER USER user3 IDENTIFIED FIRST BY PassPart1
```

user2 は次のように入力します。

```
ALTER USER user3 IDENTIFIED LAST BY PassPart2
```

設定後に、user3 はパスワード PassPart1PassPart2 を入力してログオンします。

使用法

(先頭に戻る) (92 ページ)

ユーザ ID とパスワードで禁止されていることは次のとおりです。

- 最初の文字をスペース、一重引用符または二重引用符にする
- 最後の文字をスペースにする
- セミコロンを含める

パスワードは 255 文字を超えることはできません。

PASSWORD_EXPIRY_ON_NEXT_LOGIN 値を ON に設定すると、このログインポリシーに割り当てられたすべてのユーザのパスワードは、次回ログイン時にすぐに期限切れになります。次回ログイン時にユーザにパスワードの変更を強制するには、**ALTER USER** 句と **LOGIN POLICY** 句を使用します。

二重パスワード変更プロセス中に **CHANGE_PASSWORD_DUAL CONTROL** ログインポリシーオプションを無効 (OFF) にした場合は、次のようになります。

- ターゲットユーザは、すでに定義されている単一パスワード部分を使用してログインできなくなります。単一パスワード制御構文を使用して **ALTER USER** コマンドを再発行する必要があります。
- 二重パスワード変更プロセスが完了した後、ターゲットユーザがログインする前に、このオプションを無効にしても、ターゲットユーザには何の影響もありません。ターゲットユーザは、両方のパスワード部分を使用してログインする必要があります。

二重パスワード変更プロセスの実行時にユーザがすでにログインしている場合は、新しいパスワードの両方の部分が設定されるまで、現在のセッションでユーザが

自分のパスワードを変更することはできません。二重パスワード変更プロセスが完了すると、ターゲットユーザは **GRANT CONNECT**、**ALTER USER**、**sp_password**、または **sp_iqpassword** を使用して、ログアウトしてからでなくてもパスワードを変更できます。現在のパスワードの入力を要求されたら、現在のセッションで最初に入力したパスワードではなく、新しい二重制御パスワードを使用します。

二重パスワード変更プロセスでどちらかのパスワード部分を設定している間は、**GRANT CONNECT** 文はサポートされません。ただし、二重パスワード変更プロセスが完了すると、ターゲットユーザは **GRANT CONNECT** 文、**ALTER USER**、**sp_password**、または **sp_iqpassword** を使用して、ログアウトしてからでなくても自分のパスワードを変更できます。

CHANGE PASSWORD システム権限を持つユーザがパスワードの両方の部分を正常に指定すると同時に、ターゲットユーザのパスワードの有効期限が自動的に切れます。これによって、ターゲットユーザは次回ログイン時にパスワードの変更を強制されます。

ユーザパスワードのハッシュに使用される暗号化アルゴリズムは、FIPS 認定の暗号化サポートです。

- この DLL は **dbfips10.dll** と呼ばれます。
- HASH 関数ではアルゴリズム **SHA1_FIPS** および **SHA256_FIPS** を使用できます。
- **-fips** サーバオプションを指定したときに FIPS 認定でないアルゴリズムを HASH 関数に指定すると、データベースサーバでは **SHA1** の代わりに **SHA1_FIPS** が、**SHA256** の代わりに **SHA256_FIPS** が使用されます。また、**MD5** を使用した場合はエラーが返されます (**MD5** は FIPS 認定のアルゴリズムではありません)。
- **-fips** オプションを指定した場合は、データベースサーバではパスワードハッシュ処理に **SHA256_FIPS** が使用されます。

標準

(先頭に戻る) (92 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Adaptive Server ではサポートされていません。

パーミッション

(先頭に戻る) (92 ページ)

- 自分のパスワードを変更する場合は、権限は不要です。
- 任意のユーザのパスワードを変更する場合は、CHANGE PASSWORD システム権限が必要です。

- **LOGIN POLICY** 句、**FORCE PASSWORD CHANGE** 句、**RESET LOGIN POLICY** 句、または **REFRESH DN** 句を使用する場合は、**MANAGE ANY USER** システム権限が必要です。

参照：

- **COMMENT** 文 (123 ページ)
- **CREATE LOGIN POLICY** 文 (194 ページ)
- **CREATE USER** 文 (287 ページ)
- **DROP LOGIN POLICY** 文 (330 ページ)
- **DROP USER** 文 (346 ページ)
- **ALTER LOGIN POLICY** 文 (32 ページ)
- **GRANT ROLE** 文 (394 ページ)
- **GRANT** システム権限文 (402 ページ)
- **REVOKE** システム権限文 (512 ページ)
- **REVOKE ROLE** 文 (508 ページ)

ALTER VIEW 文

ビューの定義を修正したものに置き換えます。

クイックリンク：

「パラメータ」 (97 ページ)

「使用法」 (97 ページ)

「標準」 (98 ページ)

「パーミッション」 (98 ページ)

構文

構文 1 – ビューの構造を変更します。

ALTER VIEW

```
... [ owner. ] view-name [ ( column-name [ , ... ] ) ]  
... AS select-statement  
... [ WITH CHECK OPTION ]
```

構文 2 – ビューの属性を変更します。

ALTER VIEW

```
... [ owner. ] view-name  
... { SET HIDDEN | RECOMPILE | DISABLE | ENABLE }
```

パラメータ

(先頭に戻る) (96 ページ)

- **AS** – ビューが基づいている **SELECT** 文には、**ORDER BY** 句、**SELECT** リスト内のサブクエリ、および **TOP** または **FIRST** 修飾を含めないようにする必要があります。**GROUP BY** 句を含めたり、**UNION** にすることはできません。
- **WITH CHECK OPTION** – **SELECT** 文で定義されたビューの基準と一致しないビューへの更新や挿入がすべて拒否されます。ただし、現在は、SAP Sybase IQ でこのオプションが無視されます (互換性を考慮して構文はサポートされています)。
- **SET HIDDEN** – ビューと句の定義を難読化し、このビューを SAP Control Center などのビューから見えないようにします。ビューの明示的な参照は機能しません。

警告！ SET HIDDEN 操作は元に戻せません。

SET HIDDEN を使用すれば、ビューを別のデータベースにアンロードおよび再ロードできます。デバッガを使用したデバッグでは、ビューの定義は表示されず、プロシージャプロファイリングでも表示されません。非表示のビュー定義を変更するには、ビューを削除してから、**CREATE VIEW** 文を使用して再作成します。

- **RECOMPILE** – ビューのカラム定義を再作成します。機能的には **ENABLE** 句と同じですが、無効になっていないビューで使用できるという点で異なります。
- **DISABLE** – データベースサーバでビューを使用できなくします。

DISABLE 句を使用すると、データベースサーバではそのビューをクエリへの応答に使用できなくなります。ビューの無効化は、ビューの削除と同様ですが、ビュー定義がデータベースに残る点が異なります。ビューを無効にすると、従属ビューも無効になります。このため、**DISABLE** 句では、無効化対象のビューだけではなくすべての従属ビューに対する排他的アクセスが必要になります。また、これらの従属ビューも無効化されることになります。

- **ENABLE** – 無効化されたビューを有効化します。これによりデータベースサーバは、ビューのカラム定義を再作成します。ビューを有効化する際は、そのビューが使用するすべてのビューを有効化しておく必要があります。

使用法

(先頭に戻る) (96 ページ)

ビューを変更すると、ビューに対する既存のパーミッションはそのまま維持されます。このため、パーミッションの再割り当ては必要ありません。**ALTER VIEW** 文

を使用する代わりに、**DROP VIEW** を使用してビューを削除し、**CREATE VIEW** を使用して再作成することもできます。この処理方法を使用する場合は、ビューのパーミッションを割り当てなおす必要があります。

構文 1 を使用してビュー変更を完了した後は、データベースサーバでビューを再コンパイルします。実行した変更のタイプによっては、従属ビューが存在する場合にデータベースサーバがそれらのビューを再コンパイルしようとします。従属ビューに影響を及ぼす変更を加えると、従属ビューが無効になり、従属ビューの定義の変更が必要になることがあります。

警告！ ビューを定義する **SELECT** 文にアスタリスク (*) が付いており、基になるテーブルからカラムが追加または削除されている場合は、ビュー内のカラムの数が増える可能性があります。ビューカラムの名前とデータタイプも変更される場合があります。

ビューの構造を変更するには、**CREATE VIEW** 文を使ってビューを作成する場合と同様に、ビュー定義全体を新しい定義に置き換える必要があります。

関連する動作：

- オートコミット
- プロシージャとトリガはすべてメモリからアンロードされるため、ビューを参照するプロシージャとトリガは新しいビューの定義を反映します。定期的にビューを変更する場合は、プロシージャとトリガをアンロードしたり、ロードしたりすると、パフォーマンスが低下する可能性があります。

標準

(先頭に戻る) (96 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Adaptive Server ではサポートされていません。

パーミッション

(先頭に戻る) (96 ページ)

RECOMPILE 句または **ENABLE** 句 - ビューの場合は次のいずれかが必要です。

- ALTER ANY VIEW システム権限
- ALTER ANY OBJECT システム権限
- ビューを所有していること
- 次のいずれかも必要です。
 - SELECT ANY TABLE システム権限
 - ビューの基礎となるテーブルに対する SELECT 権限

マテリアライズドビューの場合は、次のいずれかが必要です。

- ALTER ANY MATERIALIZED VIEW システム権限
- ALTER ANY OBJECT システム権限
- マテリアライズドビューを所有していること
- 次のいずれかが必要です。
 - SELECT ANY TABLE システム権限
 - マテリアライズドビューの基礎となるテーブルに対する SELECT 権限

DISABLE 句 - ビューの場合は、次のいずれかが必要です。

- ALTER ANY VIEW システム権限
- ALTER ANY OBJECT システム権限
- ビューを所有していること

マテリアライズドビューの場合は、次のいずれかが必要です。

- ALTER ANY MATERIALIZED VIEW システム権限
- ALTER ANY OBJECT システム権限
- マテリアライズドビューを所有していること

それ以外の句はすべて、次のいずれかを必要とします。

- ALTER ANY OBJECT システム権限
- ビューを所有していること

参照：

- CREATE VIEW 文 (292 ページ)
- DROP 文 (319 ページ)
- 無効な従属ビューの特定と修正 (99 ページ)

無効な従属ビューの特定と修正

基本となるテーブルに対する変更によって無効になった従属ビューがないかどうかをチェックし、存在する場合には修正します。

基本となるテーブルが変更されると、通常は、データベースサーバはビューを自動的に再コンパイルしてビューの有効性を維持します。ただし、テーブルの変更により、ビュー定義によって参照されているものが削除されたり、大幅に変更されたりすると、従属ビューが無効になることがあります。たとえば、ビュー定義で参照されているカラムが削除されると、従属ビューは無効になります。ビュー定義を修正し、ビューを手動で再コンパイルします。

1. **sa_dependent_views** を実行して、従属ビューのリストを取得します。
2. テーブルを変更する DDL 操作を実行します。サーバは従属ビューを自動的に無効にし、DDL が完了したらそれらの再コンパイルを試行します。

3. **sa_dependent_views** によってリストされるすべてのビューが無効になっていることを確認します。たとえば、**SELECT * FROM myview** のような単純なテストを実行します。
4. ビューが無効な場合は、その問題を解決するためにビュー定義の変更が必要になることもあります。実行した DDL 変更に対するビュー定義を検証し、必要な変更を行います。**ALTER VIEW RECOMPILE** を実行して、ビュー定義を修正します。
5. 修正したビューをテストして、機能することを確認します。たとえば、**SELECT * FROM myview** のような単純なテストを実行します。

sa_dependent_views システムプロシージャは、特定のテーブルまたはビューのすべての従属ビューのリストを返します。

参照：

- ALTER VIEW 文 (96 ページ)

BACKUP DATABASE 文

1 つまたは複数のアーカイブデバイスに SAP Sybase IQ データベースをバックアップします。

クイックリンク：

「パラメータ」 (101 ページ)

「例」 (104 ページ)

「使用法」 (104 ページ)

「標準」 (107 ページ)

「パーミッション」 (107 ページ)

構文

```
BACKUP DATABASE
  [ backup-option ... ]
  TO archive_device [ archive-option... ]
  ... [ WITH COMMENT string ]
```

```
backup-option - (構文に戻る)
  { READWRITE FILES ONLY |
  READONLY dbspace-or-file [, ... ] }
  CRC { ON | OFF }
  ATTENDED { ON | OFF }
  BLOCK FACTOR integer
```



```

{ FULL | INCREMENTAL | INCREMENTAL SINCE FULL }
VIRTUAL { DECOUPLED |
ENCAPSULATED 'shell_command' }
WITH COMMENT comment

dbspace-or-file - (back to backup-option)
{ DBSPACES identifier-list | FILES identifier-list }

identifier-list - (back to dbspace-or-file)
identifier [, ... ]

archive-option - (構文に戻る)
SIZE integer STACKER integer

```

パラメータ

(先頭に戻る) (100 ページ)

- **TO** – バックアップで使用する `archive_device` の名前を一重引用符で区切って指定します。 `archive_device` はアーカイブファイル用のファイル名またはテープドライブデバイス名です。複数のアーカイブデバイスを使用する場合、それぞれに別々の **TO** 句を使用して指定します(カンマで区切ったリストは使用できません)。アーカイブデバイスは異なるものでなければなりません。出力デバイスに関して SAP Sybase IQ が試みる並行処理の量は、**TO** 句の数によって決まります。
- **WITH COMMENT** – アーカイブファイルとバックアップ履歴ファイルに記録するコメント(オプション)を指定します。最大長は 32KB です。値を指定しなければ、NULL 文字列が格納されます。
- **READWRITE FILES ONLY** – **FULL**、**INCREMENTAL**、および **INCREMENTAL SINCE FULL** バックアップをデータベース内の読み込み/書き込みファイルのみに制限します。読み込み/書き込み DB 領域/ファイルは SAP Sybase IQ DB 領域である必要があります。

INCREMENTAL または **INCREMENTAL SINCE FULL** バックアップで **READWRITE FILES ONLY** 句を指定すると、依存バックアップ以降に変更された読み込み専用 DB 領域または DB ファイルのデータはバックアップされません。**INCREMENTAL** または **INCREMENTAL SINCE FULL** バックアップで **READWRITE FILES ONLY** が指定されていない場合は、読み/書きおよび読み込み専用 DB 領域の両方の、依存バックアップ以降に変更されたすべてのデータベースページがバックアップされます。

- **CRC** – ブロックごとの 32 ビット周期の冗長性検査をアクティブにします(ハードウェアで使用可能なエラー検出に加えて)。この句を指定すると、その後の

リストア操作でバックアップで計算された数字が検証されるため、両方のコマンドのパフォーマンスに影響を及ぼします。デフォルトは ON です。

- **ATTENDED** – この句はテープデバイスにバックアップする場合にのみ使用します。ATTENDED ON 句 (デフォルト値) を指定した場合、テープドライブが介入を必要とするときに、**BACKUP DATABASE** 文を発行したアプリケーションにメッセージが送信されます。たとえば、新しいテープが必要になった場合です。OFF を指定すると、**BACKUP DATABASE** は新しいテープを要求するメッセージを表示しません。追加のテープが必要で、OFF が指定されている場合、SAP Sybase IQ はエラーを送信し **BACKUP DATABASE** コマンドをアボートします。ただし、自動スタックドライブがテープを切り替えるための時間を考慮して、短い遅延が含まれています。
- **BLOCK FACTOR integer** – 同時に書き込むブロック数を指定します。値は 0 より大きくしてください。そうしないと、SAP Sybase IQ からエラーメッセージが生成されます。デフォルトは、UNIX システムの場合は 25、Windows システムの場合は 15 です (小さいサイズの固定テープブロックを使用するため)。この句はバッファが使用するメモリを効率よく制御します。実際のメモリ量は、この値にブロックサイズと、データベースからのデータの抽出に使用するスレッド数を掛けたものです。BLOCK FACTOR を少なくとも 25 に設定します。
- **FULL | INCREMENTAL | INCREMENTAL SINCE FULL** –
 - **FULL** – フルバックアップを指定します。データベースで使用中の全ブロックをアーカイブデバイスに保存します。これがデフォルトです。
 - **INCREMENTAL** – インクリメンタルバックアップを指定します。最後に行われたいずれかの種類のバックアップ以降に変更された全ブロックをアーカイブデバイスに保存します。キーワード **INCREMENTAL** は、**READONLY FILES** では使用できません。
 - **INCREMENTAL SINCE FULL** – インクリメンタルバックアップを指定します。最後のフルバックアップ以降に変更された全ブロックをアーカイブデバイスに保存します。
- **VIRTUAL DECOUPLED** – 分離仮想バックアップを指定します。バックアップを完了させるには、分離仮想バックアップが終了した後に SAP Sybase IQ の DB 領域をコピーし、非仮想のインクリメンタルバックアップを実行する必要があります。
- **VIRTUAL ENCAPSULATED** – カプセル化仮想バックアップを指定します。'shell-command' 引数には、カプセル化仮想バックアップの中で実行されるシェルコマンドを、文字列または文字列を含む変数で指定します。シェルコマンドは、バックアップ処理の中で、IQ ストアのシステムレベルのバックアップを実行します。セキュリティ上の理由から、'shell-command' には絶対パスを指定し、そのディレクトリにファイル保護を設定して、不正なプログラムの実行を防止することをおすすめします。

- **SIZE** – 出力デバイスごとのテープまたはファイルの最大容量を指定します (テープの終了マークを正確に検出できないプラットフォームもあります)。対応するデバイスで使用するボリュームは、この値より小さくしないでください。この値は、テープとディスクファイルの両方に適用されますが、サードパーティのデバイスには適用されません。単位はキロバイト (KB) です。ただし、ほとんどの場合、1GB 未満は不適切です。たとえば、3.5GB テープの場合は 3500000 を指定します。デフォルトはプラットフォームと媒体によって異なります。バックアップファイルの最終的なサイズは正確ではありません。これは、バックアップでは、大きなデータブロック単位で書き込まれるためです。

表 1 : BACKUP DATABASE のデフォルトサイズ

プラット フォーム	SIZE のデフォルト (テープ)	SIZE のデフォ ルト (ディスク)
UNIX	なし	2GB
Windows	1.5GB SIZE には 64 の倍数を指定する。それ以外の値 は、64 の倍数まで切り捨てられる。	1.5GB

SIZE パラメータは、出力デバイスごとに指定します。SIZE はデバイスごとのバイト数を制限するのではなく、ファイルサイズを制限します。出力デバイスごとに異なる SIZE パラメータを指定できます。バックアップ中、指定されたデバイスに書き込まれる情報量が SIZE パラメータに指定された値に達した場合、**BACKUP DATABASE** は次のいずれかの方法で処理します。

- ファイルシステムデバイスの場合、**BACKUP DATABASE** は現在のファイルを閉じて別のファイルを同じ名前で作成します。このファイル名には、たとえば bkup1.dat1.1、bkup1.dat1.2、bkup1.dat1.3 のように、昇順で番号が付加されます。
- テープユニットデバイスの場合、**BACKUP DATABASE** は現在のテープを閉じます。このとき、ユーザは別のテープをマウントする必要があります。
- **STACKER** – デバイスの自動的なロードを指定し、ロードされるテープの数を指定します。この値は、スタッカのテープの位置ではありません。テープの位置は 0 になります。ATTENDED が OFF で STACKER が ON の場合、SAP Sybase IQ は、次のテープを自動ロードできるようにするため、事前定義された時間だけ待機します。SIZE 句を使用して指定されたテープ数は、バックアップデータを格納する十分な領域があるかどうかを判定するために使用されます。この句は、サードパーティ製のメディア管理デバイスには使用しないでください。

例

(先頭に戻る) (100 ページ)

- **例 1** – この UNIX の例は、iqdemo データベースを、Sun Solaris プラットフォーム上のテープデバイス /dev/rmt/0 と /dev/rmt/2 にバックアップします。Solaris の場合は、デバイス名に続けて n の文字を入力すると、"no rewind on close" 機能が指定されます。**BACKUP DATABASE** では、お使いの UNIX プラットフォームに適した命名規則を使用して、必ずこの機能を指定してください (Windows はこの機能をサポートしていません)。次の例では、前回のフルバックアップ以降にデータベースに加えられた全変更をバックアップします。

```
BACKUP DATABASE
INCREMENTAL SINCE FULL
TO '/dev/rmt/0n' SIZE 10000000
TO '/dev/rmt/2n' SIZE 15000000
```

注意：サイズの単位はキロバイト (KB) です。ただし、ほとんどの場合、1GB 未満のサイズは不適切です。この例では、指定サイズは 10GB と 15GB です。

- **例 2** – 次の **BACKUP DATABASE** コマンドは、読み込み専用ファイルと DB 領域を指定します。

```
BACKUP DATABASE READONLY DBSPACES dsp1
TO '/dev/rmt/0'
```

```
BACKUP DATABASE READONLY FILES dsp1_f1, dsp1_f2
TO 'bkp.flf2'
```

```
BACKUP DATABASE READONLY DBSPACES dsp2, dsp3
READONLY FILES dsp4_f1, dsp5_f2
TO 'bkp.RO'
```

使用法

(先頭に戻る) (100 ページ)

BACKUP DATABASE コマンドの実行時に、多くのリーダーとライターが SAP Sybase IQ データベースを開いて使用している場合があります。その場合、このコマンドは読み込み専用ユーザとして機能し、SAP Sybase IQ のテーブルレベルのバージョン管理機能を使用してデータの一貫性を維持します。

BACKUP DATABASE は開始の前に **CHECKPOINT** を暗黙的に発行し、その後、データベースを記述したカタログテーブル (およびカタログストアに追加したすべてのテーブル) をバックアップします。この最初のフェーズ中、SAP Sybase IQ ではデータベースのメタデータの変更 (カラムやテーブルの追加や削除など) が許可されません。したがって、後でバックアップの **RESTORE DATABASE** を行っても、最初の **CHECKPOINT** までしかリストアできません。

BACKUP DATABASE コマンドでは、フルバックアップまたはインクリメンタルバックアップの指定が可能です。インクリメンタルバックアップには、2 種類の

選択肢があります。INCREMENTAL では、いずれかのタイプによる最後のバックアップ (インクリメンタルまたはフル) 以降に変更またはコミットされたブロックのみをバックアップします。INCREMENTAL SINCE FULL では、最後のフルバックアップ以降に変更されたブロックのすべてがバックアップされます。最初のタイプのインクリメンタルバックアップは、**BACKUP DATABASE** コマンドのバックアップ量が小さく実行速度が速くなりますが、**RESTORE DATABASE** コマンドの実行速度が遅く複雑になります。もう一方のタイプのインクリメンタルバックアップは、その反対になります。その理由は、前者のタイプからは通常、フルバックアップアーカイブごとに N セットのインクリメンタルバックアップアーカイブが生成されるためです。リストアが必要な場合は、SERVER OPERATOR システム権限を持つユーザが最初にフルバックアップアーカイブをリストアしてから、各インクリメンタルアーカイブを正しい順序でリストアする必要があります (どれが必要になるかは、SAP Sybase IQ に記録されます)。2 番目のタイプでは、SERVER OPERATOR システム権限を持つユーザがフルバックアップアーカイブと最後のインクリメンタルアーカイブをリストアするだけで済みます。

インクリメンタル仮想バックアップは、**BACKUP DATABASE** 文の VIRTUAL DECOUPLED パラメータと VIRTUAL ENCAPSULATED パラメータを使用してサポートされます。

テーブルスペースの OS レベルコピーを実行して、1 つまたは複数の読み込み専用 DB 領域の仮想バックアップを行うことができますが、仮想バックアップは SAP Sybase IQ システムテーブル内にバックアップを記録するので、仮想バックアップ文を使用することをおすすめします。

BACKUP DATABASE と **RESTORE DATABASE** によって、指定したすべてのアーカイブデバイスからの SAP Sybase IQ データの取得とそれらへの書き込みが並行して行われます。その後、カタログストアが順次最初のデバイスに書き込まれます。優れた並行処理機能によって、バックアップ時間とリストア時間が短縮されます。

SAP Sybase IQ では、最大 36 のハードウェアデバイスのバックアップがサポートされています。バックアップ時間を短縮するには、コアごとに 1 つまたは 2 つのデバイスを指定して、ハードウェアおよび IO の競合を回避します。**BACKUP DATABASE** コマンドに SIZE パラメータを設定して各バックアップデバイスに複数のファイルが作成されないようにして、**BACKUP DATABASE** コマンドの BLOCK FACTOR 句に使用する値を検討します。

古いファイルを移動するか、*archive_device* 名やパスに以前と異なるものを指定しないかぎり、**BACKUP DATABASE** は既存のアーカイブファイルを上書きします。

バックアップ API の DLL 版では、アーカイブデバイスのオープン時に DLL に渡す引数を指定できます。サードパーティ実装の場合、*archive_device* 文字列は次のフォーマットになります。

```
'DLLIdentifier::vendor_specific_information'
```

具体例：

```
'spsc::workorder=12;volname=ASD002'
```

archive_device 文字列の最大長は 1023 バイトです。*DLLIdentifier* 部分は 1 から 30 バイトの長さで、英数字とアンダースコアの文字だけを使用できます。文字列の *vendor_specific_information* 部分は、内容の検査なしでサードパーティ実装に渡されます。サードパーティ実装を使用している場合、**BACKUP DATABASE** コマンドの *SIZE* 句または *STACKER* 句は指定しないでください。この情報は文字列の *vendor_specific_information* 部分にコード化する必要があるためです。

注意：この構文を SAP Sybase IQ で使用して動作確認されているサードパーティ製品は、ごく一部に限られます。使用上のその他の指示や制約については、『リリースノート』を参照してください。サードパーティ製品をこの方法で使用して SAP Sybase IQ データベースをバックアップする場合は、その製品が動作確認されたものであるかどうかを事前に確認してください。『リリースノート』、または Technical Documents at <http://www.sybase.com/support/techdocs/> で提供されている SAP Sybase IQ 製品に対する SAP Sybase Certification Reports を参照してください。

バックアップ API の SAP Sybase IQ 版の場合、テープデバイス名またはファイル名以外の情報を指定する必要はありません。ディスクデバイスの場合は、*SIZE* 値も指定してください。指定がない場合、SAP Sybase IQ は作成される各ディスクファイルが 2GB 未満 (UNIX の場合) または 1.5GB 未満 (Windows の場合) であると想定します。

特定の UNIX システムのテープデバイスを指定する SAP Sybase API DLL のアーカイブデバイスの例を示します。

```
'/dev/rmt/0'
```

追加のテープが必要になった場合のマウント処理、および、バックアップに使用できるディスク上の領域の確保はユーザの責任に行う必要があります。

複数のデバイスを指定すると、**BACKUP DATABASE** は情報をすべてのデバイスに配布します。**BACKUP DATABASE** では、他にも以下の点に注意してください。

- **BACKUP DATABASE** ではローデバイスをアーカイブデバイスとして使用することはできません。
- Windows システムでは、テープデバイスに対する固定長の I/O 操作のみがサポートされます (この制限の詳細については、『インストールおよび設定ガイド』を参照してください)。Windows はテープパーティションをサポートしていますが、SAP Sybase IQ はテープパーティションを使用しません。このため、別のアプリケーションを使用して **BACKUP DATABASE** 用のテープをフォーマットしないでください。Windows のテープデバイスの命名方式は非常に簡単で

す。1 番目のテープデバイスは `¥¥.¥tape0`、2 番目は `¥¥.¥tape1` というように命名されます。

警告! バックアップ(およびその他のほとんどの状況)では、文字列中に出現する `¥n`、`¥x` または `¥¥` のうちの最初の `¥` は、SAP Sybase IQ によりエスケープ文字として処理されます。このためバックアップ用のテープデバイスを指定する場合は、Windows の命名規則に従って必要な場所に `¥` を 2 つずつ使用しなければなりません。たとえば、バックアップに使用する 1 番目の Windows テープデバイスを `'¥¥¥¥.¥¥tape0'`、2 番目を `'¥¥¥¥.¥¥tape1'` というようにします。追加すべき `¥` を省略したり、テープデバイス名の入力を間違えたり、システム上の有効テープデバイスではない名前を入力したりすると、SAP Sybase IQ はそれをディスクファイル名と解釈します。

- SAP Sybase IQ は、テープを使用する前にリワインドしません。バックアップやリストアに使用するテープは、テープデバイスに挿入する前に、必ず正しい開始地点まで巻かれていることを確認してください。ただし、リワインディングデバイスでテープを使用している場合は、SAP Sybase IQ がリワインドを実行します。
- バックアップやリストアの操作中に、アーカイブデバイスを開くことができず(メディアのロードが必要な場合など)、ATTENDED 句が ON の場合、SAP Sybase IQ は 10 秒間待機してから再試行します。デバイスが正常に挿入されるか、[Ctrl + C] キーを押して操作を終了しないかぎり、無制限に再試行されます。
- [Ctrl] キーを押しながら [C] キーを押すと、**BACKUP DATABASE** は失敗し、データベースはバックアップ開始前の状態に戻ります。
- RAID デバイスなどでディスクストライピングを使用している場合は、ストライプしたディスクは単一のデバイスとして処理されます。

関連する動作：

- オートコミット

標準

(先頭に戻る) (100 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - SAP Sybase Adaptive Server® Enterprise ではサポートされていません。

パーミッション

(先頭に戻る) (100 ページ)

次のいずれかが必要です。

SQL 文

- BACKUP DATABASE システム権限
- そのデータベースを所有していること

参照：

- RESTORE DATABASE 文 (487 ページ)

BEGIN ... END 文

SQL 文をグループ化します。

クイックリンク：

「パラメータ」 (109 ページ)

「例」 (109 ページ)

「使用法」 (110 ページ)

「標準」 (110 ページ)

「パーミッション」 (111 ページ)

構文

```
[ statement-label : ]  
... BEGIN [ [ NOT ] ATOMIC ]  
... [ local-declaration ; ... ]  
... statement-list  
... [ EXCEPTION [ exception-case ... ] ]  
... END [ statement-label ]
```

local-declaration - (構文に戻る)

```
{ variable-declaration  
| cursor-declaration  
| exception-declaration  
| temporary-table-declaration }
```

variable-declaration - (local-declaration に戻る)

```
DECLARE variable-name [ , ... ] data-type  
[ { = | DEFAULT } initial-value ]
```

initial-value - (variable-declaration に戻る)

```
special-value  
| string  
| [ - ] number  
| ( constant-expression )  
| built-in-function ( constant-expression )  
| NULL
```

special-value - (initial-value に戻る)

```
CURRENT {
```



```

DATABASE
| DATE
| PUBLISHER
| TIME
| TIMESTAMP
| USER
| UTC TIMESTAMP }
| USER

```

パラメータ

(先頭に戻る) (108 ページ)

- **statement-label** – 指定された場合は、最初の *statement-label* と一致する必要があります。 **LEAVE** 文を使用すると、複合文の後にある最初の文から実行を再開できます。プロシージャの本文である複合文は、プロシージャの名前と同じ暗黙のラベルを持っています。
- **initial-value** – 指定された場合は、変数とその値に設定され、データ型が *data-type* で定義された型と一致する必要があります。 *initial-value* が指定されなかった場合は、変数に NULL 値が設定されます。この値は、**SET** 文によって別の値が割り当てられるまで変わりません。

例

(先頭に戻る) (108 ページ)

- **例 1** – プロシージャの本文は複合文です。

```

CREATE PROCEDURE TopCustomer (OUT TopCompany CHAR(35), OUT
TopValue INT)
BEGIN
  DECLARE err_notfound EXCEPTION FOR
    SQLSTATE '02000' ;
  DECLARE curThisCust CURSOR FOR
    SELECT CompanyName, CAST(
      sum(SalesOrderItems.Quantity *
        Products.UnitPrice) AS INTEGER) VALUE
    FROM Customers
      LEFT OUTER JOIN Salesorders
      LEFT OUTER JOIN SalesOrderItems
      LEFT OUTER JOIN Products
      GROUP BY CompanyName ;
  DECLARE ThisValue INT ;
  DECLARE ThisCompany CHAR(35) ;
  SET TopValue = 0 ;
  OPEN curThisCust ;

  CustomerLoop:
  LOOP
    FETCH NEXT curThisCust
      INTO ThisCompany, ThisValue ;

```

```

IF SQLSTATE = err_notfound THEN
    LEAVE CustomerLoop ;
END IF ;
IF ThisValue > TopValue THEN
    SET TopValue = ThisValue ;
    SET TopCompany = ThisCompany ;
END IF ;
END LOOP CustomerLoop ;

CLOSE curThisCust ;
END

```

使用法

(先頭に戻る) (108 ページ)

プロシージャの本体は複合文です。複合文は、プロシージャ内の制御文の中でも使用できます。

複合文を使用すると、1つ以上のSQL文をまとめて、1つの単位として扱うことができます。複合文は **BEGIN** で始まり、**END** で終わります。**BEGIN** の直後に、複合文はその中だけで有効なローカル宣言を行うことができます。複合文は、変数、カーソル、テンポラリテーブル、または例外に対してローカル宣言を行うことができます。ローカル宣言は、複合文またはその中でネストされる複合文の中のどの文からでも参照できます。ローカル宣言は、複合文の中から呼び出される他のプロシージャには作用しません。

「アトミック」な文とは、完全に実行されたか、まったく実行されなかったかのいずれかの文です。たとえば、何千ものローを更新する **UPDATE** 文では、たくさんのローを更新した後にエラーが発生することがあります。文が完了しないと、すべての変更内容が元の状態に戻ります。同様に、**BEGIN** 文をアトミックに指定すると、文は完全に実行されるか、まったく実行されないかのいずれかになります。

標準

(先頭に戻る) (108 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Adaptive Server でサポートされています。複合文中のすべての文がサポートされているわけではありません。

BEGIN キーワードと **END** キーワードは、Transact-SQL では必要ありません。

BEGIN と **END** を Transact-SQL で使用して、文のグループを1つの複合文にまとめることができます。これによって **IF ... ELSE** など、単一のSQL文の動作だけに影響する制御文を複合文にまとめて、グループ全体の動作を制御できます。Adaptive Server では、**ATOMIC** キーワードはサポートされません。

Transact-SQL では、**DECLARE** 文を **BEGIN** の直後に続ける必要はなく、宣言したカーソルや変数は複合文の存続期間だけ存在します。互換性を保つために、複合文の冒頭で変数を宣言してください。

パーミッション

(先頭に戻る) (108 ページ)

なし

参照：

- **DECLARE LOCAL TEMPORARY TABLE** 文 (306 ページ)
- **DECLARE CURSOR** 文 [ESQL] [SP] (299 ページ)
- **LEAVE** 文 (428 ページ)
- **RESIGNAL** 文 (486 ページ)
- **SIGNAL** 文 (550 ページ)

BEGIN PARALLEL IQ ... END PARALLEL IQ 文

CREATE INDEX 文をグループ化して、同時に実行できるようにします。

クイックリンク：

「パラメータ」 (111 ページ)

「例」 (111 ページ)

「使用法」 (112 ページ)

「標準」 (112 ページ)

「パーミッション」 (112 ページ)

構文

```
... BEGIN PARALLEL IQ statement-list  
... END PARALLEL IQ
```

パラメータ

(先頭に戻る) (111 ページ)

- **statement-list** – **CREATE INDEX** 文のリスト

例

(先頭に戻る) (111 ページ)

SQL 文

- **例1**–次の文はアトミックに実行されます。1つのコマンドが失敗すると、文全体がロールバックします。

```
BEGIN PARALLEL IQ
  CREATE HG INDEX c1_HG on table1 (col1);
  CREATE HNG INDEX c12_HNG on table1 (col12);
  CREATE LF INDEX c1_LF on table1 (col1);
  CREATE HNG INDEX c2_HNG on table1 (col2);
END PARALLEL IQ
```

使用法

(先頭に戻る) (111 ページ)

BEGIN PARALLEL IQ ... END PARALLEL IQ 文により、単一の DDL 文であるかのように **CREATE INDEX** 文のグループを実行し、複数の IQ テーブルのインデックスを同時に作成できます。この文の実行中、ユーザは他の DDL 文を発行できません。

注意：

- 次の文は RLV 対応テーブルをサポートしません。
- 次の文は TEXT インデックスをサポートしません。

関連する動作：

- オートコミット

標準

(先頭に戻る) (111 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Adaptive Server ではサポートされていません。この文に含まれる文のサポートについては、「**CREATE INDEX** 文」を参照してください。

パーミッション

(先頭に戻る) (111 ページ)

なし

参照：

- **CREATE INDEX** 文 (179 ページ)

BEGIN TRANSACTION 文 [T-SQL]

この文を使用して、ユーザ定義のトランザクションを開始します。

注意： **BEGIN TRANSACTION** は T-SQL 構成体であるため、有効な T-SQL コマンドのみを含める必要があります。T-SQL コマンドと T-SQL 以外のコマンドを組み合わせることはできません。

クイックリンク：

「例」 (113 ページ)

「使用法」 (114 ページ)

「標準」 (115 ページ)

「パーミッション」 (115 ページ)

構文

```
BEGIN TRAN [SACTION] [ transaction-name ]
```

例

(先頭に戻る) (113 ページ)

- **例 1** – **@@trancount** の連続値を 0、1、2、1、0 としてレポートし、値をサーバーウィンドウに出力します。

```
PRINT @@trancount
BEGIN TRANSACTION
PRINT @@trancount
BEGIN TRANSACTION
PRINT @@trancount
COMMIT TRANSACTION
PRINT @@trancount
COMMIT TRANSACTION
PRINT @@trancount
```

@@trancount の値は、発行された明示的な **BEGIN TRANSACTION** 文の数の追跡以外の目的には使用しないでください。

Adaptive Server が暗黙的にトランザクションを起動する場合、**@@trancount** 変数を 1 に設定します。トランザクションが暗黙的に起動された場合、SAP Sybase IQ は **@@trancount** 値を 1 に設定しません。そのため、**BEGIN TRANSACTION** 文が始まる前は (現在のトランザクションがあるときでも)、SAP Sybase IQ の **@@trancount** 変数の値はゼロになり、Adaptive Server (連鎖モード時) の **@@trancount** の値は 1 になります。

BEGIN TRANSACTION 文で開始するトランザクションでは、最初の **BEGIN TRANSACTION** 文の後、`@@trancount` の値は SAP Sybase IQ と Adaptive Server の両方で 1 になります。異なる文でトランザクションが暗黙的に起動し、その後、**BEGIN TRANSACTION** 文が実行された場合、**BEGIN TRANSACTION** 文の後、`@@trancount` の値は、SAP Sybase IQ と Adaptive Server の両方で 2 になります。

使用法

(先頭に戻る) (113 ページ)

オプションのパラメータ *transaction-name* はこのトランザクションに割り当てられた名前です。有効な識別子にする必要があります。トランザクション名は、ネストされた **BEGIN/COMMIT** 文または **BEGIN/ROLLBACK** 文の最も外側のペアでのみ使用してください。

BEGIN TRANSACTION 文をトランザクション内で実行すると、トランザクションのネストレベルが 1 つ増加します。ネストレベルは **COMMIT** 文で減少します。トランザクションがネストされているときは、最も外側の **COMMIT** だけがデータベースへの変更を永続化します。

Adaptive Server と SAP Sybase IQ の両方に 2 つのトランザクションモードがあります。

デフォルトの Adaptive Server トランザクションモードは非連鎖モードと呼ばれ、明示的な **BEGIN TRANSACTION** 文が実行されてトランザクションを起動しないかぎり、各文を個々にコミットします。反対に、ISO SQL/2003 互換の連鎖モードは、明示的 **COMMIT** の実行時か、オートコミットする文 (データ定義文など) の実行時のみトランザクションをコミットします。

chained データベースオプションを設定してモードを制御できます。SAP Sybase IQ の ODBC 接続と Embedded SQL 接続のデフォルト設定が ON の場合、SAP Sybase IQ は連鎖モードで実行されます (ODBC ユーザは、AutoCommit ODBC 設定も確認する必要があります)。TDS 接続のデフォルトは Off です。

非連鎖モードでは、データ取得文か修正文の前にトランザクションが暗黙のうちに開始されます。これらの文には、次のようなものがあります。**DELETE**、**INSERT**、**OPEN**、**FETCH**、**SELECT**、および **UPDATE**。この場合も、トランザクションは **COMMIT** または **ROLLBACK** 文を使用して明示的に終了する必要があります。

トランザクション内で chained オプションを変更することはできません。

注意： ストアドプロシージャを呼び出すときは、目的のトランザクションモードで正しく動作することを確認してください。

現在のネストレベルはグローバル変数 `@@trancount` に格納されます。`@@trancount` 変数の値は、最初の **BEGIN TRANSACTION** 文が実行される前は 0 であり、

@@trancount の値が 1 のときに実行された **COMMIT** だけがデータベースへの変更を永続化できます。

トランザクションまたはセーブポイント名のない **ROLLBACK** 文は、常に文を最も外側の **BEGIN TRANSACTION** (明示的または暗黙的) 文にロールバックし、トランザクション全体をキャンセルします。

標準

(先頭に戻る) (113 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Adaptive Server でサポートされています。

パーミッション

(先頭に戻る) (113 ページ)

なし

参照:

- COMMIT 文 (129 ページ)
- ROLLBACK TRANSACTION 文 [T-SQL] (520 ページ)
- SAVE TRANSACTION 文 [T-SQL] (522 ページ)
- ISOLATION_LEVEL オプション (695 ページ)

CALL 文

プロシージャを起動します。

クイックリンク:

「例」 (116 ページ)

「使用法」 (116 ページ)

「標準」 (117 ページ)

「パーミッション」 (117 ページ)

構文

構文 1

```
[ variable = ] CALL procedure-name ( [ expression ] [ , ... ] )
```

構文 2

```
[ variable = ] CALL procedure-name ( [ parameter-name = expression ]
[ , ... ] )
```

例

(先頭に戻る) (115 ページ)

- **例 1** – sp_customer_list プロシージャを呼び出します。このプロシージャはパラメータがなく、結果セットを返します。

```
CALL sp_customer_list()
```

- **例 2** – 指定した ID を持つ顧客からの注文の数を返すプロシージャを作成し、結果を格納する変数を作成し、プロシージャを呼び出して、結果を表示します。

```
CREATE PROCEDURE OrderCount (IN CustomerID INT, OUT Orders INT)
BEGIN
SELECT COUNT("DBA".SalesOrders.ID)
INTO Orders
FROM "DBA".Customers
KEY LEFT OUTER JOIN "DBA".SalesOrders
WHERE "DBA".Customers.ID = CustomerID ;
END
go
-- Create a variable to hold the result
CREATE VARIABLE Orders INT
go

-- Call the procedure, FOR customer 101
-----
CALL OrderCount ( 101, Orders)
go
-----
-- Display the result
SELECT Orders FROM DUMMY
go
```

使用法

(先頭に戻る) (115 ページ)

CALL は、あらかじめ **CREATE PROCEDURE** 文を使用して作成されたプロシージャを呼び出します。プロシージャが完了すると、INOUT または OUT パラメータ値がコピーし直されます。

引数リストは、引数の位置によって、またはキーワードフォーマットを使用して指定できます。位置で指定する場合、引数はプロシージャのパラメータリスト内の対応するパラメータと一致します。キーワードでは、引数は指定したパラメータと一致します。

プロシージャ引数には、**CREATE PROCEDURE** 文のデフォルト値を割り当てることができます。パラメータが見つからない場合は、デフォルト値が割り当てられま

す。また、デフォルトが設定されていない場合、パラメータが NULL に設定されます。

プロシージャの内部では、プロシージャが結果セットを返すときに **DECLARE** 文で **CALL** 文を使用できます。

注意： **CALL SQL** 文では、テーブル UDF は参照できません。

プロシージャは **RETURN** 文を使って整数値 (ステータスインジケータとして) を返すことができます。この戻り値を変数に保存するには、代入演算子として等号を使います。

```
CREATE VARIABLE returnval INT ;
returnval = CALL proc_integer ( arg1 = val1, ... )
```

注意： この文を使用した関数の呼び出しは廃止される予定です。関数を呼び出すには、代入文を使用して関数を呼び出し、その結果を変数に代入します。次に例を示します。

```
DECLARE varname INT;
SET varname=test( );
```

標準

(先頭に戻る) (115 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Adaptive Server ではサポートされていません。サポートされている代替機能については、「EXECUTE 文 [ESQL]」を参照してください。

パーミッション

(先頭に戻る) (115 ページ)

次のいずれかが必要です。

- EXECUTE ANY PROCEDURE システム権限
- プロシージャに対する EXECUTE パーミッション
- プロシージャを所有していること

参照：

- CREATE PROCEDURE 文 (210 ページ)
- EXECUTE 文 [ESQL] (348 ページ)
- GRANT EXECUTE 文 (391 ページ)

CASE 文

CASE 文は制御文で、これを使用すると式の値に従って実行する SQL 文のリストを選択できます。

クイックリンク：

「例」 (118 ページ)

「使用法」 (119 ページ)

「標準」 (119 ページ)

「パーミッション」 (119 ページ)

構文

```
CASE value-expression
...WHEN [ constant | NULL ] THEN statement-list ...
... [ WHEN [ constant | NULL ] THEN statement-list ] ...
...ELSE statement-list
... END
```

例

(先頭に戻る) (118 ページ)

- **例 1** – デモデータベースの Products テーブルにリストされている製品を、シャツ、帽子、ショートパンツ、不明のいずれかに分類します。

```
CREATE PROCEDURE ProductType (IN product_id INT, OUT type
CHAR(10))
BEGIN
  DECLARE prod_name CHAR(20) ;
  SELECT name INTO prod_name FROM "GROUPO"."Products"
  WHERE ID = product_id;
  CASE prod_name
  WHEN 'Tee Shirt' THEN
    SET type = 'Shirt'
  WHEN 'Sweatshirt' THEN
    SET type = 'Shirt'
  WHEN 'Baseball Cap' THEN
    SET type = 'Hat'
  WHEN 'Visor' THEN
    SET type = 'Hat'
  WHEN 'Shorts' THEN
    SET type = 'Shorts'
  ELSE
    SET type = 'UNKNOWN'
  END CASE ;
END
```

使用法

(先頭に戻る) (118 ページ)

value-expression の値に対して WHEN 句が存在する場合、WHEN 句の *statement-list* が実行されます。適切な WHEN 句が存在せず、ELSE 句が存在する場合、ELSE 句の *statement-list* が実行されます。END の後に記述されている最初の文から実行が再開されます。

注意： ANSI 標準には、2つの形式で **CASE** 文を使用できます。SAP Sybase IQ も両方の形式に対応していますが、**CASE** を述部内で使用する場合に、最適なパフォーマンスを得るにはここに示されている形式を使用してください。

SQL Anywhere との互換性保持のため他の形式 (ANSI 構文とも呼ばれる) が必要な場合は、次の構文を使用します。

```
CASE
  WHEN [ search-condition | NULL] THEN statement-list ...
  [ WHEN [ search-condition | NULL] THEN statement-list ] ...
  [ ELSE statement-list ]
END [ CASE ]
```

この ANSI 構文形式を使用した場合は、**CASE** 文内で最初に一致した *search-condition* に対して文が実行されます。どの *search-conditions* も一致しなかった場合は、ELSE 句が実行されます。NULL が許容される式の場合は、最初の *search-condition* に次の構文を使用します。

```
WHEN search-condition IS NULL THEN statement-list
```

注意： **CASE** 文の構文と CASE 式の構文を混同しないでください。

標準

(先頭に戻る) (118 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Adaptive Server ではサポートされていません。

パーミッション

(先頭に戻る) (118 ページ)

なし

参照：

- BEGIN ... END 文 (108 ページ)

CHECKPOINT 文

データベースのチェックポイントを実行します。

クイックリンク：

「使用法」 (120 ページ)

「標準」 (120 ページ)

「パーミッション」 (121 ページ)

構文

CHECKPOINT

使用法

(先頭に戻る) (120 ページ)

CHECKPOINT は、データベースサーバにチェックポイントの実行を強制します。またチェックポイントは、内部アルゴリズムに従ってデータベースサーバによっても自動的に実行されます。アプリケーションは通常、**CHECKPOINT** を発行する必要がありません。

SAP Sybase IQ のチェックポイントの使用方法は、SQL Anywhere などの OLTP データベースとは異なります。多くの場合、OLTP データベースでは、少数のローにしかな影響しない短いトランザクションが実行されます。このため、ページ全体をディスクに書き込むのはコストがかかります。代わりに、OLTP データベースは通常、チェックポイントでディスクに書き込みを行い、変更されたデータローのみを書き込みます。SAP Sybase IQ は OLAP データベースです。1 つの OLAP トランザクションで数千、数百万のデータローを変更することがあります。このため、データベースサーバはチェックポイントの発生を待たずに物理的な書き込みを実行します。トランザクションがコミットするたびに更新されたデータページがディスクに書き込まれます。OLAP データベースの場合、任意のチェックポイントで少量のデータを書き込むより、データページ全体を書き込むことで効率が大幅に向上します。

チェックポイント時間の調整や明示的なチェックポイントの発行は、ほとんど必要ありません。SAP Sybase IQ は各トランザクションのコミット後に実際のデータページを書き込むため、SAP Sybase IQ でのチェックポイントの制御は OLTP データベース製品の場合ほど重要ではありません。

標準

(先頭に戻る) (120 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Adaptive Server でサポートされています。

パーミッション

(先頭に戻る) (120 ページ)

CHECKPOINT システム権限が必要です。

CLEAR 文 [Interactive SQL]

Interactive SQL (**dbisql**) の開いている結果セットをすべて閉じます。

クイックリンク：

「使用法」 (121 ページ)

「標準」 (121 ページ)

「パーミッション」 (121 ページ)

構文

CLEAR

使用法

(先頭に戻る) (121 ページ)

開いている結果セットをすべて閉じ、[SQL 文] ウィンドウ枠の内容は変更せずに保持します。

関連する動作：

CLEAR 文は、クリアしたデータに関連付けられているカーソルを閉じます。

標準

(先頭に戻る) (121 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - なし。

パーミッション

(先頭に戻る) (121 ページ)

なし

参照：

- EXIT 文 [Interactive SQL] (355 ページ)

CLOSE 文 [ESQL] [SP]

指定されたカーソルを閉じます。

クイックリンク：

「例」 (122 ページ)

「標準」 (123 ページ)

「パーミッション」 (123 ページ)

構文

```
CLOSE { identifier | host-variable }
```

例

(先頭に戻る) (122 ページ)

- **例 1** – Embedded SQL 内のカーソルを閉じます。

```
EXEC SQL CLOSE employee_cursor;  
EXEC SQL CLOSE :cursor_var;
```

- **例 2** – カーソルを使用します。

```
CREATE PROCEDURE TopCustomer (OUT TopCompany CHAR(35), OUT  
TopValue INT)  
BEGIN  
    DECLARE err_notfound EXCEPTION  
    FOR SQLSTATE '02000' ;  
    DECLARE curThisCust CURSOR FOR  
        SELECT CompanyName,  
            CAST( sum(SalesOrderItems.Quantity *  
                Products.UnitPrice) AS INTEGER) VALUE  
        FROM Customers  
        LEFT OUTER JOIN SalesOrders  
        LEFT OUTER JOIN SalesOrderItems  
        LEFT OUTER JOIN Products  
    GROUP BY CompanyName ;  
    DECLARE ThisValue INT ;  
    DECLARE ThisCompany CHAR(35) ;  
    SET TopValue = 0 ;  
    OPEN curThisCust ;  
    CustomerLoop:  
    LOOP  
        FETCH NEXT curThisCust  
        INTO ThisCompany, ThisValue ;  
        IF SQLSTATE = err_notfound THEN
```

```

    LEAVE CustomerLoop ;
  END IF ;
  IF ThisValue > TopValue THEN
    SET TopValue = ThisValue ;
    SET TopCompany = ThisCompany ;
  END IF ;
END LOOP CustomerLoop ;
CLOSE curThisCust ;
END

```

標準

(先頭に戻る) (122 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Adaptive Server でサポートされています。

パーミッション

(先頭に戻る) (122 ページ)

カーソルを事前にオープンしておく必要があります。

参照：

- DECLARE CURSOR 文 [ESQL] [SP] (299 ページ)
- OPEN 文 [ESQL] [SP] (461 ページ)
- PREPARE 文 [ESQL] (469 ページ)

COMMENT 文

データベースオブジェクトに関するコメントをシステムテーブルに格納します。

クイックリンク：

「例」 (124 ページ)

「使用法」 (124 ページ)

「標準」 (125 ページ)

「パーミッション」 (125 ページ)

構文

```

COMMENT ON
  { COLUMN [ owner.] table-name.column-name
  | DBSPACE dbspace-name
  | EVENT event-name
  | EXTERNAL [ENVIRONMENT] OBJECT object-name

```

```

| EXTERNAL ENVIRONMENT environment-name
| EXTERNAL OBJECT object-name
| FOREIGN KEY [owner.]table-name.role-name
| INDEX [ [owner.]table.]index-name
| INTEGRATED LOGIN integrated-login-id
| JAVA CLASS java-class-name
| JAVA JAR java-jar-name
| KERBEROS LOGIN "client-Kerberos-principal"
| LDAP SERVER ldap-server-name
| LOGICAL SERVER logical-server-name
| LOGIN POLICY policy-name
| LS POLICY ls-policy-name
| MATERIALIZED VIEW [owner.]materialized-view-name
| PRIMARY KEY ON [owner.]table-name
| PROCEDURE [owner.]table-name
| ROLE role-name
| SERVICE web-service-name
| SEQUENCE [owner.]sequence-name
| SPATIAL REFERENCE SYSTEM srs-name
| SPATIAL UNIT OF MEASURE uom-identifier
| TABLE [ owner.]table-name
| TEXT CONFIGURATION [ owner.]text-config-name
| TEXT INDEX text-index-name
| TRIGGER [[owner.]table-name.]trigger-name
| USER userid
| VIEW [ owner.]view-name
}
IS comment

environment-name - (back to Syntax)
    JAVA | PERL | PHP | C_ESQL32 | C_ESQL64 | C_ODBC32 | C_ODBC64

comment - (back to Syntax)
    { string | NULL }

```

例

(先頭に戻る) (123 ページ)

- **例 1** – Employees テーブルにコメントを追加します。

```

COMMENT
ON TABLE Employees
IS "Employee information"

```

- **例 2** – Employees テーブルからコメントを削除します。

```

COMMENT
ON TABLE Employees
IS NULL

```

使用法

(先頭に戻る) (123 ページ)

COMMENT 文は、`ISYSREMARK` システムテーブル内の注釈を更新します。コメントは、`NULL` に設定すると削除できます。インデックスまたはトリガに関するコメントの所有者は、インデックスまたはトリガが定義されているテーブルの所有者になります。

COMMENT ON DBSPACE 文、**COMMENT ON JAVA JAR** 文、**COMMENT ON JAVA CLASS** 文を使用すると、`SYS.ISYSREMARK` システムテーブルに `Remarks` カラムを設定できます。コメントは、`NULL` に設定して削除します。

ローカルテンポラリテーブルにはコメントを追加できません。

注意： マテリアライズドビューは、IQ カタログストアの SQL Anywhere テーブルでのみサポートされます。

標準

(先頭に戻る) (123 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Adaptive Server によりサポートされていません。

パーミッション

(先頭に戻る) (123 ページ)

句	必要な権限
COLUMN	次のいずれかが必要。 <ul style="list-style-type: none"> • テーブルを所有している • <code>CREATE ANY TABLE</code> システム権限 • <code>ALTER ANY TABLE</code> システム権限 • <code>CREATE ANY OBJECT</code> システム権限 • <code>ALTER ANY OBJECT</code> システム権限 • <code>COMMENT ANY OBJECT</code> システム権限
DBSPACE	<code>MANAGE ANY DBSPACE</code> システム権限
EVENT	次のいずれかが必要。 <ul style="list-style-type: none"> • <code>MANAGE ANY EVENT</code> • <code>CREATE ANY OBJECT</code> • <code>ALTER ANY OBJECT</code> • <code>COMMENT ANY OBJECT</code>

句	必要な権限
EXTERNAL [ENVIRONMENT] OBJECT	MANAGE ANY EXTERNAL OBJECT システム権限
EXTERNAL ENVIRONMENT	MANAGE ANY EXTERNAL ENVIRONMENT システム権限
FOREIGN KEY	次のいずれかが必要。 <ul style="list-style-type: none"> • テーブルを所有している • CREATE ANY TABLE システム権限 • ALTER ANY TABLE システム権限 • CREATE ANY OBJECT システム権限 • ALTER ANY OBJECT システム権限 • COMMENT ANY OBJECT システム権限
INDEX	次のいずれかが必要。 <ul style="list-style-type: none"> • インデックスを所有している • CREATE ANY INDEX システム権限 • ALTER ANY INDEX システム権限 • COMMENT ANY OBJECT システム権限 • CREATE ANY OBJECT システム権限 • ALTER ANY OBJECT システム権限
INTEGRATED LOGIN	MANAGE ANY USER システム権限
JAVA CLASS または JAVA JAR	MANAGE ANY EXTERNAL OBJECT システム権限
KERBEROS LOGIN	MANAGE ANY USER システム権限
LDAP SERVER	MANAGE ANY LDAP SERVER システム権限
LOGICAL SERVER	MANAGE MULTIPLEX システム権限
LOGIN POLICY	MANAGE ANY LOGIN POLICY システム権限
LS POLICY	MANAGE MULTIPLEX システム権限

句	必要な権限
MATERIALIZED VIEW	次のいずれかが必要。 <ul style="list-style-type: none"> ビューを所有している CREATE ANY MATERIALIZED VIEW システム権限 ALTER ANY MATERIALIZED VIEW システム権限 CREATE ANY OBJECT システム権限 ALTER ANY OBJECT システム権限 COMMENT ANY OBJECT システム権限
PRIMARY KEY ON	次のいずれかが必要。 <ul style="list-style-type: none"> テーブルを所有している CREATE ANY TABLE システム権限 ALTER ANY TABLE システム権限 CREATE ANY OBJECT システム権限 ALTER ANY OBJECT システム権限 COMMENT ANY OBJECT システム権限
PROCEDURE	次のいずれかが必要。 <ul style="list-style-type: none"> プロシージャを所有している CREATE ANY PROCEDURE システム権限 ALTER ANY PROCEDURE システム権限 CREATE ANY OBJECT システム権限 ALTER ANY OBJECT システム権限 COMMENT ANY OBJECT システム権限
SEQUENCE	次のいずれかが必要。 <ul style="list-style-type: none"> シーケンスを所有している CREATE ANY SEQUENCE システム権限 ALTER ANY SEQUENCE システム権限 CREATE ANY OBJECT システム権限 ALTER ANY OBJECT システム権限 COMMENT ANY OBJECT システム権限
SERVICE	MANAGE ANY WEB SERVICE システム権限

句	必要な権限
SPATIAL REFERENCE SYSTEM	次のいずれかが必要。 <ul style="list-style-type: none"> COMMENT ANY OBJECT CREATE ANY OBJECT ALTER ANY OBJECT MANAGE ANY SPATIAL OBJECT
SPATIAL UNIT OF MEASURE	次のいずれかが必要。 <ul style="list-style-type: none"> COMMENT ANY OBJECT CREATE ANY OBJECT ALTER ANY OBJECT MANAGE ANY SPATIAL OBJECT
ROLE	システムロール - コメント対象のロールに対する管理権限 ユーザ定義ロール - MANAGE ROLES システム権限またはコメント対象のロールに対する管理権限
TABLE	次のいずれかが必要。 <ul style="list-style-type: none"> テーブルを所有している CREATE ANY TABLE システム権限 ALTER ANY TABLE システム権限 CREATE ANY OBJECT システム権限 ALTER ANY OBJECT システム権限 COMMENT ANY OBJECT システム権限
TEXT CONFIGURATION	次のいずれかが必要。 <ul style="list-style-type: none"> テキスト設定の作成者である CREATE ANY TEXT CONFIGURATION システム権限 ALTER ANY TEXT CONFIGURATION システム権限 CREATE ANY OBJECT システム権限 ALTER ANY OBJECT システム権限 COMMENT ANY OBJECT システム権限

句	必要な権限
TEXT INDEX	次のいずれかが必要。 <ul style="list-style-type: none"> • テキストインデックスの作成者である • CREATE ANY INDEX システム権限 • ALTER ANY INDEX システム権限 • CREATE ANY OBJECT システム権限 • ALTER ANY OBJECT システム権限 • COMMENT ANY OBJECT システム権限
TRIGGER	次のいずれかが必要。 <ul style="list-style-type: none"> • トリガの作成者である • CREATE ANY TRIGGER システム権限 • ALTER ANY TRIGGER システム権限 • CREATE ANY OBJECT システム権限 • ALTER ANY OBJECT システム権限 • COMMENT ANY OBJECT システム権限
USER	MANAGE ANY USER システム権限
VIEW	次のいずれかが必要。 <ul style="list-style-type: none"> • ビューを所有している • CREATE ANY VIEW システム権限 • ALTER ANY VIEW システム権限 • CREATE ANY OBJECT システム権限 • ALTER ANY OBJECT システム権限 • COMMENT ANY OBJECT システム権限

COMMIT 文

データベースへの変更を保存します。またはユーザ定義のトランザクションを終了します。

クイックリンク：

「例」 (130 ページ)

「使用法」 (130 ページ)

「標準」 (131 ページ)

「パーミッション」 (131 ページ)

構文

構文 1 – トランザクションを終了し、すべての変更内容を永続化します。

```
COMMIT [ WORK ]
```

構文 2 – ネストされたトランザクションを作成します

```
COMMIT TRAN[SACTION ] [ transaction-name ]
```

例

(先頭に戻る) (129 ページ)

- **例 1** – 現在のトランザクションをコミットします。

```
COMMIT
```

- **例 2** – 次の Transact-SQL バッチは、@@trancount の連続した値を、0、1、2、1、0 としてレポートします。

```
PRINT @@trancount
BEGIN TRANSACTION
PRINT @@trancount
BEGIN TRANSACTION
PRINT @@trancount
COMMIT TRANSACTION
PRINT @@trancount
COMMIT TRANSACTION
PRINT @@trancount
go
```

使用法

(先頭に戻る) (129 ページ)

- **構文 1** – データ定義文は、自動的にコミットします。詳細については、各 SQL 文の「関連する動作」を参照してください。

データベースサーバが無効な外部キーを検出すると、**COMMIT** 文は失敗します。その場合、無効な外部キーがあることによってトランザクションを終了できなくなります。通常、外部キー整合性はそれぞれのデータ操作オペレーションごとにチェックされます。ただし、データベースオプション `WAIT_FOR_COMMIT` が `ON` に設定されているか、または特殊な外部キーが **CHECK ON COMMIT** 句によって定義されている場合、データベースサーバは **COMMIT** 文が実行されるまで整合性のチェックを遅延します。

- **構文 2** – ネストされたトランザクションはセーブポイントに似ています。ネストされたトランザクションの一番外側で実行された場合は、この文によってデータベースに対する変更が永続化されます。トランザクション内で実行した場合、**COMMIT TRANSACTION** 文はトランザクションのネストレベルを 1 つ減ら

します。トランザクションがネストされているときは、最も外側の **COMMIT** だけがデータベースに対する変更を永続化します。

オプションのパラメータ *transaction-name* はこのトランザクションに割り当てられた名前です。有効な識別子にする必要があります。トランザクション名は、ネストされた **BEGIN/COMMIT** 文または **BEGIN/ROLLBACK** 文の最も外側のペアでのみ使用してください。

さまざまなオプションを使用して、**COMMIT** 文の動作を詳細に制御できます。「**COOPERATIVE_COMMIT_TIMEOUT** オプション」、「**COOPERATIVE_COMMITS** オプション」、「**DELAYED_COMMITS** オプション」、および「**DELAYED_COMMIT_TIMEOUT** オプション」を参照してください。**Commit** 接続プロパティを使用して、現在の接続のコミット数を取得できます。

関連する動作：

- **WITH HOLD** によって開かれたカーソルを除き、すべてのカーソルを閉じます。
- **ON COMMIT PRESERVE ROWS** を使用して宣言されたものを除き、この接続で宣言されたテンポラリテーブルのすべてのローを削除します。

標準

(先頭に戻る) (129 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - Adaptive Server でサポートされています。構文 2 は、ISO/ANSI SQL 文法の Transact-SQL 拡張です。

パーミッション

(先頭に戻る) (129 ページ)

データベースに接続しておく必要があります。

参照：

- **BEGIN TRANSACTION** 文 [T-SQL] (113 ページ)
- **CONNECT** 文 [ESQL] [Interactive SQL] (133 ページ)
- **DISCONNECT** 文 [Interactive SQL] (318 ページ)
- **ROLLBACK** 文 (518 ページ)
- **SAVEPOINT** 文 (521 ページ)
- **SET CONNECTION** 文 [ESQL] [Interactive SQL] (539 ページ)
- **COOPERATIVE_COMMIT_TIMEOUT** オプション (633 ページ)

SQL 文

- COOPERATIVE_COMMITS オプション (634 ページ)
- DELAYED_COMMITS オプション (654 ページ)
- DELAYED_COMMIT_TIMEOUT オプション (653 ページ)

CONFIGURE 文 [Interactive SQL]

Interactive SQL (**dbisql**) 設定ウィンドウをアクティブにします。

クイックリンク：

「使用法」 (132 ページ)

「標準」 (132 ページ)

「パーミッション」 (132 ページ)

構文

CONFIGURE

使用法

(先頭に戻る) (132 ページ)

dbisql 設定ウィンドウには、すべての **dbisql** オプションの現在の設定が表示されます。データベースオプションは表示されず、変更を加えることもできません。

[Permanent] を選択すると、オプションはデータベースの SYSOPTION テーブルに書き込まれ、データベースサーバは自動的に **COMMIT** を実行します。[Permanent] を選択しないで、[OK] をクリックすると、オプションは一時的に設定され、現在のデータベース接続の間だけに有効になります。

標準

(先頭に戻る) (132 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 — SAP Adaptive Server® Enterprise ではサポートされていません。

パーミッション

(先頭に戻る) (132 ページ)

なし

参照：

- SET OPTION 文 (542 ページ)

CONNECT 文 [ESQL] [Interactive SQL]

engine-name で識別されるサーバ上で稼働する、*database-name* で識別されるデータベースへの接続を確立します。

クイックリンク：

「パラメータ」 (133 ページ)

「例」 (134 ページ)

「使用法」 (134 ページ)

「標準」 (135 ページ)

「パーミッション」 (136 ページ)

構文

構文 1

CONNECT

```
... [ TO engine-name ]  
... [ DATABASE database-name ]  
... [ AS connection-name ]  
... [ USER ] userid [ IDENTIFIED BY ]
```

構文 2

```
CONNECT USING connect-string
```

パラメータ

(先頭に戻る) (133 ページ)

- **engine-name** – 識別子、文字列、またはホスト変数
- **database-name** – 識別子、文字列、またはホスト変数
- **connection-name** – 識別子、文字列、またはホスト変数
- **userid** – 識別子、文字列、またはホスト変数
- **password** – 識別子、文字列、またはホスト変数
- **connect-string** – keyword=*value* 形式のパラメータ設定のリスト。一重引用符で囲む必要があります。

- **AS** – 必要に応じて句を指定して、接続に名前を付けることができます。名前を付けることによって、同じデータベースへの複数の接続、あるいは同じまたは異なるデータベースサーバへの複数の接続を、すべて同時に行えます。それぞれの接続には、固有のトランザクションがあります。ただし、2つの異なる接続から同じデータベース内の同じレコードを修正しようとした場合などに、トランザクション間でロックの競合が起こることもあります。

例

(先頭に戻る) (133 ページ)

- **例 1** – 次は、Embedded SQL 内での **CONNECT** の使用例です。

```
EXEC SQL CONNECT AS :conn_name
USER :userid IDENTIFIED BY :password;
EXEC SQL CONNECT USER "dba" IDENTIFIED BY "sql";
```

- **例 2** – 次は、**dbisql** 内での **CONNECT** の使用例です。

dbisql からデータベースに接続します。ユーザ ID とパスワードの入力を要求するプロンプトが表示されます。

```
CONNECT
```

dbisql から DBA としてデフォルトのデータベースに接続します。パスワードの入力が求められます。

```
CONNECT USER "DBA"
```

dbisql から DBA としてデモデータベースに接続します。<machine>_iqdemo はエンジン名です。

```
CONNECT
TO <machine>_iqdemo
USER "DBA"
IDENTIFIED BY sql
```

接続文字列を使用して、**dbisql** からデモデータベースに接続します。

```
CONNECT
USING 'UID=DBA;PWD=sql;DBN=iqdemo'
```

使用法

(先頭に戻る) (133 ページ)

- **Embedded SQL の動作** – Embedded SQL では、*engine-name* を指定しない場合、デフォルトのローカルデータベースサーバ (最初に起動したデータベースサーバ) が使用されます。*database-name* を指定しない場合、指定したサーバの最初のデータベースが使用されます。

WHENEVER 文、**SET SQLCA** 文、および一部の **DECLARE** 文はコードを生成しないので、ソースファイル内の **CONNECT** 文の前に置いてかまいません。それ

以外の場合は、**CONNECT** 文が正常に実行されるまで、どのような文も使用できません。

ユーザ ID とパスワードを使用して、動的 SQL 文ごとにパーミッションをチェックします。デフォルトでは、パスワードの大文字と小文字は区別されますが、ユーザ ID の大文字と小文字は区別されません。パスワードの代わりにホスト変数を使用し、ホスト変数の値を NULL ポインタに設定することによって、パスワードなしの接続が可能です。

- **dbisql** の動作 - **CONNECT** 文でデータベースもサーバも指定しないと、**dbisql** はデフォルトのサーバおよびデータベースに接続するのではなく、現在のデータベースとの接続を継続します。サーバ名を指定せずにデータベースを指定した場合、**dbisql** は現在のサーバ上の指定したデータベースへの接続を試みます。データベースファイル名ではなく、**-n** データベーススイッチで定義されたデータベース名を指定する必要があります。データベース名を指定せずにサーバ名を指定した場合、**dbisql** は指定したサーバ上のデフォルトのデータベースに接続します。たとえば、データベースへの接続中に次のバッチを実行すると、同じデータベースに 2 つのテーブルが作成されます。

```
CREATE TABLE t1 ( c1 int );  
CONNECT DBA IDENTIFIED BY sql;  
CREATE TABLE t2 ( c1 int );
```

他のデータベース文は、**CONNECT** 文が正常に実行されるまで使用できません。

ユーザ ID とパスワードを使用して、SQL 文ごとにパーミッションをチェックします。パスワード、またはユーザ ID とパスワードが指定されていない場合、足りない情報を入力するようにプロンプトが表示されます。デフォルトでは、パスワードの大文字と小文字は区別されますが、ユーザ ID の大文字と小文字は区別されません。

複数の接続を管理するときにも、現在の接続という概念が使用されます。**CONNECT** 文が正常に実行されると、その新しい接続が現在の接続になります。別の接続に切り替えるには、**SET CONNECTION** を使用します。**CONNECT** 文を実行しても、既存の接続は (仮にあったとしても) クローズされません。接続を削除するには、**DISCONNECT** を使用します。

静的 SQL 文では、**SQLPP** 文の行に **-I** オプションで指定したユーザ ID とパスワードが使用されます。**-I** オプションを指定しないと、**CONNECT** 文のユーザ ID とパスワードが静的 SQL 文でも使用されます。

標準

(先頭に戻る) (133 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。

SQL 文

- SAP Sybase Database 製品 — Open Client Embedded SQL は **CONNECT** 文に対して別の構文をサポートします。

パーミッション

(先頭に戻る) (133 ページ)

なし

参照:

- DISCONNECT 文 [Interactive SQL] (318 ページ)
- GRANT CONNECT 文 (385 ページ)
- SET CONNECTION 文 [ESQL] [Interactive SQL] (539 ページ)

CREATE AGENT 文

SAP Control Center の SAP Sybase IQ エージェントと指定のサーバを関連付けて、高可用性をサポートします。

クイックリンク:

「例」 (136 ページ)

「使用法」 (137 ページ)

「パーミッション」 (137 ページ)

構文

```
CREATE AGENT FOR MULTIPLEX SERVER server-name  
  USER username IDENTIFIED BY agentpwd PORT portnum
```

適用対象

マルチプレックスのみ。

例

(先頭に戻る) (136 ページ)

- **例 1** - mpx_writer1 という名前の SAP Sybase IQ サーバのエージェントを作成します。ユーザログインは "sqltester"、ポート番号は 1138 です。

```
CREATE AGENT FOR MULTIPLEX SERVER mpx_writer1 USER sqltester  
IDENTIFIED BY '8U3dkA' PORT 1138
```

使用法

(先頭に戻る) (136 ページ)

SYS.ISYSIQMPXSERVERAGENT システムテーブルには、サーバのエージェント接続定義が格納されています。

関連する動作

- オートコミット

パーミッション

(先頭に戻る) (136 ページ)

MANAGE MULTIPLEX システム権限が必要です。

CREATE DATABASE 文

複数のオペレーティングシステムファイルから構成されるデータベースを作成します。

クイックリンク：

「パラメータ」 (138 ページ)

「例」 (144 ページ)

「使用法」 (145 ページ)

「標準」 (149 ページ)

「パーミッション」 (149 ページ)

構文

```

CREATE DATABASE db-name
... [ [ TRANSACTION ] { LOG ON [ log-file-name ]
      [ MIRROR mirror-file-name ] } ]
... [ CASE { RESPECT | IGNORE } ]
... [ PAGE SIZE catalog-page-size ]
... [ COLLATION collation-label[( collation-tailoring-string ) ] ]
... [ ENCRYPTED { algorithm-key-spec | OFF } ]
... [ BLANK PADDING ON ]
... [ JCONNECT { ON | OFF } ]
... [ IQ PATH iq-file-name ]
... [ IQ SIZE iq-file-size ]
... [ IQ PAGE SIZE iq-page-size ]
... [ BLOCK SIZE block-size ]
... [ IQ RESERVE sizeMB ]
... [ TEMPORARY RESERVE sizeMB ]
... [ MESSAGE PATH message-file-name ]

```

```

... [ TEMPORARY PATH temp-file-name ]
... [ TEMPORARY SIZE temp-db-size ]
... [ DBA USER userid ]
... [ DBA PASSWORD password ]
... [ SYSTEM PROCEDURE AS DEFINER {ON | OFF} ]

```

catalog-page-size (bytes) - (構文に戻る)
 { 4096 | 8192 | 16384 | 32768 }

collation-label - (構文に戻る)
string

collation-tailoring-string - (構文に戻る)
keyword=value

algorithm-key-spec - (構文に戻る)
 ON
 | [ON] KEY *key* [ALGORITHM *AES-algorithm*]
 | [ON] ALGORITHM *AES-algorithm* KEY *key*
 | [ON] ALGORITHM 'SIMPLE'

AES-algorithm - (back to algorithm-key-spec)
 'AES' | 'AES256' | 'AES_FIPS' | 'AES256_FIPS'

key: - (back to algorithm-key-spec)
quoted string

iq-page-size (bytes) - (構文に戻る)
 { 65536 | 131072 | 262144 | 524288 }

block-size (bytes) - (構文に戻る)
 { 4096 | 8192 | 16384 | 32768 }

パラメータ

(先頭に戻る) (137 ページ)

- **TRANSACTION LOG** – データベースサーバがデータベースに対して加えられたすべての変更を記録するファイル。トランザクションログはシステムリカバリで重要な役割を果たします。TRANSACTION LOG 句を指定しない場合、またはファイル名のパスを省略した場合、.db ファイルと同じディレクトリに格納されます。ただし、.db と .iq は別の物理デバイス上に格納する必要があります。ローパーティション上では作成できません。
- **MIRROR** – トランザクションログと同一のコピーで、通常はデータ保護を強化するため別のデバイスで管理されます。デフォルトでは、SAP Sybase IQ はミラー化されたトランザクションログを使用しません。トランザクションログミラーを使用する場合、ファイル名を指定する必要があります。相対パスを使用すると、トランザクションログミラーはカタログストア (db-name.db) のディレ

クトリを基準にした相対位置に作成されます。ヒント：トランザクションログのミラーコピーは必ず作成してください。

- **CASE – CASE RESPECT** を使用して作成したデータベースでは、比較と文字列操作で、対象となるすべての値の大文字と小文字が区別されます。カラム、プロシージャ、ユーザ ID などのデータベースオブジェクトの名前は対象外です。CASE の指定に関係なく、DB 領域名は常に大文字と小文字が区別されます。デフォルト (RESPECT) では、すべての比較において大文字と小文字が区別されます。CASE RESPECT を使用すれば、CASE IGNORE よりもパフォーマンスが高くなります。
- **PAGE SIZE** – カタログテーブルを含むデータベースの SQL Anywhere セグメントのページサイズは、4096、8192、16384、または 32768 バイトです。通常、デフォルトの 4096 (4KB) を使用します。データベースが大きい場合は、このデフォルトよりも大きいページサイズが必要になることがあり、その結果、パフォーマンスが高くなる場合があります。それより小さい値ではデータベースがサポートできるカラム数が制限される可能性があります。4096 よりも小さいページサイズを指定しても、SAP Sybase IQ によってページサイズは 4096 になります。
- **COLLATION** – データベース内の文字データ型のソートと比較で使用される照合順。照合によって、使用するエンコード (文字セット) に文字の比較と順序付けに関する情報が提供されます。COLLATION 句を指定していない場合、SAP Sybase IQ はオペレーティングシステムの言語とエンコードに基づいて照合を選択します。ほとんどのオペレーティングシステムでは、デフォルトの照合順である ISO_BINENG を使用して、最適なパフォーマンスを発揮します。ISO_BINENG では、照合順序は ASCII 文字セットの文字順序と同じです。すべての大文字がすべての小文字に先行します (たとえば、'A' も 'B' も 'a' に先行します)。

照合は、サポートされる照合リストから選択できます。SAP Sybase IQ サーバで作成された SQL Anywhere データベースでは、照合に Unicode Collation Algorithm (UCA) を使用することもできます。UCA を指定する場合、ENCODING 句も指定してください。SAP Sybase IQ では SAP Sybase IQ データベースの UCA ベースの照合はサポートされていません。UCA ベースの照合がデータベースの **CREATE DATABASE** 文で指定されている場合は、サーバからエラー「UCA collation is not supported」が返され、データベースを作成できません。照合順はデータベースの作成後は変更できません。

必要に応じて、文字列のソートや比較を詳細に制御することために、照合の調整オプション (*collation-tailoring-string*) を指定することもできます。これらのオプションは、keyword=value ペアの形式で指定し、カッコで囲んで、照合名に続けて指定します。

SAP Sybase IQ で使用する照合の調整オプションには、SAP Sybase IQ データベースの照合の調整オプション (*collation-tailoring-string*) で指定できるキーワード、別の形式、値が含まれています。

表 2 : SAP Sybase IQ で使用する照合の調整オプション

キーワード	照合	別の形式	指定可能な値
CaseSensitivity	サポートされているすべての照合	CaseSensitive、Case	<ul style="list-style-type: none"> • respect – 文字の大文字小文字の違いが考慮される。UCA 照合の場合、UpperFirst の指定と等しくなる。その他の照合では、respect の値は照合自体によって異なる。 • ignore – 文字の大文字小文字の違いは無視される。 • UpperFirst – 常に、大文字を先にソートする (Aa)。 • LowerFirst – 常に、小文字を先にソートする (aA)。

注意： 照合の適合化オプションの中には、SAP Sybase IQ サーバで作成された SQL Anywhere データベース用の UCA 照合を指定する場合にサポートされているものもあります。その他の照合と SAP Sybase IQ の場合は、大文字と小文字の区別の適合化オプションのみがサポートされます。また、照合の適合化オプションで作成したデータベースは、15.0 より前のデータベースサーバでは起動できません。

- **ENCRYPTED** – 物理データベースファイルに格納されたデータを判読不能にします。データベース全体を暗号化するには、CREATE DATABASE ENCRYPTED キーワード (TABLE キーワードなし) を使用します。ENCRYPTED TABLE 句は、SQL Anywhere テーブルのテーブル暗号化のみを有効にする場合に使用します。テーブルレベルの暗号化は、SAP Sybase IQ テーブルではサポートされていません。テーブルの暗号化を有効化すると、以降に ENCRYPTED 句を使用して作成されるテーブルや変更されるテーブルがデータベースの作成時に指定した設定を使用して暗号化されることとなります。

データベース暗号化には、単純と強力という 2 つのレベルがあります。

- 単純暗号化は、難読化と同じです。データは判読できませんが、暗号に関する専門知識を持つユーザはデータを解読できます。単純暗号化の場合は、CREATE DATABASE 句の ENCRYPTED ON ALGORITHM 'SIMPLE' または ENCRYPTED ALGORITHM 'SIMPLE' を指定するか、アルゴリズムやキーを指定せずに ENCRYPTED ON 句を指定します。

- 強力な暗号化は、128 ビットのアルゴリズムとセキュリティキーを使用して処理されます。データは判読不能で、キーがなければ事実上解読できません。強力な暗号化では、128 ビットまたは 256 ビットの AES アルゴリズムを使用する **CREATE DATABASE** 句の **ENCRYPTED ON ALGORITHM** を指定し、**KEY** 句を使用して暗号化キーを指定します。キーの値には、少なくとも 16 文字の長さを持ち、大文字と小文字を含み、数字、文字、特殊文字を使用したものを選ぶことをおすすめします。
この暗号化キーは、データベースを起動するたびに入力する必要があります。

暗号化は、データベースの作成時にのみ指定できます。暗号化を既存のデータベースに導入するには、完全にアンロードした後でデータベースを再作成し、すべてのデータを再ロードする必要があります。アルゴリズムを指定せずに **ENCRYPTED** 句を使用すると、デフォルトの AES が使用されます。デフォルトでは、暗号化は OFF です。

警告！ 暗号化キーは保護する必要があります。キーのコピーを安全な場所に保管してください。キーを失うと、データベースにまったくアクセスできなくなります。そのうえ、修復する方法もありません。

- BLANK PADDING** – 比較の際に後続ブランクは無視されます (**BLANK PADDING ON**)。Embedded SQL プログラムは文字配列にフェッチされた文字列を埋め込みます。このオプションは、ISO/ANSI SQL 標準との互換性を保つために用意されています。**CREATE DATABASE** では、**BLANK PADDING OFF** がサポートされなくなりました。
- JCONNECT** – SAP Sybase jConnect for JDBC ドライバを使用してシステムカタログ情報にアクセスするには、jConnect サポートをインストールします。**JCONNECT** を OFF に設定すると、jConnect システムオブジェクトが除外されます (デフォルトは ON)。その場合でも、システム情報にアクセスしないかぎり、JDBC を使用できます。
- IQ PATH** – SAP Sybase IQ データを含むメインセグメントファイルのパス名。オペレーティングシステムファイル、または I/O デバイスのローパーティションを指定できます (ローパーティションを指定する場合のフォーマットについては、各プラットフォーム用の『インストールおよび設定ガイド』を参照してください)。SAP Sybase IQ は指定したパス名に基づいて型を自動的に検出します。相対パスを使用すると、ファイルはカタログストア (.db ファイル) のディレクトリに相対して作成されます。

IQ PATH 句を省略して、次のいずれかのオプションを指定するとエラーが生成されます。**IQ SIZE**、**IQ PAGE SIZE**、**BLOCK SIZE**、**MESSAGE PATH**、**TEMPORARY PATH**、および **TEMPORARY SIZE**。

- **IQ SIZE – IQ PATH** 句で指定するローパーティションまたはオペレーティングシステムファイルのいずれかのサイズ (MB 単位)。ローパーティションの場合は、IQ SIZE を指定せずに、常にデフォルトを使用してください。これにより、SAP Sybase IQ がローパーティション全体を使用できます。IQ SIZE を指定する場合は、IQ SIZE の値を I/O デバイスのサイズに一致させる必要があります。そうしないと、SAP Sybase IQ からエラーが返されます。オペレーティングシステムファイルの場合は、次の表の最小値から最大値 (4TB) までの間で指定できます。

オペレーティングシステムファイルのデフォルトサイズは IQ PAGE SIZE に応じて異なります。

表 3 : IQ ストアファイルとテンポラリストアファイルのデフォルトと最小サイズ

IQ ページサイズ	IQ SIZE のデフォルト	TEMPORARY SIZE のデフォルト	明示的な IQ SIZE の最小値	明示的な TEMPORARY SIZE の最小値
65536	4096000	2048000	4MB	2MB
131072	8192000	4096000	8MB	4MB
262144	16384000	8192000	16MB	8MB
524288	32768000	16384000	32MB	16MB

- **IQ PAGE SIZE** – (IQ のテーブルとインデックスが格納された) データベースの SAP Sybase IQ セグメントのページサイズ (バイト単位)。値は 65536 から 524288 バイトまでの 2 の累乗です。デフォルトは 131072 (128KB) です。サイズにこれ以外の値を指定すると、その値より大きい次の値に変更されます。IQ ページサイズは、データベースのデフォルト I/O 転送ブロックサイズと最大データ圧縮比を決定します。

最良のパフォーマンスを得るには、次の最小ページサイズを使用します。

- 最も大きいテーブルのロー数が 10 億以下のデータベース、または合計サイズが 8TB 未満のデータベースの場合は、64KB (IQ PAGE SIZE 65536)。これは新規データベースの最小サイズです。32 ビットプラットフォームでは、IQ ページサイズを 64KB にすると最高のパフォーマンスが得られます。
- 最も大きいテーブルに含まれるローの数が 10 億より多く 40 億未満であるか、合計サイズが 8TB 以上になる可能性がある、64 ビットプラットフォーム上のデータベースの場合は、128KB (IQ PAGE SIZE 131072)。128KB はデフォルトの IQ ページサイズです。

- 最も大きいテーブルに含まれるローの数が 40 億を超え、合計サイズが 8TB 以上になる可能性がある、64 ビットプラットフォーム上のデータベースの場合は、256KB (IQ PAGE SIZE 262144)。
- **BLOCK SIZE** – データベースの SAP Sybase IQ セグメントの I/O 転送ブロックサイズ (バイト単位)。IQ PAGE SIZE よりも小さい、4096 から 32768 までの 2 の累乗の値を指定してください。それ以外の値をサイズに指定すると、その値より大きい次のサイズに変更されます。デフォルト値は IQ PAGE SIZE 句の値に応じて異なります。ほとんどのアプリケーションでは、デフォルト値を使用するのが最適です。
- **IQ RESERVE** – 将来の DB ファイルのサイズ拡大に備えてメイン IQ ストア (IQ_SYSTEM_MAIN DB 領域) 用に確保する領域のサイズ (メガバイト単位)。sizeMB パラメータには、0 よりも大きい任意の数値を指定できます。DB 領域の作成後は、この予約サイズを変更できなくなります。IQ RESERVE を指定すると、データベースの内部 (フリーリスト) 構造により多くの領域が使用されるようになります。予約サイズが大きすぎると、内部構造体で必要とする領域が、指定されたサイズよりも大きくなり、エラーになります。
- **TEMPORARY RESERVE** – 将来の DB ファイルのサイズ拡大に備えてテンポラリ IQ ストア (IQ_SYSTEM_TEMP DB 領域) 用に確保する領域のサイズ (メガバイト単位)。sizeMB パラメータには、0 よりも大きい任意の数値を指定できます。DB 領域の作成後は、この予約サイズを変更できなくなります。TEMPORARY RESERVE を指定すると、データベースの内部 (フリーリスト) 構造により多くの領域が使用されるようになります。予約サイズが大きすぎると、内部構造体で必要とする領域が、指定されたサイズよりも大きくなり、エラーになります。

注意： データベースがバックアップからリストアされると、テンポラリ DB 領域用に確保された領域とモードは破棄されます。

- **MESSAGE PATH** – SAP Sybase IQ メッセージトレースファイルを格納するセグメントのパス名。オペレーティングシステムファイルを指定する必要があります。メッセージファイルはローパーティション上には作成できません。相対パスを指定するか、パスを省略した場合、メッセージファイルは .db ファイルのディレクトリに相対して作成されます。
- **TEMPORARY SIZE** – TEMPORARY PATH 句で指定するローパーティションまたはオペレーティングシステムファイルのいずれかのサイズ (メガバイト単位)。ローパーティションの場合、TEMPORARY SIZE を指定せずに常にデフォルトを使用してください。これによって SAP Sybase IQ はローパーティション全体を使用できるようになります。オペレーティングシステムファイルのデフォルトは、常に IQ SIZE 値の 2 分の 1 です。IQ ストアがローパーティションにあり、テンポラリストアがオペレーティングシステムファイルである場合、

デフォルトの TEMPORARY SIZE は IQ ストアのローパーティションの半分のサイズになります。

- **DBA USER – SYS_AUTH_DBA_ROLE** システムロールが付与されたデフォルトのユーザアカウントのユーザ名。この句を指定しない場合、SAP Sybase IQ によってデフォルトの DBA ユーザ ID が作成されます。
- **DBA PASSWORD – SYS_AUTH_DBA_ROLE** システムロールが付与されたデフォルトのユーザアカウントのパスワード。
- **SYSTEM PROCEDURE AS DEFINER** – 権限付きシステムプロシージャが invoker (プロシージャの実行者) と definer (プロシージャの所有者) のどちらの権限で実行されるかを定義します。OFF (デフォルト) の場合、つまり指定しないと、すべての権限付きシステムプロシージャが、invoker の権限で実行されます。sp_proc_priv() を使用して、システムプロシージャの実行に必要なシステム権限を指定します。

ON の場合、16.0 より前の権限付きシステムプロシージャは、definer の権限で実行されます。16.0 以降の権限付きシステムプロシージャは、invoker の権限で実行されます。

例

(先頭に戻る) (137 ページ)

- **例 1** – 次の Windows の例は、mydb という名前の SAP Sybase IQ データベースを、関連する mydb.db、mydb.iq、mydb.iqtmp、mydb.iqmsg の各ファイルと共に C:\s1\data ディレクトリに作成します。

```
CREATE DATABASE 'C:\s1\data\mydb'
BLANK PADDING ON
IQ PATH 'C:\s1\data'
IQ SIZE 2000
IQ PAGE SIZE 131072
```

- **例 2** – 次の UNIX コマンドは、IQ PATH と TEMPORARY PATH 用のローデバイスで SAP Sybase IQ データベースを作成します。IQ のデフォルトのページサイズである 128KB が適用されています。

```
CREATE DATABASE '/s1/data/bigdb'
IQ PATH '/dev/md/rdisk/bigdb'
MESSAGE PATH '/s1/data/bigdb.iqmsg'
TEMPORARY PATH '/dev/md/rdisk/bigtmp'
```

- **例 3** – 次の Windows コマンドは、IQ PATH 用のローデバイスで SAP Sybase IQ データベースを作成します。ローデバイス名には円記号を 2 つ続けて指定します (Windows の仕様)。

```
CREATE DATABASE 'company'
IQ PATH '¥¥¥¥.¥¥E:'
```

```
JCONNECT OFF
IQ SIZE 40
```

- **例 4** – 次の UNIX の例は、AES 暗号化アルゴリズムとキー "is!seCret" を使用して、強力な暗号で保護された SAP Sybase IQ データベースを作成します。

```
CREATE DATABASE 'marvin.db'
BLANK PADDING ON
CASE RESPECT
COLLATION 'ISO_BINENG'
IQ PATH '/filesystem/marvin.main1'
IQ SIZE 6400
IQ PAGE SIZE 262144
TEMPORARY PATH '/filesystem/marvin.temp1'
TEMPORARY SIZE 3200
ENCRYPTED ON KEY 'is!seCret' ALGORITHM 'AES'
```

使用法

(先頭に戻る) (137 ページ)

指定された名前と属性でデータベースを作成します。SAP Sybase IQ データベースの作成には IQ PATH 句が必須です。この句が指定されていない場合は、標準の SQL Anywhere データベースが作成されます。

SAP Sybase IQ は、データベース作成時に、データベースを構成するさまざまなタイプのデータを格納するための 4 つのデータベースファイルを自動生成します。各ファイルは DB 領域に対応し、その論理名を使って SAP Sybase IQ はデータベースファイルを識別します。

- *db-name.db* は、カタログ DB 領域 SYSTEM を保持するファイルです。システムテーブルのほか、データベースとユーザが追加するすべての標準 SQL Anywhere データベースオブジェクトについて記述するストアプロシージャが格納されます。*.db* 拡張子を付けずに指定すると、SAP Sybase IQ で自動的に追加されます。この最初の DB 領域には、カタログストアが格納されており、後で、DB 領域を追加してカタログストアのサイズを大きくすることが可能です。ローパーティション上では作成できません。
- *db-name.iq* は、IQ テーブルとインデックスを格納するメインデータ DB 領域 IQ_SYSTEM_MAIN を保持するファイルのデフォルト名です。IQ PATH 句を使用して別のファイル名を指定できます。この最初の DB 領域には、IQ ストアが格納されています。

警告！ IQ_SYSTEM_MAIN は特殊な DB 領域で、IQ db_identity ブロック、IQ チェックポイントログ、コミットされた各トランザクションおよびチェックポイントが設定されたアクティブな各トランザクションの IQ ロールフォワード / ロールバックビットマップ、インクリメンタルバックアップビットマップ、およびフリーリストルートページといった、データベースを開く際に必要なす

すべての構造体が格納されます。IQ_SYSTEM_MAIN は、データベースが開いているときは常にオンラインです。

管理者は、特に、テーブルが小さく、重要なテーブルである場合に、IQ_SYSTEM_MAIN でのユーザテーブルの作成を許可できます。ただし、一般的な方法では、データベースの作成後直ちに、管理者は、2つ目のメイン DB 領域を作成して、DB 領域 IQ_SYSTEM_MAIN における CREATE 権限をすべてのユーザから取り消します。その後、新しいメイン DB 領域に対する CREATE 権限を選択したユーザに付与して、PUBLIC.default_dbSPACE を新しいメイン DB 領域に設定します。

- *db-name.iqtmp* は、初期テンポラリ DB 領域 IQ_SYSTEM_TEMP を保持するファイルのデフォルト名です。特定のクエリによって生成されたテンポラリテーブルを格納します。このファイルの必要サイズはクエリのタイプとデータ量に応じて異なります。TEMPORARY PATH 句を使用すると別のファイル名を指定できます。この最初の DB 領域には、テンポラリストアが格納されています。
- *db-name.iqmsg* は、メッセージトレース DB 領域 IQ_SYSTEM_MSG を格納するファイルのデフォルト名です。MESSAGE PATH 句を使用すると別のファイル名を指定できます。

これらのファイルのほかに、データベースにはトランザクションログファイル (*db-name.log*) が存在します。また、トランザクションログミラーファイルが存在する場合もあります。

ファイル名 (*db-name*、*log-file-name*、*mirror-file-name*、*iq-file-name*、*message-file-name*、*temp-file-name*) は、オペレーティングシステムファイル名を含む文字列です。リテラル文字列として、一重引用符で囲んでください。

Windows でパスを指定する場合、円記号 (¥) の後に n または x がある場合は円記号を2つ重ねます。こうすることで、SQL の文字列の規則に従って、改行文字 (¥n) または 16 進数字 (¥x) として解釈されるのを回避できます。円記号は常にエスケープした方が安全です。次に例を示します。

```
CREATE DATABASE 'c:¥¥sybase¥¥mydb.db'
LOG ON 'e:¥¥logdrive¥¥mydb.log'
JCONNECT OFF
IQ PATH 'c:¥¥sybase¥¥mydb'
IQ SIZE 40
```

パスを指定しない場合、または相対パスを指定する場合は、以下のことに注意してください。

- カタログストアファイル (*db-name.db*) は、サーバの作業ディレクトリを基準とする相対位置に作成されます。

- IQ ストア、テンポラリストア、メッセージログのファイルは、カタログストアと同じディレクトリ内に作成されるか、カタログストアを基準とする相対位置に作成されます。

相対パス名の使用をおすすめします。

警告！ データベースファイル、テンポラリ DB 領域ファイル、トランザクションログファイルは、データベースサーバと同じ物理マシンに配置する必要があります。データベースファイルとトランザクションログファイルをネットワークドライブに置かないでください。ただし、トランザクションログは、そのミラーファイルとは別のデバイスに置く必要があります。

UNIX 系オペレーティングシステムでは、指定するファイルのパス名を含む間接的なポインタであるシンボリックリンクを作成できます。シンボリックリンクを相対パス名として使用できます。データベースファイル名のシンボリックリンクを作成することには、以下の利点があります。

- 実際のデバイス名は一般に意味を持たないが、ローデバイスへのシンボリックリンクには、意味のある名前を付けることができる。
- シンボリック名を使用すると、バックアップ以降に別のディレクトリへ移動されたデータベースファイルをリストアする際の問題を回避できる。

シンボリックリンクを作成するには、**ln -s** コマンドを使用します。次に例を示します。

```
ln -s /disk1/company/iqdata/company.iq company_iq_store
```

このリンクを作成しておけば、完全修飾パス名を使用する代わりに **CREATE DATABASE** または **RESTORE DATABASE** などのコマンドでシンボリックリンクを指定できます。

データベースや DB 領域を作成する際、パスはすべての DB 領域ファイルでユニークである必要があります。**CREATE DATABASE** コマンドで、これら 2つのストアに対して同じパスとファイル名を指定すると、エラーが返されます。

次のいずれかの方法で、ユニークなパスを作成できます。

- ファイルごとに異なる拡張子を指定する (mydb.iq、mydb.iqtmp など)。
- 異なるファイル名を指定する (mydb.iq および mytmp.iq など)。
- 異なるパス名 (/iqfiles/main/iq、/iqfiles/temp/iq など) または異なるローパーティションを指定する。
- データベースの作成時に、**TEMPORARY PATH** を省略する。この場合、テンポラリストアは、デフォルトの名前と拡張子 `dbname.iqtmp` でカタログストアと同じパスに作成されます。*dbname* は、データベース名です。

警告！ UNIX 系オペレーティングシステムでデータベースの一貫性を保つためには、別のファイルへのリンクであるファイルの名前を指定する必要があります。

SAP Sybase IQ では、リンクされたファイルが指すターゲットを検出できません。コマンド内でファイル名が異なっている場合、同じオペレーティングシステムファイルポイントを指していないか確認してください。

テーブルに挿入される文字列は、データベースで大文字と小文字が区別されるかどうかに関係なく、常に入力された大文字と小文字がそのまま格納されます。文字列 Value は、文字データ型カラムに挿入される場合、常に大文字の V と残りの文字が小文字でデータベースに格納されます。**SELECT** 文は、Value として文字列を返します。ただし、データベースで大文字と小文字が区別されない場合は、すべての比較で 'value' は 'value'、'VALUE' などと同じとみなされます。SAP Sybase IQ サーバは大文字と小文字を任意に組み合わせた結果を返すので、大文字と小文字を区別しない (CASE IGNORE) データベースで、大文字と小文字を区別する結果は期待できません。

たとえば、次のテーブルとデータがあるとします。

```
CREATE TABLE tb (id int NOT NULL,  
                 string VARCHAR(30) NOT NULL);  
INSERT INTO tb VALUES (1, 'ONE');  
SELECT * FROM tb WHERE string = 'oNe';
```

SELECT の結果は "oNe" (WHERE 句で指定) となることがあり、必ずしも "ONE" (データベースに格納される形) になるとはかぎりません。

同じように、次の結果は

```
SELECT * FROM tb WHERE string = 'One';
```

"One" となることがあり、次の結果は

```
SELECT * FROM tb WHERE string = 'ONE';
```

"ONE" となることがあります。

すべてのデータベースは、次のように 1 つ以上のユーザ ID と、

```
DBA
```

パスワードを指定して作成されます。

```
sql
```

新しいデータベースでは、データベース内の設定に関係なく、すべてのパスワードの大文字と小文字が区別されます。ユーザ ID は CASE RESPECT 設定に影響されません。

データベースの起動時に、ページサイズを現在のサーバのページサイズより大きくすることはできません。サーバのページサイズは、最初に起動されたデータベースによって決まるか、またはサーバのコマンドラインで **-gp** コマンドラインオプションを指定して設定します。

すべての文で、コマンドラインの長さはカタログページサイズに制限されます。デフォルトの 4KB で十分なことがほとんどですが、多数の DB 領域を参照する **RESTORE DATABASE** コマンドなど、非常に長いコマンドに対応できるように、大きな PAGE SIZE 値が必要となるケースもまれにあります。多数のテーブルまたはビューを伴うクエリの実行でも、ページサイズを大きくする必要が生じる場合があります。

デフォルトのカタログページサイズは 4KB なので、ページサイズが 1024 である `utility_db` などのデータベースに接続する場合にのみ問題が発生します。この制限によって、多くの DB 領域を参照する **RESTORE DATABASE** コマンドが失敗する可能性があります。この問題を回避するには、SQL コマンドラインの長さがカタログページサイズ以上にならないようにします。

または、**-gp 32768** を指定してエンジンを開始して、カタログページサイズを増加させます。

関連する動作：

- オートコミット

標準

(先頭に戻る) (137 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 — Adaptive Server は、**CREATE DATABASE** 文を用意していますが、オプションは異なります。

パーミッション

(先頭に戻る) (137 ページ)

この文を実行するのに必要なパーミッションは、以下のように、**-gu** サーバコマンドラインオプションを使用して設定します。

- **NONE** – この文を発行できるユーザはいません。
- **DBA – SERVER OPERATOR** システム権限が必要です。
- **UTILITY_DB** – `utility_db` データベースに接続できるユーザのみがこの文を発行できます。

サーバを実行中のアカウントには、ファイルが作成されたディレクトリの書き込みパーミッションが必要です。

参照：

- **CREATE DBSPACE** 文 (150 ページ)

- DROP DATABASE 文 (325 ページ)

CREATE DBSPACE 文

IQ メインストア、キャッシュ DB 領域、カタログストア、または RLV ストアに使用する新しい DB 領域および関連する DB ファイルを作成します。

クイックリンク：

「パラメータ」 (151 ページ)

「例」 (152 ページ)

「使用法」 (153 ページ)

「標準」 (154 ページ)

「パーミッション」 (154 ページ)

構文

構文 1 - カタログストア DB 領域のみ (SQL Anywhere (SA) DB 領域) に使用します。

```
CREATE DBSPACE dbspace-name AS file-path CATALOG STORE
```

構文 2 - IQ メインストア DB 領域に使用します。

```
CREATE DBSPACE dbspace-name USING file-specification  
[ IQ STORE ] iq-dbspace-opts
```

構文 3 - RLV DB 領域に使用します。

```
CREATE DBSPACE dbspace-name USING file-specification  
IQ RLV STORE
```

構文 4 - キャッシュ DB 領域に使用します。

```
CREATE DBSPACE dbspace-name USING FILE file-specification  
IQ CACHE STORE
```

file-specification - (back to Syntax 2) or (back to Syntax 3)
{ **single-path-spec** | **new-file-spec** [, ...] }

single-path-spec - (back to file-specification)
'*file-path*' | **iq-file-opts**

new-file-spec - (back to file-specification)
FILE *logical-file-name* | '*file-path*' **iq-file-opts**

iq-file-opts - (back to new-file-spec)
[[**SIZE**] *file-size*]
...[**KB** | **MB** | **GB** | **TB**]]

```
[ RESERVE size
...[ KB | MB | GB | TB ] ]
iq-dbspace-opts - (back to Syntax 2)
[ STRIPING ] {ON | OFF} ] ...[ STRIPESIZEKB sizeKB ]
```

パラメータ

(先頭に戻る) (150 ページ)

- **new-file-spec** – IQ メインストアに使用する DB 領域を作成します。IQ メインストアでは、1つまたは複数の DB ファイルを指定できます。DB ファイル名と物理ファイルパスはファイルごとに必要で、それぞれユニークでなければなりません。
- **RESERVE** – 将来、DB 領域のサイズを増加できるように、予約領域のサイズをキロバイト (KB)、メガバイト (MB)、ギガバイト (GB)、テラバイト (TB) の単位で指定します。size パラメータには、0 より大きい任意の値を指定できます。デフォルトはメガバイトです。DB 領域の DB ファイルの作成後は、この予約サイズを変更できません。

RESERVE を指定すると、データベースの内部 (フリーリスト) 構造により多くの領域が使用されるようになります。予約サイズが大きすぎると、内部構造体で必要とする領域が、指定されたサイズよりも大きくなり、エラーになります。

- **dbspace-name** と **dbfile-name** – DB 領域と DB ファイルの内部名。データベースは、データベース作成時に作成した最初の DB 領域を含め、(32KB - 1) までの DB 領域を持つことが可能です。ただし、使用しているオペレーティングシステムで、データベースごとの DB ファイル数が制限されることもあります。

注意： 複数のユーザ DB 領域を作成するには、IQ_VLDBMGMT オプションのライセンスが必要です。

- **file-path** – DB ファイルの実際のオペレーティングシステムファイル名であり、必要に応じてその前にパスを付けます。明示的なディレクトリ指定がない *file-path* は、データベースのカタログストアと同じディレクトリに作成されます。相対ディレクトリはカタログストアに対して相対となります。
- **SIZE** – オペレーティングシステムファイルの場合、*file-path* で指定したファイルのサイズを 0 から 4 テラバイトまでの間で指定します。デフォルトはストアタイプとブロックサイズに応じて異なります。IQ メインストアの場合、デフォルトのバイト数は 1000 * ブロックサイズになります。カタログストアに SIZE 句を指定することはできません。SIZE に 0 の値を指定すると、最小サイズの DB 領域が作成されます。これは、IQ メインストアでは 8MB です。

ローパーティションの場合は、SIZE を明示的に指定しないでください。SAP Sybase IQ は自動的にこのパラメータを最大ローパーティションサイズに設定し、それ以外のサイズが指定されるとエラーを返します。

- **STRIPESIZEKB** – ディスクストライピングアルゴリズムが指定した DB 領域の次のストライプに移動する前に、各ファイルに書き込むデータ量をキロバイト (KB) で指定します。

ストライピングまたはストライプサイズを指定しない場合は、DEFAULT_DISK_STRIPING オプションと DEFAULT_KB_PER_STRIPE オプションのデフォルト値が適用されます。

例

(先頭に戻る) (150 ページ)

- **例 1** – UNIX システム上で 2 つの DB ファイルとともに DspHist という名前の DB 領域を IQ メインストアに作成します。各 DB ファイルのサイズは 1GB で、500MB までの拡張が可能です。

```
CREATE DBSPACE DspHist USING FILE
FileHist1 '/History1/data/file1'
SIZE 1000 RESERVE 500,
FILE FileHist2 '/History1/data/file2'
SIZE 1000 RESERVE 500;
```

- **例 2** – DspCat2 という名前の 2 番目のカタログ DB 領域を作成します。

```
CREATE DBSPACE DspCat2 AS
'catalog_file2'
CATALOG STORE;
```

- **例 3** – EmpStore1 という名前の IQ メイン DB 領域を IQ ストアに作成します (3 種類の構文を例示しています)。

```
CREATE DBSPACE EmpStore1
USING FILE EmpStore1
'EmpStore1.IQ' SIZE 8 MB IQ STORE;
```

```
CREATE DBSPACE EmpStore1
USING FILE EmpStore1
'EmpStore1.IQ' 8 IQ STORE;
```

```
CREATE DBSPACE EmpStore1
USING FILE EmpStore1
'EmpStore1.IQ' 8;
```

- **例 4** – d1 と呼ばれる RLV ストア DB 領域を作成します。

```
CREATE DBSPACE d1
USING FILE f1
'f1.iq' SIZE 1000 IQ RLV STORE;
```

- **例 5** – 200GB の DB ファイルが含まれる myDAS という名前のキャッシュ DV 領域を作成します。

```
CREATE DBSPACE myDAS
USING FILE iqdas1
'iqdas1.iq' SIZE 200 GB IQ CACHE STORE
```

使用法

(先頭に戻る) (150 ページ)

CREATE DBSPACE は、IQ メインストア、キャッシュ DB 領域、カタログストア、または RLV ストアに新しい DB 領域を作成します。追加する DB 領域を最初の DB 領域とは別のディスクデバイス上に置き、1 台の物理デバイスの容量を超えるストアを作成することも可能です。

構文 1 は、カタログストアに DB 領域を作成します。DB 領域および DB ファイルの論理名は同じです。カタログストアにある各 DB 領域には、ファイルは 1 つしかありません。

DB 領域名と DB ファイル名では、常に大文字と小文字が区別されません。物理ファイルパスについては、データベースが CASE RESPECT の場合はオペレーティングシステムに応じて大文字と小文字が区別され、CASE IGNORE の場合は区別されません。

IQ テンポラリストアに DB 領域を作成することはできません。新しいデータベースの作成時、または SAP Sybase IQ 15.3 より前のバージョンで作成されたデータベースをアップグレードする場合は、IQ_SYSTEM_TEMP という名前のテンポラリー DB 領域が 1 つ作成されます。**ALTER DBSPACE ADD FILE** 構文を使用すると、IQ_SYSTEM_TEMP DB 領域にファイルを追加できます。

注意： RLV ストレージでは、前提条件として 1 つ以上のファイルが含まれる RLV DB 領域を作成する必要があります。シンプレックスサーバで RLV ストレージを有効にする前に、RLV DB 領域が存在することを確認します。

シンプレックスノードまたはマルチプレックスノードに作成できるキャッシュ DV 領域は 1 つのみです。セカンドキャッシュ DB 領域を作成しようとするエラーが発生します。

次のいずれかの方法で、ユニークなパスを作成できます。

- ファイルごとに異なる拡張子を指定する (mydb.iq など)。
- 異なるファイル名を指定する (mydb2.iq など)。
- 異なるパス名 (/iqfiles/main/iq など) または異なるローパーティションを指定する。

警告！ UNIX プラットフォームでは、データベースの一貫性を保つために、別のファイルにリンクするファイル名を指定してください。SAP Sybase IQ は、リンク

SQL 文

されたファイルが指すターゲットを検出できません。コマンド内でファイル名が異なっても、同じオペレーティングシステムファイルをポイントしていないか確認してください。

関連する動作：

- オートコミット
- 自動チェックポイント

標準

(先頭に戻る) (150 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Adaptive Server ではサポートされていません。

パーミッション

(先頭に戻る) (150 ページ)

MANAGE ANY DBSPACE システム権限が必要です。

参照：

- CREATE DATABASE 文 (137 ページ)
- DROP 文 (319 ページ)

CREATE DOMAIN 文

データベースにユーザ定義データ型を作成します。

クイックリンク：

「パラメータ」 (155 ページ)

「例」 (155 ページ)

「使用法」 (155 ページ)

「標準」 (156 ページ)

「パーミッション」 (156 ページ)

構文

```
CREATE { DOMAIN | DATATYPE } domain-name data-type
... [ NOT ] NULL ]
... [ DEFAULT default-value ]
```

```

default-value - (構文に戻る)
special-value
| string
| global variable
| [ - ] number
| ( constant-expression )
| built-in-function( constant-expression )
| AUTOINCREMENT
| CURRENT DATABASE
| CURRENT REMOTE USER
| NULL
| TIMESTAMP
| LAST USER

special-value (back to default-value)
CURRENT
{ DATE
| TIME
| TIMESTAMP
| USER
| PUBLISHER }
| USER

```

パラメータ

(先頭に戻る) (154 ページ)

- **data-type** – 精度と位取りを指定した組み込みデータ型

例

(先頭に戻る) (154 ページ)

- **例 1** – 35 文字の文字列を格納し、NULL が使用できる `address` という名前のデータ型を作成します。

```
CREATE DOMAIN address CHAR( 35 ) NULL
```

使用法

(先頭に戻る) (154 ページ)

ユーザ定義データ型は、必要に応じて精度と位取りを含めた組み込みデータ型のエイリアスです。データベース内の使いやすさを改善し、一貫性を高めます。

注意： **CREATE DOMAIN** は ANSI/ISO SQL3 用語なので、**CREATE DATATYPE** ではなく、**CREATE DOMAIN** を使用してください。

データ型を作成するユーザは、自動的にそのデータ型の所有者となります。**CREATE DATATYPE** 文の中では、所有者を指定できません。ユーザ定義データ型の名前はユニークでなければなりません。また、プレフィクスとして所有者を使用しなくても、すべてのユーザがデータ型にアクセスできます。

ユーザ定義データ型は、データベースの中のオブジェクトです。識別子のルールに従って名前を付ける必要があります。ユーザ定義データ型の名前では、組み込みデータ型の名前と同様に、常に大文字と小文字が区別されません。

ユーザ定義データ型では、デフォルトで NULL を使用できます。ただし、**allow_nulls_by_default** データベースオプションが OFF に設定されている場合は除きます。この場合、新しいユーザ定義データ型では、デフォルトで NULL を使用できません。ユーザ定義データ型で作成したカラムで NULL を使用できるかどうかは、そのカラムを参照するときの **allow_nulls_by_default** オプションの設定ではなく、ユーザ定義データ型の定義の設定によって決まります。カラム定義で NULL または NOT NULL を明示的に設定すると、ユーザ定義データ型の設定より優先されます。

CREATE DOMAIN 文を使用すると、ユーザ定義データ型の DEFAULT 値を指定できます。DEFAULT 値の指定は、そのデータ型で定義されたすべてのカラムに継承されます。カラムに対して明示的に指定した DEFAULT 値は、データ型に対して指定したものより優先されます。

CREATE DOMAIN 文を使用して、CHECK 条件と呼ばれるルールをユーザ定義データ型の定義に組み込むことができます。

SAP Sybase IQ では、ベーステーブル、グローバルテンポラリテーブル、ローカルテンポラリテーブル、およびユーザ定義データ型に CHECK 制約が適用されます。

データベースからデータ型を削除するには **DROP** 文を使用します。ユーザ定義データ型を削除するには、そのデータ型の所有者であるか、**CREATE DATATYPE** または **CREATE ANY OBJECT** システム権限を持っている必要があります。

関連する動作：

- オートコミット

標準

(先頭に戻る) (154 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - Adaptive Server ではサポートされていません。Transact-SQL は **sp_addtype** システムプロシージャ、**CREATE DEFAULT** 文、**CREATE RULE** 文を使用して、同様の機能を提供しています。

パーミッション

(先頭に戻る) (154 ページ)

次のいずれかが必要です。

- CREATE DATATYPE システム権限
- CREATE ANY OBJECT システム権限

参照：

- DROP 文 (319 ページ)

CREATE EVENT 文

イベントとそのハンドラ (事前定義のアクションを自動化する) を定義します。また、スケジュールされたアクションも定義します。

クイックリンク：

「パラメータ」 (158 ページ)

「例」 (161 ページ)

「使用法」 (162 ページ)

「標準」 (162 ページ)

「パーミッション」 (163 ページ)

構文

```
CREATE EVENT event-name
  [ TYPE event-type
    [ WHERE trigger-condition [ AND trigger-condition ], ... ]
    | SCHEDULE schedule-spec, ... ]
  ...[ ENABLE | DISABLE ]
  ...[ AT { CONSOLIDATED | REMOTE | ALL } ]
  ...[ HANDLER
    BEGIN
  ...
    END ]
```

event-type - (構文に戻る)

```
BackupEnd
| "Connect"
| ConnectFailed
| DatabaseStart
| DBDiskSpace
| "Disconnect"
| GlobalAutoincrement
| GrowDB
| GrowLog
| GrowTemp
| IQMainDBSpaceFree
| IQTempDBSpaceFree
| LogDiskSpace
| "RAISERROR"
```

```

| ServerIdle
| TempDiskSpace

trigger-condition - (構文に戻る)
  event_condition( condition-name )
  { =
  | <
  | >
  | !=
  | <=
  | >= } value

schedule-spec - (構文に戻る)
[ schedule-name ]
{ START TIME start-time | BETWEEN start-time AND end-time }
[ EVERY period { HOURS | MINUTES | SECONDS } ]
[ ON { ( day-of-week, ... ) | ( day-of-month, ... ) } ]
[ START DATE start-date ]

```

パラメータ

(先頭に戻る) (157 ページ)

- **event-name** – イベントには作成者が関連付けられます。これはイベントを作成したユーザであり、イベントハンドラはその作成者のパーミッションで実行されます。これはストアプロシージャの実行と同じです。他のユーザが所有するイベントを作成することはできません。イベント名は、システムテーブル SYSEVENT に対するクエリを実行してリストできます。次に例を示します。

```
SELECT event_id, event_name FROM SYS.SYSEVENT
```

- **event-type** – システム定義のイベントタイプのうちのいずれか。イベントタイプの大文字と小文字は区別されません。この *event-type* がイベントをトリガする条件を指定するには、WHERE 句を使用します。
- **DiskSpace** - データベースに DiskSpace タイプのいずれかに対応するイベントハンドラが存在する場合、データベースサーバは、関係ファイルに関連付けられた各デバイスの空き領域を 30 秒ごとにチェックします。
データベースが別々のドライブに複数の DB 領域を持っている場合、DBDiskSpace は各ドライブをチェックし、その中で最小の空き領域に基づいて動作します。
- **LogDiskSpace** - トランザクションログとミラーリングトランザクションログのロケーションをチェックし、最小の空き領域に基づいてレポートします。
- **Globalautoincrement** - テーブルの GLOBAL AUTOINCREMENT のデフォルト値が、その範囲の限界の 1 パーセント以内になった場合に起動されます。このハンドラのアクションでは一般に、GLOBAL_DATABASE_ID 句の新しい値が要求されます。

EVENT_CONDITION 関数と **RemainingValues** をこのイベントタイプの引数として使用できます。

- **ServerIdle** - データベースに **ServerIdle** タイプのイベントハンドラがある場合、サーバは 30 秒ごとにサーバのアクティビティをチェックします。
- **WHERE** 句 - トリガ条件は、イベントが起動する条件を決定します。たとえば、トランザクションログを格納するディスクの使用量が 80% を超えたときにアクションを起こす場合は、次のトリガ条件を使用します。

```
...
WHERE event_condition( 'LogDiskSpacePercentFree' ) < 20
...
```

EVENT_CONDITION 関数には、イベントタイプに有効な引数を指定してください。複数の **AND** 条件を使って **WHERE** 句を構成できますが、**OR** 条件やその他の条件は使用できません。

- **SCHEDULE** - スケジュールされたアクションをいつ実行するかを指定します。時刻のシーケンスは、イベントハンドラに定義された関連するアクションのトリガ条件セットとして機能します。1つのイベントとそのハンドラに対し、複数のスケジュールを作成できます。これにより、複雑なスケジュールの実装が可能になります。複数のスケジュールがある場合はスケジュール名を必ず指定しなければなりません、スケジュールが1つしかない場合はスケジュール名の指定を省略できます。

スケジュール名は、システムテーブル **SYSSCHEDULE** に対するクエリを実行することでリストできます。次に例を示します。

```
SELECT event_id, sched_name FROM SYS.SYSSCHEDULE
```

各イベントにはユニークなイベント ID があります。イベントと関連するスケジュールの対応付けには、**SYSEVENT** と **SYSSCHEDULE** の **event_id** カラムを使用します。

1 回限りの実行としてスケジュールされたイベントが渡された場合、そのスケジュールは削除されますが、イベントハンドラは削除されません。

スケジュールされたイベントの時刻は、スケジュールの作成時に計算され、イベントハンドラの実行が完了したときに再計算されます。次のイベント時刻を計算するときには、イベントのスケジュールが調べられ、将来の次回スケジュール時刻が決定されます。9:00 から 5:00 の間に 1 時間ごとに実行され、実行に 65 分を要するイベントハンドラは、9:00、11:00、1:00、3:00、5:00 に実行されます。実行を重複させる場合は、複数のイベントを作成する必要があります。

スケジュール定義のサブ句は、次のとおりです。

- **START DATE** - スケジュールされたイベントが開始される日付。デフォルトは現在の日付です。
- **START TIME** - イベントがスケジュールされた各日の最初のスケジュール時刻。START DATE を指定した場合、START TIME はその日付を参照します。START DATE を指定しない場合、START TIME の対象となる日付は、現在の日付 (指定された時刻を過ぎていない場合) とそれ以降の各日となります。
- **BETWEEN ... AND** - 1 日のうち、スケジュールされた時刻が発生する時間範囲。START DATE を指定した場合、スケジュールされた時刻はその日が来るまで発生しません。
- **EVERY** - 連続してスケジュールされたイベントの発生間隔。スケジュールされたイベントは、その日の START TIME より後、または BETWEEN ... AND で指定された範囲内でのみ発生します。
- **ON** - スケジュールされたイベントが発生する日のリスト。デフォルトは毎日です。曜日または日付で指定できます。曜日は Monday、Tuesday などのように指定します。Mon、Tue などのように省略形を使用することもできます。データベースサーバは SAP Sybase IQ でサポートされるすべての言語で、表記された日付を正式名と省略形の両方で認識します。日付は、0 ~ 31 の整数で指定します。0 は月の末日を表します。

スケジュールされたイベントハンドラが完了するたびに、次のスケジュール時刻と日付が計算されます。

- **EVERY** 句を使用した場合は、次のスケジュール時刻が現在の日付にあり、BETWEEN ... AND で指定された範囲の終わりより前であるかどうかを調べます。そうであれば、これが次のスケジュール時刻となります。
- 次のスケジュール時刻が現在の日付にない場合は、イベントが実行される次の日付を調べます。
- その日付の START TIME、または BETWEEN ... AND で指定された範囲の開始を確認します。
- **ENABLE | DISABLE** - イベントハンドラはデフォルトで有効になっています。DISABLE を指定すると、スケジュール時刻に達したりトリガ条件が発生しても、イベントハンドラは起動しません。TRIGGER EVENT 文は、無効に設定されているイベントハンドラを起動しません。
- **AT** - SQLRemote の設定時にリモートのデータベースまたは統合データベースでイベントを実行するには、この句を使用してイベントを処理するデータベースを制限します。デフォルトでは、すべてのデータベースがイベントを実行します。
- **HANDLER** - イベントごとに 1 つのハンドラが存在します。ストアードプロシージャの本体と同様、ハンドラも複合文です。ただし、いくつか相違点があります。複合文内で EXCEPTION 句を使用してエラーを処理できます。ただし、ス

トアドプロシージャ内で使用可能な ON EXCEPTION RESUME 句は使用できません。

例

(先頭に戻る) (157 ページ)

- **例 1** – データベースサーバに対して、毎日 1 a.m. に自動インクリメンタルバックアップを実行するように指示します。

```
CREATE EVENT IncrementalBackup
SCHEDULE
START TIME '1:00AM' EVERY 24 HOURS
HANDLER
  BEGIN
    BACKUP DATABASE INCREMENTAL
    TO 'backups/daily.incr'
  END
```

- **例 2** – データベースサーバに対して、システムストアドプロシージャ **sp_iqspaceused** を 10 minutes ごとに呼び出し、その戻り値である現在の日付と時刻、データベースへの現在の接続数、およびメイン IQ ストアとテンポラリ IQ ストアの使用に関する現在の情報をテーブルに格納するように指示します。

```
CREATE TABLE mysummary(dt DATETIME,
  users INT, mainKB UNSIGNED BIGINT,
  mainPC UNSIGNED INT,
  tempKB UNSIGNED BIGINT,
  tempPC UNSIGNED INT) ;

CREATE EVENT mysummary
SCHEDULE sched_mysummary
START TIME '00:01 AM' EVERY 10 MINUTES
HANDLER
BEGIN
  DECLARE mt UNSIGNED BIGINT;
  DECLARE mu UNSIGNED BIGINT;
  DECLARE tt UNSIGNED BIGINT;
  DECLARE tu UNSIGNED BIGINT;
  DECLARE conncount UNSIGNED INT;

  SET conncount = DB_PROPERTY('ConnCount');
  CALL SP_IQSPACEUSED(mt,mu,tt,tu);

  INSERT INTO mysummary VALUES( NOW(),
  conncount, mu, (mu*100)/mt, tu,
  (tu*100)/tt );
END;
```

- **例 3** – トランザクションログファイルが格納されているデバイス上の空きディスク領域が 30% 未満になった場合に、サーバログにメッセージを記録します。ただし、ハンドラが実行されるのは、300 秒に 1 回だけです。

```
CREATE EVENT LowTxnLogDiskSpace
TYPE DBDiskSpace
WHERE event_condition( 'DBFreePercent' ) < 30
AND event_condition( 'Interval' ) >= 300
HANDLER
BEGIN
message 'Disk space for Transaction Log is low.';
END;
```

使用法

(先頭に戻る) (157 ページ)

イベント定義は2つの部分からなります。トリガ条件とは、ディスク領域の使用量が指定のスレッシュホルドを超えるなどの出来事に相当します。スケジュールとは、時刻のセットのことで、それぞれの時刻がトリガ条件の役割を果たします。トリガ条件が満たされると、イベントハンドラが実行されます。イベントハンドラには、複合文 (**BEGIN... END**) 内で指定された1つ以上のアクションが含まれています。

トリガ条件やスケジュールを指定しない場合は、明示的な **TRIGGER EVENT** 文だけがイベントをトリガします。開発時には **TRIGGER EVENT** を使用してイベントハンドラを開発およびテストし、テストが完了してからスケジュールや **WHERE** 句を追加することもできます。

イベントエラーはデータベースサーバコンソールに記録されます。

イベントハンドラがトリガされると、サーバはイベントをトリガさせた接続 ID などのコンテキスト情報を作成して、イベントハンドラが **EVENT_PARAMETER** 関数を使用する際に利用できるようにします。

注意： イベント内では結果セットを返す文を使用することはできませんが、イベントがストアードプロシージャを呼び出してその結果をテンポラリテーブルに挿入できるようにすることはできます。

関連する動作：

- オートコミット
- イベントハンドラの動作は、実行中にエラーが検出されなければコミットされ、エラーが検出された場合はロールバックされます。

標準

(先頭に戻る) (157 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Adaptive Server ではサポートされていません。

パーミッション

(先頭に戻る) (157 ページ)

次のいずれかが必要です。

- `MANAGE ANY EVENT` システム権限
- `CREATE ANY OBJECT` システム権限

イベントハンドラは、イベント所有者の権限に基づいて別の接続上で実行されません。`MANAGE ANY EVENT` システム権限以外の権限で実行するには、イベントハンドラの内部からプロシージャを呼び出します。この場合、プロシージャはその所有者のパーミッションで実行されます。

参照：

- `ALTER EVENT` 文 (20 ページ)
- `BEGIN ... END` 文 (108 ページ)
- `COMMENT` 文 (123 ページ)
- `DROP` 文 (319 ページ)
- `TRIGGER EVENT` 文 (560 ページ)

CREATE EXISTING TABLE 文

リモートサーバ上の既存のテーブルを表す新しいプロキシテーブルを作成します。

クイックリンク：

「パラメータ」 (164 ページ)

「例」 (165 ページ)

「使用法」 (165 ページ)

「標準」 (166 ページ)

「パーミッション」 (166 ページ)

構文

```
CREATE EXISTING TABLE [owner.]table_name
  [ ( column-definition, ... ) ]
  AT 'location-string'
```

column-definition - (構文に戻る)
column-name *data-type* [**NOT NULL**]

location-string - (構文に戻る)

```
remote-server-name.[db-name].[owner].object-name | remote-server-
name; [db-name]; [owner]; object-name
```

パラメータ

(先頭に戻る) (163 ページ)

- column-definition** – カラム定義を指定しない場合、SAP Sybase IQ は、リモートテーブルから取得するメタデータからカラムリストを導出します。カラム定義を指定した場合、SAP Sybase IQ は、そのカラム定義を検証します。SAP Sybase IQ では、カラム名、データ型、長さ、NULL プロパティについて、次の点をチェックします。
 - カラム名が一致しなければなりません (大文字小文字は無視されます)。
 - CREATE EXISTING TABLE** 文のデータ型は、リモートロケーションのカラムのデータ型と一致するか、またはそのデータ型に変換可能でなければなりません。たとえば、ローカルカラムのデータ型が NUMERIC として定義されているのに対して、リモートカラムのデータ型が MONEY である場合があります。データ型が一致しないテーブル、またはその他の不整合が存在するテーブルから選択した場合、エラーが発生する可能性があります。
 - 各カラムの NULL プロパティがチェックされます。ローカルカラムの NULL プロパティがリモートカラムの NULL プロパティと同じでない場合、警告メッセージが出力されますが、文はアボートしません。
 - 各カラムの長さがチェックされます。CHAR、VARCHAR、BINARY、DECIMAL、NUMERIC の各カラムの長さが一致しない場合は、警告メッセージが出力されますが、コマンドはアボートしません。**CREATE EXISTING** 文には、実際のリモートカラムリストのサブセットだけをインクルードできます。
- AT** – リモートオブジェクトのロケーションを指定します。AT 句は、デリミタとしてセミコロン (;) をサポートします。セミコロンがロケーション文字列のどこかにある場合、そのセミコロンはフィールドデリミタです。セミコロンがない場合は、ピリオドがフィールドデリミタです。ピリオドを使用すると、データベースおよび所有者の各フィールドにファイル名と拡張子を使用できます。セミコロンのフィールドデリミタは、現在サポートされていないサーバクラスで主に使用されていますが、ピリオドもフィールドデリミタとして機能する状況ではセミコロンも使用できます。

たとえば、次の文は、テーブル proxy_a1 をリモートサーバ myasa の SQL Anywhere データベース mydb にマッピングします。

```
CREATE EXISTING TABLE
proxy_a1
AT 'myasa;mydb;;a1'
```


例

(先頭に戻る) (163 ページ)

- **例 1** – リモートサーバ `server_a` にある `nation` テーブルのプロキシテーブル `nation` を作成します。

```
CREATE EXISTING TABLE nation
( n_nationkey int,
  n_name char(25),
  n_regionkey int,
  n_comment char(152))
AT 'server_a.db1.joe.nation'
```

- **例 2** – リモートサーバ `server_a` にある `blurbs` テーブルのプロキシテーブル `blurbs` を作成します。SAP Sybase IQ は、リモートテーブルから取得したメタデータからカラムリストを導出します。

```
CREATE EXISTING TABLE blurbs
AT 'server_a.db1.joe.blurbs'
```

- **例 3** – SAP Sybase IQ リモートサーバ `remote_iqdemo_srv` にある `Employees` テーブルのプロキシテーブル `rda_employee` を作成します。

```
CREATE EXISTING TABLE rda_employee
AT 'remote_iqdemo_srv..dba.Employees'
```

使用法

(先頭に戻る) (163 ページ)

CREATE EXISTING TABLE は **CREATE TABLE** 文の変形です。EXISTING キーワードを **CREATE TABLE** で使用すると、テーブルがすでにリモートに存在していて、そのメタデータが SAP Sybase IQ にインポートされるように指定できます。これにより、リモートテーブルはユーザに対して可視なエンティティとして作成されます。SAP Sybase IQ はテーブルを作成する前に、テーブルが外部の場所に存在することを確認します。

プロキシテーブルとして使用するテーブルに、30 文字よりも長い名前を付けることはできません。

オブジェクトが (ホストデータファイルまたはリモートサーバオブジェクトのどちらかとして) 存在しない場合、この文は拒否されてエラーメッセージが出力されません。

ホストデータファイルまたはリモートサーバテーブルのインデックス情報が抽出されて、システムテーブル `sysindexes` のローを作成するために使用されます。これにより、サーバ用語のインデックスとキーが定義されて、クエリオプティマイザがこのテーブルに存在する可能性のあるすべてのインデックスを考慮できるようになります。

参照整合性制約は、必要に応じてリモートロケーションに渡されます。

シンプレックス環境では、同じノード上でリモートテーブルを参照するプロキシテーブルを作成することはできません。マルチプレックス環境では、マルチプレックス内で定義されたリモートテーブルを参照するプロキシテーブルを作成することはできません。

たとえば、シンプレックス環境で、同じノード上で定義されたベーステーブル `Employees` を参照するプロキシテーブル `proxy_e` を作成しようとする、**CREATE EXISTING TABLE** 文が拒否され、エラーメッセージが表示されます。マルチプレックス環境では、マルチプレックス内で定義されたリモートテーブル `Employees` を参照する任意のノード (コーディネータまたはセカンダリ) からプロキシテーブル `proxy_e` を作成する場合に、**CREATE EXISTING TABLE** 文が拒否されます。

標準

(先頭に戻る) (163 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - Open Client/Open Server でサポートされています。

パーミッション

(先頭に戻る) (163 ページ)

自分が所有するテーブルの場合、次のいずれかが必要です。

- CREATE ANY TABLE システム権限
- CREATE ANY OBJECT システム権限

任意のユーザが所有するテーブルの場合、CREATE ANY TABLE システム権限が必要です。

参照：

- CREATE TABLE 文 (257 ページ)

CREATE EXTERNLOGIN 文

リモートサーバとの通信に使用する代替ログイン名とパスワードを割り当てます。

クイックリンク：

「パラメータ」 (167 ページ)

「例」 (167 ページ)

「使用法」 (167 ページ)

「標準」 (168 ページ)

「パーミッション」 (168 ページ)

構文

```
CREATE EXTERNLOGIN login-name
TO remote-server
REMOTE LOGIN remote-user
[ IDENTIFIED BY remote-password ]
```

パラメータ

(先頭に戻る) (166 ページ)

- **login-name** – ローカルユーザログイン名を指定します。統合化ログインを使用する場合、*login-name*は Windows ユーザ ID のマッピング先となるデータベースユーザです。
- **TO** – リモートサーバ名を指定します。
- **REMOTE LOGIN** – ローカルユーザ *login-name* に対して、*remote-server* 上のユーザアカウントを指定します。
- **IDENTIFIED BY** – *remote-password* が *remote-user* のパスワードになるように指定します。IDENTIFIED BY 句を省略すると、NULL のパスワードがリモートサーバに送信されます。IDENTIFIED BY "" (空の文字列) を指定すると、空の文字列がパスワードとして送信されます。

例

(先頭に戻る) (166 ページ)

- **例 1** – サーバ *sybase1* に接続するときに、ローカルユーザ *dba* をパスワード *4TKNOX* が設定されたユーザ *sa* にマッピングします。

```
CREATE EXTERNLOGIN dba
TO sybase1
REMOTE LOGIN sa
IDENTIFIED BY 4TKNOX
```

使用法

(先頭に戻る) (166 ページ)

CREATE EXTERNLOGIN による変更は、次にリモートサーバに接続するまで有効になりません。

デフォルトでは、SAP Sybase IQ は、クライアントに代わってリモートサーバに接続するときは、常にそのクライアントの名前とパスワードを使用します。**CREATE EXTERNLOGIN** は、リモートサーバとの通信に使用される代替ログイン名とパスワードを割り当てます。パスワードは暗号形式で内部に格納されます。

remote_server をローカルサーバに認識させるには、ISYSSERVER システムテーブルのエントリを指定します。詳細については、「CREATE SERVER 文」を参照してください。

CREATE EXTERNLOGIN 文でリモートログインを作成し、**CREATE SERVER** 文でリモートサーバを定義すると、ユーザがどのような状況でもログインとパスワードを使用できるように、INSERT...LOCATION の外部ログインとパスワードが設定されます。これによって、ログインまたはパスワードによるアクセス不能が原因で発生する可能性のあるエラーを回避されます。リモートサーバへの接続はこの方法で行うことをおすすめします。

注意：現在の接続のユーザ ID とパスワードを使用しているときに、ユーザがパスワードを変更した場合、サーバを停止して再起動するまで、新しいパスワードはリモートサーバ上で有効になりません。**CREATE EXTERNLOGIN** で作成されたりリモートログインは、デフォルトユーザ ID のパスワード変更の影響を受けません。

パスワードの自動有効期限を使用するサイトでは、外部ログインのために、定期的なパスワードの更新を計画する必要があります。

CREATE EXTERNLOGIN は、トランザクション内では使用できません。

remote-user と *remote-password* の組み合わせは、*remote-server* 上で有効でなければなりません。

関連する動作：

- オートコミット。

標準

(先頭に戻る) (166 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - Open Client/Open Server でサポートされています。

パーミッション

(先頭に戻る) (166 ページ)

MANAGE ANY USER システム権限が必要です。

参照：

- DROP EXTERNLOGIN 文 (327 ページ)

- INSERT 文 (412 ページ)
- CREATE SERVER 文 (241 ページ)

CREATE FUNCTION 文

データベースにユーザ定義関数を作成します。所有者名を指定することにより、別のユーザが使用する関数を作成できます。パーミッションがあれば、ユーザ定義関数も他の非集合関数とまったく同じように使用できます。

クイックリンク：

「パラメータ」 (170 ページ)

「例」 (173 ページ)

「使用法」 (174 ページ)

「標準」 (175 ページ)

「パーミッション」 (175 ページ)

構文

構文 1

```
CREATE [ OR REPLACE ] [ TEMPORARY ] FUNCTION [ owner.]function-name
( [ parameter, ... ] )
  [ SQL SECURITY { INVOKER | DEFINER } ]
  RETURNS data-type ON EXCEPTION RESUME
  | [ NOT ] DETERMINISTIC
  { compound-statement | AS tsq-compound-statement
  | EXTERNAL NAME library-call
  | EXTERNAL NAME java-call LANGUAGE JAVA }
```

構文 2

```
CREATE FUNCTION [ owner.]function-name ( [ parameter, ... ] )
  RETURNS data-type
  URL url-string
  [ HEADER header-string ]
  [ SOAPHEADER soap-header-string ]
  [ TYPE { 'HTTP[:{ GET | POST }]' | 'SOAP[:{ RPC | DOC }]' } ]
  [ NAMESPACE namespace-string ]
  [ CERTIFICATE certificate-string ]
  [ CLIENTPORT clientport-string ]
  [ PROXY proxy-string ]
```

parameter - (back to Syntax 1) or (back to Syntax 2)
IN parameter-name data-type [**DEFAULT** expression]

tsq-compound-statement - (back to Syntax 1)
 sql-statement
 sql-statement ...

```

library-call - (back to Syntax 1)
  '[ operating-system : ] function-name@library; ...'

operating-system - (back to library-call)
  UNIX

java-call - (back to Syntax 1)
  '[ package-name. ] class-name.method-name method-signature '

method-signature - (back to java-call)
  ( [ field-descriptor, ... ] ) return-descriptor

field-descriptor and return-descriptor - (back to method-signature)
  Z | B | S | I | J | F | D | C | V | [descriptor | L class-name;

url-string - (back to Syntax 2)
  ' { HTTP | HTTPS | HTTPS_FIPS } ://[user:password@]hostname[:port] [/
  path ] '

```

パラメータ

(先頭に戻る) (169 ページ)

- **CREATE [OR REPLACE]** – パラメータ名は、データベース識別子に関するルールに従って付ける必要があります。また、有効な SQL データ型でなければなりません。さらに、この引数が関数に値を渡す式であることを示すキーワード **IN** を先頭に付ける必要があります。

CREATE 句を指定すると新しい関数が作成され、**OR REPLACE** 句を使用すると既存の関数が同じ名前でも置き換えられます。関数が置き換えられた場合、関数の定義は変更されますが、既存のパーミッションは保持されます。**OR REPLACE** 句をテンポラリ関数で使用することはできません。

- **TEMPORARY** – 作成した接続でのみ参照できる関数になり、接続を削除すると関数も自動的に削除されます。テンポラリ関数を明示的に削除することもできます。テンポラリ関数に対して **ALTER**、**GRANT**、および **REVOKE** 操作を実行することはできません。また他の関数とは異なり、テンポラリ関数はカタログやトランザクションログに記録されません。

テンポラリ関数は、作成者 (現在のユーザ) のパーミッションで実行されます。また作成者のみが所有できます。そのため、テンポラリ関数を作成するときは所有者を指定しないでください。読み込み専用のデータベースに接続するときに、テンポラリ関数の作成と削除を行うことができます。

- **SQL SECURITY – INVOKER** (関数を呼び出しているユーザ) または **DEFINER** (関数を所有しているユーザ) のどちらとして関数が実行されるかを定義します。デフォルトは **DEFINER** です。

INVOKER を指定すると、プロシージャを呼び出す各ユーザに対して注釈を付ける必要があるため、より多くのメモリが使用されます。また、名前解決も

INVOKER として実行されます。そのため、適切な所有者で、すべてのオブジェクト名 (テーブル、プロシージャなど) を修飾するように注意してください。

- **data-type** – LONG BINARY と LONG VARCHAR を戻り値のデータ型として使用することはできません。
- **compound-statement** – BEGIN と END で囲まれ、セミコロンで区切られた SQL 文のセットです。「BEGIN ... END 文」を参照してください。
- **tsql-compound-statement** – Transact-SQL 文のバッチです。
- **external-name** – 外部ライブラリにある関数への呼び出しのラッパーです。RETURNS 句の後に他の句を持つことはできません。ライブラリ名にはファイル拡張子が付く場合があります。この拡張子は通常、Windows では .dll、UNIX では .so です。拡張子がない場合、ライブラリに対するプラットフォーム固有のデフォルトのファイル拡張子が追加されます。

external-name 句は、テンポラリ関数ではサポートされていません。

- **LANGUAGE JAVA** – Java メソッドのラッパーです。Java プロシージャの呼び出しについては、「CREATE PROCEDURE 文」を参照してください。
- **ON EXCEPTION RESUME** – Transact-SQL のようなエラー処理を使用します。「CREATE PROCEDURE 文」を参照してください。
- **[NOT] DETERMINISTIC** – クエリで呼び出されるたびに、関数が再評価されます。このように指定されていない関数の結果は、パフォーマンスを向上させるためにキャッシュされ、クエリの評価中に同じパラメータで関数が呼び出されるたびに再利用されます。

基本となるデータを修正するなどの関連動作を伴う関数は、NOT DETERMINISTIC として宣言する必要があります。たとえば、プライマリキー値を生成し、INSERT ... SELECT 文で使用される関数は、次のように NOT DETERMINISTIC として宣言してください。

```
CREATE FUNCTION keygen( increment INTEGER )
RETURNS INTEGER
NOT DETERMINISTIC
BEGIN
    DECLARE keyval INTEGER;
    UPDATE counter SET x = x + increment;
    SELECT counter.x INTO keyval FROM counter;
    RETURN keyval
END
INSERT INTO new_table
SELECT keygen(1), ...
FROM old_table
```

特定の入力パラメータに対して常に同じ値を返す関数は、DETERMINISTIC として宣言できます。NOT DETERMINISTIC として宣言しないかぎり、ユーザ定

義関数はすべて DETERMINISTIC 関数として処理されます。DETERMINISTIC 関数は同じパラメータに対して一貫した結果を返し、副次的影響はありません。つまり、データベースサーバは、同じ関数を同じパラメータで 2 回連続して呼び出した場合は同じ結果が返され、クエリのセマンティックに不要な副次的影響が生じることはない想定します。

- **URL** – HTTP または SOAP Web サービスクライアント関数を定義する場合のみ使用します。Web サービスの URL を指定します。オプションのユーザ名とパスワードのパラメータは、HTTP 基本認証に必要なクレデンシャルとして機能します。HTTP 基本認証は、ユーザとパスワードの情報を base-64 でエンコードし、HTTP 要求の "Authentication" ヘッダに渡します。

Web サービスクライアント関数の場合、SOAP 関数と HTTP 関数の戻り型は、VARCHAR などの文字データ型のいずれかでなければなりません。返される値は、HTTP 応答の本文です。HTTP ヘッダ情報は含まれません。ステータス情報などの詳細情報が必要な場合は、関数の代わりにプロシージャを使用します。

パラメータ値は要求の一部として渡されます。使用される構文は、要求のタイプによって決まります。HTTP:GET の場合、パラメータは URL の一部として渡されます。HTTP:POST 要求の場合、値は要求の本文に置かれます。SOAP 要求へのパラメータは、常に要求本文にバンドルされます。

- **HEADER** – HTTP Web サービスクライアント関数を作成する場合は、この句を使用して HTTP 要求ヘッダのエントリを追加または変更します。HTTP ヘッダに指定できるのは印字可能な ASCII 文字のみで、大文字と小文字は区別されません。この句の使用法の詳細については、「CREATE PROCEDURE 文」の HEADER 句を参照してください。
- **SOAPHEADER** – SOAP Web サービスを関数として宣言する場合は、この句を使用して 1 つ以上の SOAP 要求ヘッダのエントリを指定します。SOAP ヘッダは、静的定数として宣言したり、代入パラメータメカニズムを使用して動的に設定したりできます (hd1、hd2 などに IN、OUT、または INOUT パラメータを宣言)。Web サービス関数では、1 つ以上の IN モード代入パラメータを定義できますが、INOUT または OUT 代入パラメータは定義できません。
- **TYPE** – Web サービス要求を行う場合に使用するフォーマットを指定します。SOAP が指定されている場合、または type 句が含まれていない場合は、デフォルトのタイプである SOAP:RPC が使用されます。HTTP は HTTP:POST を暗黙的に指定します。SOAP 要求は常に XML 文書として送信されるため、SOAP 要求の送信には常に HTTP:POST が使用されます。
- **NAMESPACE** – SOAP クライアント関数のみに適用されます。この句は、SOAP:RPC 要求と SOAP:DOC 要求の両方に通常必要なメソッドネームスペースを示します。要求を処理する SOAP サーバは、このネームスペースを使用し

て、SOAP 要求メッセージ本文内のエンティティの名前を解釈します。ネームスペースは、Web サービスサーバから使用できる SOAP サービスの WSDL 記述から取得できます。デフォルト値は、プロシージャの URL のオプションのパスコンポーネントの直前までです。

- **CERTIFICATE** – 安全な (HTTPS) 要求を行うには、HTTPS サーバで使用される証明書にクライアントがアクセスできる必要があります。必要な情報は、セミコロンで区切られたキー／値のペアの文字列で指定されます。証明書はファイルに置かれ、file キーを使用して提供されるファイルの名前、または証明書全体を文字列に配置できますが、両方は配置できません。次のキーを使用できます。

キー	省略形	説明
file		証明書のファイル名
certificate	cert	証明書自体
company	co	証明書で指定された会社
unit		証明書で指定された会社の部門
name		証明書で指定された共通名

証明書は、HTTPS サーバに対する直接の要求、または安全でないサーバから安全なサーバにリダイレクトされる可能性がある要求に対してのみ必要です。

- **CLIENTPORT** – HTTP クライアントプロシージャが TCP/IP を使用して通信するポート番号を示します。この句は、ファイアウォールを介する通信のためのものであり、このような通信にのみおすすすめします。ファイアウォールは TCP/UDP ポートに従ってフィルタします。単一のポート番号、ポート番号の範囲、または両方の組み合わせを指定できます。たとえば、CLIENTPORT '85,90-97' を指定できます。
- **PROXY** – プロキシサーバの URI を指定します。クライアントがプロキシを介してネットワークにアクセスする場合に使用します。プロシージャがプロキシサーバに接続し、そのプロキシサーバを介して Web サービスに要求を送信することを示します。

例

(先頭に戻る) (169 ページ)

- **例 1** – 文字列 `firstname` と文字列 `lastname` を連結します。

```
CREATE FUNCTION fullname (
  firstname CHAR(30),
  lastname CHAR(30) )
RETURNS CHAR(61)
```

SQL 文

```
BEGIN
  DECLARE name CHAR(61);
  SET name = firstname || ' ' || lastname;
  RETURN (name);
END
```

fullname 関数は、次の例のように使用します。

- 2つの提供された文字列からフルネームを戻します。

```
SELECT fullname ('joe','smith')
```

fullname('joe', 'smith')
joe smith

- 全従業員の名前をリストします。

```
SELECT fullname (givenname, surname)
FROM Employees
```

fullname (givenname, surname)
Fran Whitney
Matthew Cobb
Philip Chin
Julie Jordan
Robert Breault
...

- **例 2** – Transact-SQL 構文を使用しています。

```
CREATE FUNCTION DoubleIt ( @Input INT )
RETURNS INT
AS
DECLARE @Result INT
SELECT @Result = @Input * 2
RETURN @Result
```

文 `SELECT DoubleIt(5)` は、10 の値を返します。

- **例 3** – Java で記述された外部関数を作成します。

```
CREATE FUNCTION dba.encrypt( IN name char(254) )
RETURNS VARCHAR
EXTERNAL NAME
'Scramble.encrypt (Ljava/lang/String;)Ljava/lang/String;'
LANGUAGE JAVA
```

使用法

(先頭に戻る) (169 ページ)

ユーザ定義関数を修正したり、定義にスクランブルをかけて関数の内容を隠したりするには、**ALTER FUNCTION** 文を使用します。

関数を実行するときに、必ずしもすべてのパラメータを指定する必要はありません。**CREATE FUNCTION** 文の中にデフォルト値がある場合、パラメータが指定されていないと、デフォルト値が割り当てられます。呼び出し元によって引数が指定されておらず、デフォルトも設定されていない場合は、エラーが発生します。

関連する動作：

- オートコミット

標準

(先頭に戻る) (169 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - Adaptive Server ではサポートされていません。

パーミッション

(先頭に戻る) (169 ページ)

自分が所有する関数の場合、CREATE PROCEDURE システム権限が必要です。

任意のユーザが所有する関数の場合、次のいずれかが必要です。

- CREATE ANY PROCEDURE システム権限
- CREATE ANY OBJECT システム権限

外部参照を含む関数を作成するには、関数の所有者であるかどうかに関係なく、CREATE EXTERNAL REFERENCE システム権限も必要です。

参照：

- ALTER FUNCTION 文 (22 ページ)
- BEGIN ... END 文 (108 ページ)
- CREATE PROCEDURE 文 (210 ページ)
- DROP 文 (319 ページ)
- RETURN 文 (497 ページ)

CREATE FUNCTION 文 (Java UDF)

新しい外部 Java テーブル UDF 関数をデータベース内に作成します。

クイックリンク：

「パラメータ」 (176 ページ)

「例」 (178 ページ)

「使用法」 (178 ページ)

「標準」 (178 ページ)

「パーミッション」 (179 ページ)

構文

```
CREATE [ OR REPLACE | TEMPORARY ] FUNCTION [ owner.]function-name
  ( [ parameter (176 ページ), ...] )
  [ SQL SECURITY { INVOKER | DEFINER } ]
  RETURNS data-type
  ON EXCEPTION RESUME
  | [ NOT ] DETERMINISTIC
  { compound-statement | AS tsq-compound-statement (176 ページ)
  | EXTERNAL NAME 'java-call (176 ページ)' LANGUAGE JAVA [ ALLOW | DISALLOW
SERVER SIDE REQUESTS ] environment-name}
```

parameter - (構文に戻る) (176 ページ)

```
IN parameter-name data-type [ DEFAULT expression ]
```

tsq-compound-statement - (構文に戻る) (176 ページ)

```
sql-statement
sql-statement ...
```

java-call - (構文に戻る) (176 ページ)

```
' [ package-name.]class-name.method-name method-signature (176 ページ)'
```

method-signature - (back to java-call) (176 ページ)

```
( [ field-descriptor (176 ページ), ...] ) return-descriptor (176 ページ)
```

field-descriptor and **return-descriptor** - (back to method-signature) (176 ページ)

```
Z | B | S | I | J | F | D | C | V | [ descriptor | L class-name;
```

パラメータ

(先頭に戻る) (175 ページ)

- **CREATE [OR REPLACE]** - パラメータ名は、データベース識別子に関するルールに従って付ける必要があります。また、有効な SQL データ型でなければなりません。さらに、この引数が関数に値を渡す式であることを示すキーワード **IN** を先頭に付ける必要があります。

CREATE 句を指定すると新しい関数が作成され、**OR REPLACE** 句を使用すると既存の関数が同じ名前でも置き換えられます。関数が置き換えられた場合、関数の定義は変更されますが、既存のパーミッションは保持されます。**OR REPLACE** 句をテンポラリ関数で使用することはできません。

- **TEMPORARY** – 作成した接続でのみ参照できる関数になり、接続を削除すると関数も自動的に削除されます。テンポラリ関数を明示的に削除することもできます。テンポラリ関数に対して **ALTER**、**GRANT**、および **REVOKE** 操作を実行することはできません。また他の関数とは異なり、テンポラリ関数はカタログやトランザクションログに記録されません。

テンポラリ関数は、作成者 (現在のユーザ) のパーミッションで実行されます。また作成者のみが所有できます。そのため、テンポラリ関数を作成するときは所有者を指定しないでください。読み込み専用のデータベースに接続するときに、テンポラリ関数の作成と削除を行うことができます。

- **SQL SECURITY – INVOKER** (関数を呼び出しているユーザ) または **DEFINER** (関数を所有しているユーザ) のどちらとして関数が実行されるかを定義します。デフォルトは **DEFINER** です。

INVOKER を指定すると、プロシージャを呼び出す各ユーザに対して注釈を付ける必要があるため、より多くのメモリが使用されます。また、名前解決も **INVOKER** として実行されます。そのため、適切な所有者で、すべてのオブジェクト名 (テーブル、プロシージャなど) を修飾するように注意してください。

- **data-type** – **LONG BINARY** と **LONG VARCHAR** を戻り値のデータ型として使用することはできません。
- **compound-statement** – **BEGIN** と **END** で囲まれ、セミコロンで区切られた SQL 文のセットです。「**BEGIN ... END 文**」を参照してください。
- **tsql-compound-statement** – Transact-SQL 文のバッチです。
- **[NOT] DETERMINISTIC** – クエリで呼び出されるたびに、関数が再評価されます。このように指定されていない関数の結果は、パフォーマンスを向上させるためにキャッシュされ、クエリの評価中に同じパラメータで関数が呼び出されるたびに再利用されます。

基本となるデータを修正するなどの関連動作を伴う関数は、**NOT DETERMINISTIC** として宣言する必要があります。たとえば、プライマリキー値を生成し、**INSERT ... SELECT** 文で使用される関数は、次のように **NOT DETERMINISTIC** として宣言してください。

```
CREATE FUNCTION keygen( increment INTEGER )
RETURNS INTEGER
NOT DETERMINISTIC
BEGIN
    DECLARE keyval INTEGER;
    UPDATE counter SET x = x + increment;
    SELECT counter.x INTO keyval FROM counter;
    RETURN keyval
END
INSERT INTO new_table
```

```
SELECT keygen(1), ...
FROM old_table
```

特定の入力パラメータに対して常に同じ値を返す関数は、**DETERMINISTIC** として宣言できます。**NOT DETERMINISTIC** として宣言しないかぎり、ユーザ定義関数はすべて **DETERMINISTIC** 関数として処理されます。**DETERMINISTIC** 関数は同じパラメータに対して一貫した結果を返し、副次的影響はありません。つまり、データベースサーバは、同じ関数を同じパラメータで 2 回連続して呼び出した場合は同じ結果が返され、クエリのセマンティックに不要な副次的影響が生じることはない想定します。

- **LANGUAGE JAVA** – Java メソッドのラッパーです。Java プロシージャの呼び出しについては、「CREATE PROCEDURE 文」を参照してください。
- **environment-name** – Java メソッドのラッパーです。

DISALLOW 句がデフォルトです。**ALLOW** 句はサーバ側接続を許可することを示します。

注意： **ALLOW** 句は、必要な場合以外は指定しないでください。**ALLOW** は特定の種類の SAP Sybase IQ テーブルジョインの速度を低下させます。UDF を使用するとき **ALLOW** 句と **DISALLOW SERVER SIDE REQUESTS** 句の両方を同じクエリの中で指定しないでください。

例

(先頭に戻る) (175 ページ)

- **例 1** – 次の文は、Java で記述された外部関数を作成します。

```
CREATE FUNCTION dba.encrypt( IN name char(254) )
RETURNS VARCHAR
EXTERNAL NAME
'Scramble.encrypt (Ljava/lang/String;)Ljava/lang/String;'
LANGUAGE JAVA
```

使用法

(先頭に戻る) (175 ページ)

関数を実行するときに、必ずしもすべてのパラメータを指定する必要はありません。**CREATE FUNCTION** 文の中にデフォルト値がある場合、パラメータが指定されていないと、デフォルト値が割り当てられます。呼び出し元によって引数が指定されておらず、デフォルトも設定されていない場合は、エラーが発生します。

標準

(先頭に戻る) (175 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - Adaptive Server ではサポートされていません。

パーミッション

(先頭に戻る) (175 ページ)

自分が所有する関数の場合、CREATE PROCEDURE システム権限が必要です。

任意のユーザが所有する関数の場合、次のいずれかが必要です。

- CREATE ANY PROCEDURE システム権限
- CREATE ANY OBJECT システム権限。

外部参照を含む関数を作成するには、関数の所有者であるかどうかに関係なく、CREATE EXTERNAL REFERENCE システム権限も必要です。

CREATE INDEX 文

指定したテーブル、またはテーブルのペアにインデックスを作成します。一度インデックスを作成すると、**DROP INDEX** 文を使用して削除するとき以外は、SQL 文中で再び参照されることはありません。

クイックリンク：

「パラメータ」 (180 ページ)

「例」 (183 ページ)

「使用法」 (183 ページ)

「標準」 (187 ページ)

「パーミッション」 (188 ページ)

構文

```
CREATE [ UNIQUE ] [ index-type ] INDEX [ IF NOT EXISTS ] index-name
...ON [ owner.]table-name
... ( column-name [ , column-name ] ...)
...[ { IN | ON } dbspace-name ]
...[ NOTIFY integer ]
...[ DELIMITED BY 'separators-string' ]
...[ LIMIT maxwordsize-integer ]
```

index-type

```
{ CMP | HG | HNG | LF | WD | DATE | TIME | DTTM }
```

パラメータ

(先頭に戻る) (179 ページ)

- **index-type** – SAP Sybase IQ テーブルのカラムの場合、HG (High_Group)、HNG (High_Non_Group)、LF (Low_Fast)、WD (Word)、DATE、TIME、または DTTM (Datetime) を index-type に指定できます。index-type を指定しない場合は、HG インデックスがデフォルトで作成されます。

IQ メインストアテーブルの 2 つのカラム間の関係にインデックスを作成するには、index-type に CMP (Compare) を指定します。2 つのカラムのデータ型、精度、位取りは同じであることが必要です。CHAR、VARCHAR、BINARY、または VARBINARY のカラムの場合、精度は 2 つのカラムの幅が同じであることを意味します。

クエリ速度をできるだけ高速にするために、どのインデックスタイプを選択すべきかは、次の条件に応じて異なります。

- カラム内のユニークな値の数
- クエリ内での当該カラムの使用方法
- 使用可能なディスク領域量

IQ メインストアテーブルの 1 つのカラムに複数のインデックスを指定できますが、各インデックスは異なるインデックスタイプでなければなりません。

CREATE INDEX を使用して、重複するインデックスタイプを追加することはできません。現在のクエリまたはその一部において利用可能なインデックスの中で最速のものが SAP Sybase IQ によって選択されます。ただし、追加する各インデックスタイプによって、そのテーブルの必要領域量が大幅に増加することがあります。

- **column-name** – インデックスを作成するカラムの名前を指定します。カラム名は、オプションの相関名の後に続く識別子です (通常、相関名はテーブル名です。相関名の詳細については、「FROM 句」を参照してください)。カラム名に英字、数字、アンダースコア以外の文字が使用されている場合は、二重引用符 (") で囲んでください。

UNIQUE 句を指定すると、指定可能なインデックスは HG と LF だけになります。外部キーにはユニークでないインデックスが必要で、複合外部キーにはユニークでない複合 HG インデックスが必要です。マルチカラムの複合キーの場合、ユニークでもユニークでなくても、HG インデックスの幅の最大値は 5300 バイトです。CHAR または VARCHAR のデータは、複合キーまたはシングルカラムの HG、LF、HNG、DATE、TIME、または DTTM のインデックスの一部である場合は、255 バイト以内にする必要があります。

- **UNIQUE** – インデックスの全カラムで、テーブル内に同じ値を持つローが 2 つ存在しないことを保証します。各インデックスキーはユニークであるか、少な

くとも 1 つのカラムで NULL を格納している必要があります。複数のカラムを持つユニークな HG インデックスは作成できますが、それ以外のインデックスタイプを使用してマルチカラムインデックスを作成することはできません。UNIQUE は、CMP、HNG、WD、DATE、TIME、または DTTM のインデックスタイプを使用して指定することはできません。

SAP Sybase IQ では、カラム定義で NULL 値の使用が許可され、制約 (プライマリーキーまたは一意性) も適用されていない場合に、ユーザが作成したユニークなマルチカラム HG インデックスのデータ値で NULL を使用できます。詳細については、「注意事項」の「マルチカラムインデックス」を参照してください。

- **IF NOT EXISTS** – 指定したオブジェクトがすでに存在する場合、変更は行われず、エラーは返されません。
- **IN** – インデックスの配置を指定します。IN 句を省略した場合は、テーブルが作成される DB 領域にインデックスが作成されます。インデックスは常に、そのインデックスのテーブルと同じ型の DB 領域 (IQ ストアまたはテンポラリストア) に配置されます。インデックスをロードすると、そのタイプのデータベースファイルで、利用可能な領域のあるものすべてにデータが分配されます。SAP Sybase IQ は指定された *dbspace-name* がそのインデックスに適合することが保証されます。テンポラリテーブル上のインデックスに対して `IQ_SYSTEM_MAIN` または他のメイン DB 領域を指定した場合や、その逆の場合には、エラーが発生します。**CREATE DATABASE... CASE IGNORE** または **CASE RESPECT** の指定に関係なく、DB 領域名では常に大文字と小文字が区別されません。
- **DELIMITED BY** – カラムの文字列を複数のワードに解析して、そのカラムの WD インデックスに格納するためのセパレータを指定します。この句が省略されるか、値に空の文字列が指定された場合、SAP Sybase IQ はデフォルトのセパレータのセットを使用します。デフォルトのセパレータのセットは、デフォルトの照合順序 (ISO-BINENG) にあわせて作成されています。7 ビットの ASCII 文字 (7 ビットの ASCII 英数字ではありません) で構成され、ハイフンと一重引用符は含まれません。ハイフンと一重引用符は、デフォルトではワードの一部となります。デフォルトのセパレータのセットには 64 文字が含まれます。たとえば、カラムの値が次の文字列だとします。

```
The cat is on the mat
```

また、データベースが **CASE IGNORE** の設定でデフォルトのセパレータを使用して作成されたとすると、この文字列から WD インデックスに次のワードが格納されます。

```
cat is mat on the
```

DELIMITED BY 句と LIMIT 句を複数指定した場合は、エラーにはなりません
が、各タイプの最後に指定した句だけが使用されます。

- **separators-string** – データベースの作成時に使用した照合順序に含まれる文字を 0 個以上並べて指定する必要があります。separators-string の文字は、それぞれセパレータとして扱われます。separators-string に文字が指定されなければ、デフォルトのセパレータのセットが使用されます (セパレータには使用する照合順序に含まれる単一の文字を指定してください)。separators-string に 256 を超える文字 (セパレータ) を指定することはできません。

タブをデリミタに指定するには、separators-string に <TAB> の文字を入力するか、タブ文字を表す 16 進の ASCII コード (¥x09) を使用します。"¥t" と入力すると、¥ と t が指定されます。改行をデリミタに指定するには、<RETURN> の文字か、16 進の ASCII コード (¥x0a) を入力します。

たとえば、DELIMITED BY ' : ; . ¥ / t ' 句では、次の 7 つのセパレータが指定されます。空白 : ; . ¥ / t

表 4: タブと改行をデリミタに指定

デリミタ	DELIMITED BY 句のセパレータ文字列
タブ	' ' (<TAB> を入力) または '¥x09'
改行文字	' ' (<RETURN> を入力) または '¥x0a'

- **LIMIT** – WD インデックスの作成時にのみ使用できます。WD インデックスで許可される、ワードの最大長を指定します。解析中にこれよりも長いワードが検出されるとエラーが発生します。デフォルトは 255 バイトです。指定可能な値の最小値は 1 で、最大値は 255 です。**CREATE INDEX** 文に指定された、またはデフォルトで定義された最大ワード長がカラム幅を超える場合は、使用された最大ワード長がカラム幅まで自動的に削減されます。最大ワード長を短く設定すると、挿入、削除、更新に必要な領域と時間が節約できます。空のワード (セパレータを 2 つ続けて入力) は自動的に無視されます。WD インデックスを作成すると、そのカラムに挿入されるデータはすべて、作成時に定義したセパレータと最大ワードサイズで解析されるようになります。セパレータや最大ワードサイズを、インデックスが作成された後で変更することはできません。
- **NOTIFY** – n 個のレコードが正常にインデックスに追加された後で、通知メッセージを表示します。メッセージは標準出力デバイスに送信されます。メッセージには、メモリの使用状況、データベース領域、使用中のバッファ数に関する情報が含まれます。デフォルトは 100,000 レコードです。NOTIFY をオフにするには、NOTIFY を 0 に設定します。

例

(先頭に戻る) (179 ページ)

- **例 1** – projected_earnings カラムと current_earnings カラムに比較インデックスを作成します。この 2 つのカラムは、精度と位取りが同じである 10 進のカラムです。

```
CREATE CMP INDEX proj_curr_cmp
ON sales_data
( projected_earnings, current_earnings )
```

- **例 2** – SalesOrderItems テーブルの ID カラムに High_Group インデックスを作成します。このインデックスのデータページは、DB 領域 Dsp5 から割り付けられます。

```
CREATE HG INDEX id_hg
ON SalesOrderItems
( ID ) IN Dsp5
```

- **例 3** – SalesOrderItems テーブルの ProductID カラムに High_Group インデックスを作成します。

```
CREATE HG INDEX item_prod_hg
ON Sales_OrderItems
( ProductID)
```

- **例 4** – 通知メッセージを生成しないで、SalesOrderItems テーブルの同じ ProductID カラムに Low_Fast インデックスを作成します。

```
CREATE LF INDEX item_prod
ON SalesOrderItems
( ProductID)
NOTIFY 0
```

- **例 5** – earnings_report テーブルで **WD** インデックスを作成します。文字列のデリミタに、スペース、コロン、セミコロン、ピリオドを指定しています。文字列の長さは 25 に制限されます。

```
CREATE WD INDEX earnings_wd
ON earnings_report_table(varchar)
DELIMITED BY ' :;. '
LIMIT 25
```

- **例 6** – SalesOrders テーブルの OrderDate カラムに **DTTM** インデックスを作成します。

```
CREATE DTTM INDEX order_dttm
ON SalesOrders
( OrderDate )
```

使用法

(先頭に戻る) (179 ページ)

- **CREATE INDEX** 文ではインデックスの所有者を指定できません。インデックスが定義されているテーブルの所有者が、自動的にそのインデックスの所有者になります。各所有者に対して、インデックス名はユニークである必要があります。
- ビューに対してインデックスを作成することはできません。各インデックス名は特定のテーブルに対してユニークである必要があります。
- **CREATE INDEX** 文は、他の接続で現在変更中のテーブルに影響を及ぼす場合は処理されません。ただし、インデックスを追加しているテーブルに対して、クエリを実行することは可能です。
- **WD** インデックスを作成すると、そのカラムへ挿入されるデータはすべて、セパレータで解析されます。最大ワードサイズはインデックスの作成後に変更できません。CHAR カラムを使用する場合は、少なくともスペースをセパレータの1つに指定するか、デフォルトのセパレータセットを使用します。SAP Sybase IQ では、最大カラム幅になるまで CHAR カラムに自動的にスペースが埋め込まれます。カラムに文字データのほかにブランクが含まれる場合は、**WD** インデックスが作成されたデータに対するクエリで、誤った結果が返されることがあります。たとえば、CompanyName というカラムにセパレータで区切られた2つの単語が含まれているとします。ただし、2つ目の単語には次のようにブランクが埋め込まれています。

```
'Concord' 'Farms'
```

このとき、ユーザが次のクエリを入力したとします。

```
SELECT COUNT(*) FROM Customers WHERE CompanyName contains ('Farms')
```

パーサはこの文字列を次のように解釈します。

```
'Farms'
```

次の文字列に一致するとは解釈しないため、

```
'Farms'
```

1 ではなく 0 を返します。この問題を防ぐには、CHAR カラムではなく VARCHAR カラムを使用します。

- データ型:
 - **CREATE INDEX** を使用して、BIT データを格納するカラムにインデックスを作成することはできません。
 - 255 バイトを超える CHAR および VARCHAR データに対して作成できるのは、デフォルトインデックス、**CMP** インデックス、および **WD** インデックスのみです。
 - LONG VARCHAR データに対して作成できるのは、デフォルトのインデックスタイプと **WD** インデックスタイプのみです。

- 255 バイトよりも大きい **BINARY** と **VARBINARY** のデータに作成可能なインデックスタイプは、デフォルトインデックス、**CMP** インデックス、および **TEXT** インデックスのみです。
- **FLOAT**、**REAL**、または **DOUBLE** のデータを格納するカラムに **HNG** インデックスおよび **CMP** インデックスを作成することはできません。
- **TIME** インデックスを作成できるのは、**TIME** データ型のカラムだけです。
- **DATE** インデックスを作成できるのは、**DATE** データ型のカラムだけです。
- **DTTM** インデックスを作成できるのは、**DATETIME** データ型または **TIMESTAMP** データ型のカラムだけです。
- 複数のカラムから、ユニークなまたはユニークでない **HG** インデックスを作成できます。SAP Sybase IQ では、外部キーを構成するカラムセットに対して、ユニークでない **HG** インデックスが自動的に作成されます。
複数のカラムに対応できるインデックスタイプは、**HG** と **CMP** だけです。複数のカラムから、ユニークな **HNG** または **LF** のインデックス、**DATE**、**TIME**、または **DTTM** のインデックスは作成できません。

マルチカラムを連結したキーの最大幅は 5KB (5300 バイト) です。連結可能なカラムの数は、5KB に収まるカラムの数によって決まります。CHAR または VARCHAR のデータは、それがシングルカラムの **HG**、**LF**、**HNG**、**DATE**、**TIME**、または **DTTM** のインデックスの複合キーの一部である場合は、255 バイトを超えることはできません。

マルチカラムインデックスに対する **INSERT** には、インデックスのすべてのカラムを含める必要があります。

ORDER BY 句にカラムが 1 つだけある場合は、マルチカラム **HG** インデックスを使用するとクエリが速く実行されます。次に例を示します。

```
SELECT abs (x) from t1
ORDER BY x
```

上記の例では、**HG** インデックスはソート順に *x* を縦方向に射影します。

クエリパフォーマンスを高めるには、マルチカラム **HG** インデックスを使用して、次の状況にある **SELECT** または **ORDER BY** 句にある複数のカラム (**ROWID** を含めることも可能) に対して、**ORDER BY** 操作を実行します。

- すべての射影されたカラムとすべての順序カラム (**ROWID** を除く) がインデックス内に存在する。
- 順序キーが先頭の **HG** カラムに順に一致する。

複数のマルチカラム **HG** インデックスが上記の条件を満たしている場合は、個別カウント数が最も低いインデックスが使用されます。

クエリに **ORDER BY** 句が含まれており、**ORDER BY** カラムリストが (**SELECT** リストに参照されるすべてのカラムがマルチカラムインデックス内にある) マルチカラムインデックスのプレフィクスである場合、マルチカラムインデックスは次のように縦方向の射影を使用します。

```
SELECT x, z, y FROM T
ORDER BY x, y
```

SELECT リストのベースカラム上に式が存在し、すべての式で参照された全カラムがマルチカラムインデックスにある場合、クエリは次のようにマルチカラムインデックスを使用します。

```
SELECT power(x,2), x+y, sin(z) FROM T
ORDER BY x, y
```

上記の2つの例以外にも、**ROWID()** 関数が **SELECT** リストの式にある場合は、マルチカラムインデックスが使用されます。次に例を示します。

```
SELECT rowid()+x, z FROM T
ORDER BY x, y, z
```

上記の3つの例以外にも、**ROWID()** が **ORDER BY** リストの終わりにあり、そのリストのカラム (**ROWID()** のリストを除く) が正確な順序でマルチカラムインデックスを使用している場合は、マルチカラムインデックスがクエリに対して使用されます。次に例を示します。

```
SELECT z, y FROM T
ORDER BY x, y, z, ROWID()
```

SAP Sybase IQ では、カラム定義で NULL 値の使用が許可され、制約 (プライマリキーまたは一意性) も適用されていない場合に、ユーザが作成したユニークなマルチカラム **HG** インデックスのデータ値の中で NULL を使用できます。この機能のルールを次に示します。

- NULL は未定義の値として扱われる。
- ユニークなインデックスカラムで、複数のローに NULL 値を設定できる。
 1. シングルカラムインデックスで、インデックスカラムの複数のローに NULL 値を設定できる。
 2. マルチカラムインデックスで、1つまたは複数のインデックスカラムの複数のローに NULL 値を設定できる。ただし、インデックスのユニーク性が確保できるように他のカラムが非 NULL 値の場合に限る。
 3. マルチカラムインデックスで、インデックスに関与しているすべてのカラムで複数のローに NULL 値を設定できる。

次にこれらの規則の例を示します。次のようにして作成されたテーブル `table1` があるとします。

```
CREATE TABLE table1
(c1 INT NULL, c2 INT NULL, c3 INT NOT NULL);
```

NULL 値が使用できるカラムに、ユニークなシングルカラムの **HG** インデックスを作成します。

```
CREATE UNIQUE HG INDEX c1_hg1 ON table1 (c1);
```

上記の規則 1 に従って、複数のローのインデックスカラムに NULL 値を挿入します。

```
INSERT INTO table1(c1,c2,c3) VALUES (NULL,1,1);
INSERT INTO table1(c1,c2,c3) VALUES (NULL,2,2);
```

NULL 値が使用できるカラムに、ユニークなマルチカラムの **HG** インデックスを作成します。

```
CREATE UNIQUE HG INDEX c1c2_hg2 ON table1(c1,c2);
```

上記の規則 2 に従って、インデックスのユニークさを確保する必要があります。次の **INSERT** は成功しません。これは、ロー 1 とロー 3 のマルチカラムインデックス c1c2_hg2 に同じ値が入るためです。

```
INSERT INTO table1(c1,c2,c3) VALUES (NULL,1,3);
```

しかし、次の **INSERT** 操作は、規則 1 と規則 3 に従って成功します。

```
INSERT INTO table1(c1,c2,c3) VALUES (NULL,NULL,3);
INSERT INTO table1(c1,c2,c3) VALUES (NULL,NULL,4);
```

マルチカラムインデックスでのユニークさが保持されています。

次の **UPDATE** 操作は成功します。これは、規則 3 によって、複数のローでマルチカラムインデックスのすべてのカラムに NULL 値を設定できるためです。

```
UPDATE table1 SET c2=NULL WHERE c3=1
```

マルチカラム **HG** インデックスが一意性制約に制御されている場合は、インデックスに参与しているカラムには NULL 値を設定できません。

- **BEGIN PARALLEL IQ ... END PARALLEL IQ** 文を使用して、複数の IQ メインストアテーブルに対する **CREATE INDEX** 文をグループにまとめることで、それらの文を 1 つの DDL 文であるかのように実行できます。詳細については、「**BEGIN PARALLEL IQ ... END PARALLEL IQ** 文」を参照してください。

警告! コミットされていないデータを含むローカルテンポラリテーブルで **CREATE INDEX** コマンドを使用するとエラーになり、「インデックスを作成するためにローカルテンポラリテーブル <tablename> をコミットする必要があります。」というエラーメッセージが表示されます。ローカルテンポラリテーブルのデータをコミットしてからインデックスを作成してください。

関連する動作：

- オートコミット

標準

(先頭に戻る) (179 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Adaptive Server には、SAP Sybase IQ よりも複雑な **CREATE INDEX** 文があります。Adaptive Server 構文は SAP Sybase IQ で使用できませんが、一部の句やキーワードは無視されます。Adaptive Server **CREATE INDEX**

文の完全な構文については、『Adaptive Server リファレンスマニュアル 第2巻：コマンド』を参照してください。

Adaptive Server のインデックスは、クラスタドインデックスまたはノンクラスタドインデックスのどちらかになります。クラスタドインデックスを使用すると、ほとんどの場合ノンクラスタドインデックスよりも速くデータを検索できます。テーブルごとに1つだけクラスタドインデックスが許可されます。

SAP Sybase IQ は、クラスタドインデックスをサポートしていません。CLUSTERED と NONCLUSTERED のキーワードは SQL Anywhere で許可されますが、SAP Sybase IQ では無視されます。*index-type*が指定されない場合、SAP Sybase IQ は指定のカラムに対して HG インデックスを作成します。

SAP Sybase IQ では、DESC キーワードを指定できません。

インデックス名は、SAP Sybase IQ でも Adaptive Server でも、指定されたテーブル内でユニークである必要があります。

パーミッション

(先頭に戻る) (179 ページ)

インデックスが作成される DB 領域に対する CREATE 権限が必要です。次のいずれかも必要です。

- CREATE ANY INDEX システム権限
- CREATE ANY OBJECT システム権限
- インデックス付きのテーブルに対する REFERENCES 権限
- 基本となるテーブルを所有していること

参照：

- BEGIN PARALLEL IQ ... END PARALLEL IQ 文 (111 ページ)
- DROP 文 (319 ページ)
- INDEX_PREFERENCE オプション (691 ページ)
- FROM 句 (372 ページ)

CREATE LDAP SERVER 文

LDAP ユーザ認証用の新しい LDAP サーバ設定オブジェクトを作成します。LDAP サーバ設定オブジェクトの作成中に定義されたパラメータは、ISYSLDAPSERVER (システムビュー SYSLDAPSERVER) システムテーブルに格納されます。

クイックリンク：

「パラメータ」 (189 ページ)

「例」 (190 ページ)

「標準」 (192 ページ)

「パーミッション」 (192 ページ)

構文

```
CREATE LDAP SERVER ldapua-server-name
  [ ldapua-server-attribs ]
  [ WITH ACTIVATE ]
```

ldapua-server-attribs

SEARCH DN

```
URL { 'URL_string' | NULL }
| ACCESS ACCOUNT { 'DN_string' | NULL }
| IDENTIFIED BY ( 'password' | NULL )
| IDENTIFIED BY ENCRYPTED { encrypted-password | NULL }
| AUTHENTICATION URL { 'URL_string' | NULL }
| CONNECTION TIMEOUT timeout_value
| CONNECTION RETRIES retry_value
| TLS { ON | OFF }
```

パラメータ

(先頭に戻る) (188 ページ)

- **URL** – 指定されたユーザ ID のホスト (名前または IP アドレスで指定)、ポート番号、および DN ルックアップで実行される検索を指定します。この値は、ISYSLDAPSERVER システムテーブルに格納される前に、LDAP URL 構文が正しいかどうかを検証されます。この文字列の最大サイズは 1024 バイトです。
- **ACCESS ACCOUNT** – SAP Sybase IQ 内のユーザではなく、SAP Sybase IQ が使用するために LDAP サーバで作成されたユーザ。このユーザの識別名 (DN) は、LDAP サーバへの接続に使用されます。このユーザは、SEARCH DN URL で指定された場所でユーザ ID によって DN を検索するためのパーミッションを、LDAP サーバ内に保持しています。この文字列の最大サイズは 1024 バイトです。
- **IDENTIFIED BY** – ACCESS ACCOUNT ユーザに関連付けられたパスワードを指定します。このパスワードは、対称暗号化を使用してディスクに保存されます。パスワードを解除して何も設定しない場合は、値 NULL を指定します。クリアテキストのパスワードの最大サイズは 255 バイトです。
- **IDENTIFIED BY ENCRYPTED** – ACCESS ACCOUNT に指定されている識別名に関連付けられた暗号化形式のパスワードを設定します。バイナリ値は暗号化されたパスワードであるため、ディスクにそのまま保存されます。パスワード

を解除して何も設定しない場合は、値 NULL を指定します。バイナリの最大サイズは 289 バイトです。暗号化キーは有効な varbinary 値である必要があります。暗号化キーは、引用符で囲まないでください。

- **AUTHENTICATION URL** – ユーザの認証に使用する LDAP サーバのホスト (名前または IP アドレスで指定) とポート番号を指定します。これは、URL_string として定義された値で、ISYSLDAPSERVER に格納される前に、LDAP URL 構文が正しいかどうかを検証されます。事前の DN 検索によって得られたユーザの DN とユーザパスワードによって、新しい接続が認証 URL にバインドされません。LDAP サーバへの正常な接続は、接続ユーザの ID の証明とみなされます。この文字列の最大サイズは 1024 バイトです。
- **CONNECTION TIMEOUT** – DN 検索と認証の両方に使用する SAP Sybase IQ から LDAP サーバへの接続のタイムアウトを指定します。この値はミリ秒で指定します。デフォルト値は 10 秒です。
- **CONNECTION RETRIES** – DN 検索と認証の両方に使用する SAP Sybase IQ から LDAP サーバへの接続の再試行回数を指定します。有効な値の範囲は 1 ~ 60 で、デフォルト値は 3 です。
- **TLS** – DN 検索と認証の両方に使用する LDAP サーバへの接続に、TLS とセキュア LDAP プロトコルのいずれを使用するかを定義します。ON に設定すると、TLS プロトコルが使用され、URL は "ldap://" で始まります。OFF に設定すると (または指定しないと)、セキュア LDAP プロトコルが使用され、URL は "ldaps://" で始まります。TLS プロトコルを使用する場合は、LDAP サーバによって使用される証明書に署名する認証局 (CA) の証明書が含まれているファイル名を使用して、データベースセキュリティオプション TRUSTED_CERTIFICATES_FILE を指定します。
- **WITH ACTIVATE** – LDAP サーバ設定オブジェクトを有効にして、作成時にすぐに使用できるようにします。これによって、1 つの文で LDAP ユーザ認証の定義と有効化を行うことができます。WITH ACTIVATE を使用すると、LDAP サーバ設定オブジェクトのステータスは READY に変わります。

例

(先頭に戻る) (188 ページ)

- **例 1** – 検索パラメータ、認証 URL を設定し、タイムアウトを 3 秒に設定し、ユーザの認証を開始できるようにサーバを有効化します。TLS プロトコルまたは SECURE LDAP プロトコルを使用しないで LDAP サーバに接続します。

```
SET OPTION PUBLIC.login_mode = 'Standard,LDAPUA'
CREATE LDAP SERVER apps_primary
SEARCH DN
```

```

URL 'ldap://my_LDAPserver:389/dc=MyCompany,dc=com??sub?cn=*'
ACCESS ACCOUNT 'cn=aseadmin, cn=Users, dc=mycompany, dc=com'
IDENTIFIED BY 'Secret99Password'
AUTHENTICATION URL 'ldap://my_LDAPserver:389/'
CONNECTION TIMEOUT 3000
WITH ACTIVATE

```

- 例 2** – 例 1 と同じ検索パラメータを使用しますが、“ldaps” を指定することにより、ホスト my_LDAPserver、ポート 636 の LDAP サーバとのセキュア LDAP 接続が確立されるようにします。このポートにはセキュア LDAP プロトコルを使用している LDAP クライアントだけが接続できます。データベースセキュリティオプション TRUSTED_CERTIFICATE_FILE は、“ldaps://my_LDAPserver:636” の LDAP サーバで使用される証明書に署名した認証局 (CA) の証明書が含まれるファイル名を使用して設定されている必要があります。LDAP サーバとのハンドシェイクの際に、LDAP サーバから提示された証明書が SAP Sybase IQ サーバ (LDAP クライアント) によってチェックされ、ファイル内にリストされているいずれかの証明書によって署名されているかどうか確認されます。これで、そのサーバが、自身が示しているサーバであるというクライアントからの信頼が構築されます。ACCESS ACCOUNT パラメータと IDENTIFIED BY パラメータは、そのクライアントが、自身が示しているクライアントであるという LDAP サーバからの信頼を構築します。

注意： TLS プロトコルではなくセキュア LDAP を使用する場合、TLS パラメータを OFF に設定する必要があります。

```

SET OPTION PUBLIC.login_mode = 'Standard,LDAPUA'
SET OPTION PUBLIC.trusted_certificates_file = '/mycompany/shared/trusted.txt'
CREATE LDAP SERVER secure_primary
SEARCH DN
    URL 'ldaps://my_LDAPserver:636/dc=MyCompany,dc=com??sub?cn=*'
    ACCESS ACCOUNT 'cn=aseadmin, cn=Users, dc=mycompany, dc=com'
    IDENTIFIED BY 'Secret99Password'
AUTHENTICATION URL 'ldaps://my_LDAPserver:636/'
CONNECTION TIMEOUT 3000
TLS OFF
WITH ACTIVATE

```

- 例 3** – ポート 389 で TLS プロトコルを構築します。また、データベースセキュリティオプション TRUSTED_CERTIFICATE_FILE はファイル名を指定して設定する必要があります。例 2 と同じセキュリティタイプを提供します。この例では、LDAP サーバベンダのサポート対象を拡大するため、TLS プロトコルは ON にします。

注意： SAP Sybase IQ サーバでセキュア LDAP または TLS を設定する方法を決定する際、使用するすべての LDAP サーバの要件を確認します。

```

SET OPTION PUBLIC.login_mode = 'Standard,LDAPUA'
SET OPTION PUBLIC.trusted_certificates_file = '/mycompany/shared/

```

SQL 文

```
trusted.txt'  
CREATE LDAP SERVER tls_primary  
SEARCH DN  
    URL 'ldap://my_LDAPserver:389/dc=MyCompany,dc=com??sub?cn=*'  
    ACCESS ACCOUNT 'cn=aseadmin, cn=Users, dc=mycompany, dc=com'  
    IDENTIFIED BY 'Secret99Password'  
AUTHENTICATION URL 'ldap://my_LDAPserver:389/'  
CONNECTION TIMEOUT 3000  
TLS ON  
WITH ACTIVATE
```

標準

(先頭に戻る) (188 ページ)

ANSI SQL – 準拠レベル：Transact-SQL 拡張。

パーミッション

(先頭に戻る) (188 ページ)

MANAGE ANY LDAP SERVER システム権限が必要です。

CREATE LOGICAL SERVER 文

ユーザ定義の論理サーバを作成します。次の文では、論理サーバで共有される物理ノード間で、一貫した共有システムテナポラリストア設定が強制的に適用されます。

クイックリンク：

「パラメータ」 (192 ページ)

「例」 (194 ページ)

「パーミッション」 (194 ページ)

構文

```
CREATE LOGICAL SERVER logical-server-name [  
    { ls-create-clause, ... } ] [ WITH STOP SERVER ]
```

ls-create-clause - (back to Syntax)

```
{ MEMBERSHIP ( { ls-member, ... } ) | POLICY ls-policy-name }
```

ls-member - (back to ls-create-clause)

```
FOR LOGICAL COORDINATOR | mpx-server-name
```

パラメータ

(先頭に戻る) (192 ページ)

- **logical-server-name** – 次を除く任意のユーザ定義の識別子
 - ALL
 - AUTO
 - COORDINATOR
 - DEFAULT
 - NONE
 - OPEN
 - SERVER
- **MEMBERSHIP** – コーディネータに対する論理的なメンバーシップを定義するには、MEMBERSHIP 句に FOR LOGICAL COORDINATOR を含めます。

論理サーバの作成時にメンバーを指定しないと、論理サーバは空で作成されません。

注意： OPEN 論理サーバや SERVER 論理サーバなどに対する暗黙的な論理サーバメンバーシップ定義が格納されることはありません。

SYS.ISSYSLOGICALMEMBER システムテーブルには、論理サーバメンバーシップの定義が格納されています。

ルート論理サーバポリシーの **ALLOW_COORDINATOR_AS_MEMBER** オプションを ON から OFF に変更しても、カタログに格納されているメンバーシップ情報に影響はありません。論理サーバ設定の有効性のみに影響します。

ALLOW_COORDINATOR_AS_MEMBER オプションが OFF に設定されている場合でも、マルチプレックスサーバ名を指定するか、FOR LOGICAL COORDINATOR 句を使用して、現在のコーディネータに対する論理サーバのメンバーシップを定義できます。メンバーシップ定義はカタログに格納されますが、マルチプレックスサーバがコーディネータとして動作している間は非アクティブです。

カタログには、論理サーバとそのメンバーシップの定義が格納されています。

- **POLICY** – 論理サーバとユーザ定義の論理サーバポリシーを関連付けます。POLICY 句が指定されていない場合、論理サーバはルートポリシーと関連付けられます。

SYS.ISSYSIQLOGICALSERVER システムテーブルには、対応する論理サーバの論理サーバポリシーについての情報が格納されています。

- **ls-policy-name** – ROOT を除く任意のユーザ定義の識別子です。
- **WITHSTOPSERVER** – TEMP_DATA_IN_SHARED_TEMP オプションが直接的または間接的に変更された場合、論理サーバ内のすべてのサーバを自動的に停止します。

適用対象

マルチプレックスのみ。

例

(先頭に戻る) (192 ページ)

- **例 1**–3つのマルチプレックスノードを含む、ユーザ定義の論理サーバ *ls1* をメンバーとして作成します。

```
CREATE LOGICAL SERVER ls1 MEMBERSHIP ( n1, n2, n3 )
```

- **例 2**–3つのメンバーノードを含む、ユーザ定義の論理サーバ *ls1* を作成し、論理サーバポリシー名 *lsp1* を定義します。

```
CREATE LOGICAL SERVER ls1 MEMBERSHIP ( w1_svr, w2_svr, r2_svr )
POLICY lsp1
```

- **例 3**–例 2 同様にサーバを作成します。ただし、TEMP_DATA_IN_SHARED_TEMP オプションが直接的または間接的に変更された場合、**WITH STOP SERVER** は論理サーバ内の全サーバを自動的に停止します。

```
CREATE LOGICAL SERVER ls1 MEMBERSHIP ( w1_svr, w2_svr, r2_svr )
POLICY lsp1 WITH STOP SERVER
```

- **例 4**–論理サーバポリシー *lspolicy1* を指定して、メンバーノードが存在しないユーザ定義の論理サーバ *ls1* を作成します。

```
CREATE LOGICAL SERVER ls1 POLICY lspolicy1
```

- **例 5**–ここで、*n1* は現在のコーディネータです。マルチプレックスノードの名前付きメンバーシップ *n2* および *n3* と、コーディネータの論理メンバーシップを指定して論理サーバ *ls2* を作成します。また、*ls2* の論理サーバポリシーを *lspolicy2* に設定します。

```
CREATE LOGICAL SERVER ls2 POLICY
MEMBERSHIP FOR LOGICAL COORDINATOR
lspolicy1, n1, n2, n3 POLICY lspolicy2
```

パーミッション

(先頭に戻る) (192 ページ)

MANAGE MULTIPLEX システム権限が必要です。

CREATE LOGIN POLICY 文

ログインポリシーをデータベースに作成します。

クイックリンク：

「パラメータ」 (195 ページ)

「例」 (196 ページ)

「使用法」 (196 ページ)

「パーミッション」 (196 ページ)

構文

```
CREATE LOGIN POLICY policy-name policy-option
```

```
policy-option - (構文に戻る)
  policy-option-name = policy-option-value
```

```
policy-option-name - (back to policy-option)
```

```
AUTO_UNLOCK_TIME
| CHANGE_PASSWORD_DUAL_CONTROL
| DEFAULT_LOGICAL_SERVER
| LOCKED
| MAX_CONNECTIONS
| MAX_DAYS_SINCE_LOGIN
| MAX_FAILED_LOGIN_ATTEMPTS
| MAX_NON_DBA_CONNECTIONS
| PASSWORD_EXPIRY_ON_NEXT_LOGIN
| PASSWORD_GRACE_TIME
| PASSWORD_LIFE_TIME
| ROOT_AUTO_UNLOCK_TIME
| LDAP_PRIMARY_SERVER
| LDAP_SECONDARY_SERVER
| LDAP_AUTO_FAILBACK_PERIOD
| LDAP_FAILOVER_TO_STD
| LDAP_REFRESH_DN
```

```
policy-option-value - (back to policy-option)
  { UNLIMITED | DEFAULT | value }
```

パラメータ

(先頭に戻る) (194 ページ)

- **policy-name** – ログインポリシーの名前。ルートを指定してルートログインポリシーを修正します。
- **policy-option-name** – ポリシーオプションの名前。各オプションの詳細については、「ログインポリシーオプション」と「LDAP ログインポリシーオプション」を参照してください。
- **policy-option-value** – ログインポリシーオプションに割り当てられる値。UNLIMITED を指定すると、制限は使用されません。DEFAULT を指定すると、デフォルトの制限が使用されます。各オプションでサポートされている値につ

いては、「ログインポリシーオプション」と「LDAP ログインポリシーオプション」を参照してください。

適用対象

シンプレックスとマルチプレックス。

例

(先頭に戻る) (194 ページ)

- **例 1** – Test1 ログインポリシーを作成します。このログインポリシーでは、パスワードは無期限で、アカウントがロックされるまでに許容されるユーザパスワードの入力回数が最大 5 回に設定されています。

```
CREATE LOGIN POLICY Test1
password_life_time=UNLIMITED
max_failed_login_attempts=5;
```

使用法

(先頭に戻る) (194 ページ)

ポリシーオプションを指定しない場合は、ルートログインポリシーからこのログインポリシーの値が取得されます。新しいポリシーは、MAX_NON_DBA_CONNECTIONS および ROOT_AUTO_UNLOCK_TIME ポリシーオプションを継承しません。

パーミッション

(先頭に戻る) (194 ページ)

MANAGE ANY LOGIN POLICY システム権限が必要です。

次のシステム権限は、記載されているログインポリシーオプションを上書きできます。

例外システム権限	ログインポリシーオプション
SERVER OPERATOR システム権限または DROP CONNECTION システム権限	MAX_NON_DBA_CONNS MAX_CONNECTIONS
MANAGE ANY USER システム権限	LOCKED MAX_DAYS_SINCE_LOGIN

ログインポリシーオプション

ルートログインポリシーとユーザ定義ログインポリシーで使用可能なオプションを次に示します。

オプション	説明
AUTO_UNLOCK_TIME	<p>MANAGE ANY USER システム権限が付与されていないアカウントがロックされてから自動的にロック解除されるまでの時間。このオプションは、ルートログインポリシーを含む任意のログインポリシーで定義できる。</p> <ul style="list-style-type: none"> • 値 - 0 ~ UNLIMITED • デフォルト - UNLIMITED • 適用対象 - MANAGE ANY USER システム権限が付与されていないすべてのユーザ。
CHANGE_PASSWORD_DUAL_CONTROL	<p>別のユーザのパスワードを変更するために、CHANGE PASSWORD システム権限が付与されている 2 人のユーザからの入力を要求する。</p> <ul style="list-style-type: none"> • 値 - ON、OFF • デフォルト - OFF • 適用対象 - すべてのユーザ。

オプション	説明
DEFAULT_LOGICAL_SERVER	<p>接続文字列で論理サーバが指定されていない場合、ユーザはユーザのログインポリシーで指定されている DEFAULT_LOGICAL_SERVER オプションに接続する。</p> <ul style="list-style-type: none"> • 値 – <ul style="list-style-type: none"> • 既存のユーザ定義論理サーバの名前。 • ALL – すべての論理サーバへのアクセスを許可する。 • AUTO – ルートログインポリシーのデフォルト論理サーバの値。 • COORDINATOR – 現在のコーディネータノード。 • NONE – あらゆるマルチプレックスサーバへのアクセスを拒否する。 • OPEN – 単独またはユーザ定義論理サーバの名前とともに使用する。どのユーザ定義論理サーバのメンバーでもないすべてのマルチプレックスノードへのアクセスを許可する。 • SERVER – SERVER 論理サーバのセマンティックに従って、すべてのマルチプレックスノードへのアクセスを許可する。 • デフォルト – AUTO • 適用対象 – すべてのユーザ。MANAGE MULTIPLEX システム権限が必要。
LOCKED	<p>ON に設定すると、ユーザは新しい接続を確立できない。この設定は一時的にログインポリシーユーザへのアクセスを拒否する。このオプションは、論理サーバに設定されたログインポリシーの上書きはできない。</p> <ul style="list-style-type: none"> • 値 – ON、OFF • デフォルト – OFF • 適用対象 – MANAGE ANY USER システム権限を持つユーザを除くすべてのユーザ。

オプション	説明
MAX_CONNECTIONS	<p>1 ユーザに許可される最大同時接続数。このオプションは、論理サーバごとの設定を指定できる。</p> <ul style="list-style-type: none"> • 値 - 0 ~ 2147483647 • デフォルト - UNLIMITED • 適用対象 - SERVER OPERATOR または DROP CONNECTION システム権限を持つユーザを除くすべてのユーザ。
MAX_DAYS_SINCE_LOGIN	<p>同一ユーザによる連続する 2 回のログインの間で許容される最大経過日数。</p> <ul style="list-style-type: none"> • 値 - 0 ~ 2147483647 • デフォルト - UNLIMITED • 適用対象 - MANAGE ANY USER システム権限を持つユーザを除くすべてのユーザ。
MAX_FAILED_LOGIN_ATTEMPTS	<p>前回のユーザアカウントへのログイン成功以降、アカウントがロックされるまでのログイン失敗の最大回数。</p> <ul style="list-style-type: none"> • 値 - 0 ~ 2147483647 • デフォルト - UNLIMITED • 適用対象 - すべてのユーザ。
MAX_NON_DBA_CONNECTIONS	<p>SERVER OPERATOR または DROP CONNECTION システム権限を持たないユーザが確立できる同時接続の最大数。このオプションは、ルートログインポリシーでのみサポートされる。</p> <ul style="list-style-type: none"> • 値 - 0 ~ 2147483647 • デフォルト - UNLIMITED • 適用対象 - SERVER OPERATOR または DROP CONNECTION システム権限を持つユーザを除くすべてのユーザ。

オプション	説明
PASSWORD_EXPIRY_ON_NEXT_LOGIN	<p>ON に設定すると、次のログイン時にユーザのパスワードの有効期限が切れる。</p> <ul style="list-style-type: none"> • 値 - ON、OFF • デフォルト - OFF • 適用対象 - すべてのユーザ。 <hr/> <p>注意： この機能は現在、SAP Control Center へのログイン時に実装されなくなっています。ユーザは、パスワードの変更を要求されません。ただし、SAP Control Center 外から(たとえば Interactive SQL を使用して) SAP Sybase IQ にログインする際には要求されます。</p>
PASSWORD_GRACE_TIME	<p>パスワードの有効期限が切れるまでの日数(ログインが可能だが、デフォルトの post_login プロシージャによって警告が発行される期間)。</p> <ul style="list-style-type: none"> • 値 - 0 ~ 2147483647 • デフォルト - 0 • 適用対象 - すべてのユーザ。
PASSWORD_LIFE_TIME	<p>パスワードの変更が必要となるまでの最大日数。</p> <ul style="list-style-type: none"> • 値 - 0 ~ 2147483647 • デフォルト - UNLIMITED • 適用対象 - すべてのユーザ。
ROOT_AUTO_UNLOCK_TIME	<p>MANAGE ANY USER システム権限が付与されているアカウントがロックされてから自動的にロック解除されるまでの時間。このオプションは、ルートログインポリシーでのみ定義できる。</p> <ul style="list-style-type: none"> • 値 - 0 ~ UNLIMITED • デフォルト - 15 • 適用対象 - MANAGE ANY USER システム権限が付与されているすべてのユーザ。

LDAP ログインポリシーオプション

LDAP ユーザ認証で使用可能なログインポリシーオプションを示します。

オプション	説明
LDAP_PRIMARY_SERVER	<p>プライマリ LDAP サーバの名前を指定する。</p> <ul style="list-style-type: none"> • 値 - 該当なし • デフォルト - なし • 適用対象 - すべてのユーザ。
LDAP_SECONDARY_SERVER	<p>セカンダリ LDAP サーバの名前を指定する。</p> <ul style="list-style-type: none"> • 値 - 該当なし • デフォルト - なし • 適用対象 - すべてのユーザ。
LDAP_AUTO_FAILBACK_PERIOD	<p>プライマリサーバへの自動フェールバックが試行されるまでの時間 (分単位) を指定する。</p> <ul style="list-style-type: none"> • 値 - 0 ~ 2147483647 • デフォルト - 15 分 • 適用対象 - すべてのユーザ。
LDAP_FAILOVER_TO_STD	<p>システムリソース、ネットワークの停止、接続のタイムアウト、または同様のシステム障害が原因で LDAP サーバによる認証に失敗した場合に、標準認証による認証を許可する。ただし、LDAP サーバから返された実際の認証の失敗を標準認証にフェールバックすることは許可しない。</p> <ul style="list-style-type: none"> • 値 - ON、OFF • デフォルト - ON • 適用対象 - すべてのユーザ。

オプション	説明
LDAP_REFRESH_DN	<p>ISYSLOGINPOLICYOPTION システムテーブル内の ldap_refresh_dn の値を協定世界時 (UTC) の現在時刻で更新する。</p> <p>ISYSLOGINPOLICYOPTION の ldap_refresh_dn の値が ISYSUSER の user_dn の値より新しい場合、LDAP によるユーザ認証のたびに新しいユーザ DN の検索が行われる。その場合、新しいユーザ DN で user_dn の値が更新され、現在時刻で user_dn_changed_at の値が再更新される。</p> <ul style="list-style-type: none"> • 値 - NOW • ROOT ポリシーの初期値 - NULL • ユーザ定義ログインポリシーの初期値 - 現在時刻を UTC で格納 • 適用対象 - すべてのユーザ。

マルチプレックスログインポリシーの設定

マルチプレックスサーバのログインポリシーを設定します。

例

この例では、論理サーバのログインポリシー設定が上書きされ、論理サーバ ls1 の最大接続数が増加します。

```
ALTER LOGIN POLICY lp1 max_connections=20 LOGICAL SERVER ls1;
```

使用法

マルチプレックスにのみ適用されます。

任意のマルチプレックスサーバ上で実行するログイン管理コマンドは、マルチプレックス内のすべてのサーバに自動的に伝達されます。最高のパフォーマンスを実現するには、これらのコマンドまたは DDL をコーディネータで実行します。

論理サーバレベルで上書きすると、特定のログインポリシーオプションが、論理サーバごとに設定が異なることとなります。

SYS.ISYSIQLSLOGINPOLICYOPTION には、論理サーバ上書きのためのログインポリシーオプション値が格納されています。ISYSIQLSLOGINPOLICYOPTION には、ログインポリシーオプションの論理サーバの上書きのそれぞれに対応するローが存在します。

CREATE LS POLICY 文

ユーザ定義の論理サーバのポリシーを作成します。次の文では、論理サーバで共有される物理ノード間で、一貫した共有システムテナポラリストア設定が強制的に適用されます。

クイックリンク：

「パラメータ」 (203 ページ)

「例」 (204 ページ)

「使用法」 (204 ページ)

「標準」 (206 ページ)

「パーミッション」 (206 ページ)

構文

```
CREATE LS POLICY ls-policy-name ls-option-value-list [ WITH STOP SERVER ]
```

```
ls-option-value-list - (back to Syntax)
  { ls-option-name = ls-policy-option-value } ...
```

```
ls-option-name - (back to ls-option-value-list)
```

```
ALLOW_COORDINATOR_AS_MEMBER
| DQP_ENABLED
| ENABLE_AUTOMATIC_FAILOVER
| LOGIN_REDIRECTION
| REDIRECTION_WAITERS_THRESHOLD
| TEMP_DATA_IN_SHARED_TEMP
```

パラメータ

(先頭に戻る) (203 ページ)

- **ls-policy-name** – 論理サーバポリシーの名前。ポリシー名には root を除く任意の識別子が指定できます。
- **ls-option-value-list** – 論理サーバポリシーオプションの名前。「使用法 (204 ページ)」を参照してください。
- **ls-policy-option-value** – 指定されていないオプションの値はすべて、ルート論理サーバポリシーから継承されます。「使用法 (204 ページ)」を参照してください。

- **WITH STOP SERVER – TEMP_DATA_IN_SHARED_TEMP** オプションが直接的または間接的に変更された場合、論理サーバ内のすべてのサーバを自動的に停止します。

適用対象

マルチプレックスのみ。

例

(先頭に戻る) (203 ページ)

- **例 1 – lspolicy1** という名前のユーザ定義の論理サーバポリシーを作成します。

```
CREATE LS POLICY lspolicy1
ALLOW_COORDINATOR_AS_MEMBER=ON;
```

使用法

(先頭に戻る) (203 ページ)

より小さな IQ_SYSTEM_TEMP DB 領域が必要な場合は、TEMP_DATA_IN_SHARED_TEMP を ON に設定します。こうすることによって、テンポラリデータが IQ_SYSTEM_TEMP ではなく、IQ_SHARED_TEMP に書き込まれます。ただし、分散クエリ処理環境では、DQP_ENABLED と TEMP_DATA_IN_SHARED_TEMP の両方を ON に設定すると、IQ_SHARED_TEMP 内の追加データで SAN が飽和する可能性があります。これは、IQ_SHARED_TEMP に対する追加の I/O 処理が DQP のパフォーマンスに悪影響を及ぼす可能性があるためです。

オプション	説明
ALLOW_COORDINATOR_AS_MEMBER	<p>ROOT 論理サーバポリシーに対してのみ設定可能。ON (デフォルト) の場合は、コーディネータを任意のユーザ定義論理サーバのメンバーにできる。OFF の場合は、コーディネータを任意のユーザ定義論理サーバのメンバーとして使用できない。</p> <ul style="list-style-type: none"> • 値 – ON、OFF • デフォルト – ON

オプション	説明
DQP_ENABLED	<p>0 に設定すると、クエリ処理が分散されない。1 (デフォルト) に設定した場合、書き込み可能な共有テンポラリファイルが存在するかぎり、クエリ処理が分散される。2 に設定した場合、クエリ処理がネットワーク上で分散され、共有テンポラリストアは使用されない。</p> <ul style="list-style-type: none"> • 値 - 0、1、2 • デフォルト - 1
ENABLE_AUTO-MATIC_FAILOVER	<p>ROOT 論理サーバポリシーに対してのみ設定可能。ON は、指定のログインポリシーが適用される論理サーバの自動フェールオーバーを有効化。OFF (デフォルト) は、論理サーバレベルの自動フェールオーバーを無効にし、手動フェールオーバーを可能にする。デフォルト値の設定に戻すには DEFAULT を指定する。</p> <ul style="list-style-type: none"> • 値 - ON、OFF、DEFAULT • デフォルト - OFF
LOGIN_REDIRECTION	<p>ON は、指定のログインポリシーが適用される論理サーバのログインリダイレクションを有効化。OFF (デフォルト) は、論理サーバレベルのログインリダイレクションを無効にし、外部接続管理を可能にする。</p> <ul style="list-style-type: none"> • 値 - ON、OFF • デフォルト - OFF
REDIRECTION_WAITERS_THRESHOLD	<p>SAP Sybase IQ がこの論理サーバへの接続を別のサーバにリダイレクトする前に、キューに入れることが可能な接続数を指定する。任意の整数値に設定可能。デフォルトは 5。</p> <ul style="list-style-type: none"> • 値 - 整数 • デフォルト - 5

オプション	説明
TEMP_DATA_IN_SHARED_TEMP	<p>ON の場合、共有テンポラリストアに 1 つ以上の読み書きファイルが追加されていると、すべてのテンポラリテーブルデータと対象スクラッチデータが共有テンポラリストアに書き込まれる。このオプションを設定した後、または、読み書きファイルを共有テンポラリストアに追加した後は、すべてのマルチプレックスノードの再起動が必要 (共有テンポラリストアに読み書きファイルがない場合、または、ノードを再起動しなかった場合は、データが IQ_SYSTEM_TEMP に書き込まれる)。OFF (デフォルト) の場合は、すべてのテンポラリテーブルデータとスクラッチデータがローカルテンポラリストアに書き込まれる。</p> <ul style="list-style-type: none"> • 値 - ON、OFF • デフォルト - OFF

標準

(先頭に戻る) (203 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Adaptive Server ではサポートされていません。

パーミッション

(先頭に戻る) (203 ページ)

MANAGE MULTIPLEX システム権限が必要です。

CREATE MESSAGE 文 [T-SQL]

PRINT 文と **RAISERROR** 文で使用されるユーザ定義メッセージを SYSUSERMESSAGES システムテーブルに追加します。

クイックリンク：

「パラメータ」 (207 ページ)

「使用法」 (207 ページ)

「標準」 (207 ページ)

「パーミッション」 (207 ページ)

構文

```
CREATE MESSAGE message-number  
... AS 'message-text'
```

パラメータ

(先頭に戻る) (206 ページ)

- **message_number** – 追加するメッセージのメッセージ番号。ユーザ定義メッセージのメッセージ番号は 20000 以上にしてください。
- **message_text** – 追加するメッセージのテキスト。最大長は、255 バイトです。**PRINT** 文と **RAISERROR** 文はメッセージ内のプレースホルダを認識して出力します。1つのメッセージには、ユニークなプレースホルダを任意の順序で 20 個まで含めることができます。これらのプレースホルダは、メッセージのテキストをクライアントに送信するときにメッセージの後に付けた任意の引数のフォーマット文字列に置換されます。

メッセージを別の文法構文の言語に変換するときに引数の順番を変えられるように、プレースホルダには番号が付けられます。引数のプレースホルダは "%nn!", すなわちパーセント記号 (%), 1 ~ 20 の整数、感嘆符 (!) の順に表示されます。整数は引数リスト内の引数の位置を表し、たとえば "%1!" は 1 番目の引数、"%2!" は 2 番目の引数などのように表されます。

使用法

(先頭に戻る) (206 ページ)

CREATE MESSAGE はメッセージ番号をメッセージ文字列に関連付けます。このメッセージ番号は **PRINT** 文と **RAISERROR** 文で使用できます。

sp_addmessage の *language* 引数に対応するパラメータはありません。

関連する動作：

- オートコミット。

標準

(先頭に戻る) (206 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - **CREATE MESSAGE** の機能は、Adaptive Server の **sp_addmessage** プロシージャによって提供されています。

パーミッション

(先頭に戻る) (206 ページ)

SQL 文

次のいずれかが必要です。

- CREATE MESSAGE システム権限
- CREATE ANY OBJECT システム権限

参照：

- PRINT 文 [T-SQL] (472 ページ)
- RAISERROR 文 [T-SQL] (476 ページ)

CREATE MULTIPLEX SERVER 文

マルチプレックスサーバを作成します。

クイックリンク：

「パラメータ」 (208 ページ)

「使用法」 (209 ページ)

「パーミッション」 (209 ページ)

構文

```
CREATE MULTIPLEX SERVER server-name DATABASE
  'dbfile' host-port-list [ ROLE { READER | WRITER } ]
  [ STATUS | { INCLUDED | EXCLUDED } ]
```

host-port-list

```
{ [ PRIVATE ] HOST 'hostname' PORT port number }
```

パラメータ

(先頭に戻る) (208 ページ)

- **PRIVATE** – 特定の HOST PORT ペアがプライベート相互接続用であることを指定します。MIPC (マルチプレックスプロセス間通信) 用の個別のプライベート相互接続を使用することにより、可用性が高く、パフォーマンスに優れたネットワーク設定を実現できます。SAP Sybase IQ は、プライベートポートを自動的に開きます。サーバの起動に使用される *host-port-list* にリストする必要はありません。すべてのパブリックポートとプライベートポートは、競合を避けるために、ユニークなポート番号を必要とします。
- **server-name** – サーバ起動オプション *-n* のルールに基づいた、マルチプレックスサーバの名前です。
- **ROLE** – 指定しない場合のデフォルトは、READER です。

- **STATUS** – 指定しない場合のデフォルトは、**INCLUDED** です。

適用対象

マルチプレックスのみ。

使用法

(先頭に戻る) (208 ページ)

サーバパスに UNIX のソフトリンク (シンボリックリンク) を使用することを計画している場合、ソフトリンクを作成してから **CREATE MULTIPLEX SERVER** を実行します。新しいサーバを起動する場合、データベースファイルパスは、サーバを作成したときに指定したデータベースファイルパスと一致している必要があります。

最初のマルチプレックスサーバの作成時に、コーディネータノードローとセカンダリノードローの両方が `SYS.ISYSIQMPXSERVER` に追加されます。この操作は、コーディネータノード用のコマンドとセカンダリノード用のコマンドの2つの異なる **CREATE MULTIPLEX SERVER** コマンドとしてトランザクションログに記録されます。

最初のセカンダリノードを作成すると、コーディネータが自動的に停止します。

`SYS.ISYSIQMPXSERVER` システムテーブルには、HOST ホスト名 PORT ポート名のペアが `host:port[;host:port...]` の形式で `connection_info` 文字列に格納されています。

注意： マルチプレックスサーバを実行中のコンピュータに、異なるネットワークアドレスにマップされている複数の冗長ネットワークカードがある場合は、複数の `host:port` ペアを使用してください。

`DATABASE`、`host-port list`、`ROLE`、`STATUS` の各句は、どのような順序で指定してもかまいません。

サーバの追加時にはコーディネータが実行中である必要がありますが、**CREATE MULTIPLEX SERVER** コマンドはマルチプレックス内のいずれのサーバからでも実行できます。

この文は自動的にコミットされます。

パーミッション

(先頭に戻る) (208 ページ)

`MANAGE MULTIPLEX` システム権限が必要です。

CREATE PROCEDURE 文

データベースに新しいユーザ定義の SQL プロシージャを作成します。

外部プロシージャインタフェースの作成方法については、「CREATE PROCEDURE 文 (外部プロシージャ)」を参照してください。

クイックリンク：

「パラメータ」 (210 ページ)

「例」 (214 ページ)

「使用法」 (215 ページ)

「標準」 (216 ページ)

「パーミッション」 (216 ページ)

構文

```
CREATE [ OR REPLACE | TEMPORARY ] PROCEDURE [ owner. ] procedure-name
( [ parameter, ... ] ) {
  [ SQL SECURITY { INVOKER | DEFINER } ]
  [ RESULT ( result-column, ... ) | NO RESULT SET ]
  [ ON EXCEPTION RESUME ] compound statement | AT location-string
```

parameter - (構文に戻る)

```
parameter_mode parameter-name data-type [ DEFAULT expression ] |
SQLCODE | SQLSTATE
```

parameter_mode - (back to parameter)

```
IN | OUT | INOUT
```

result-column - (構文に戻る)

```
column-name data-type
```

パラメータ

(先頭に戻る) (210 ページ)

- **parameter-name** - パラメータ名は、カラム名などの他のデータベース識別子に関するルールに従って付ける必要があります。また、有効な SQL データ型でなければなりません。キーワードには次の意味があります。

パラメータは、IN、OUT、INOUT のキーワードのいずれかをプレフィクスにできます。キーワードを指定しない場合、パラメータはデフォルトで INOUT になります。キーワードには次の意味があります。

- **IN** - パラメータは、プロシージャに値を渡す式です。
 - **OUT** - パラメータは、プロシージャから値を与えられる変数です。
 - **INOUT** - パラメータはプロシージャに値を渡し、プロシージャから新しい値を与えられる変数です。
- **SQLSTATE および SQLCODE** - プロシージャが終了するときに、SQLSTATE 値または SQLCODE 値を出力する特別なパラメータです (OUT パラメータです)。SQLSTATE パラメータと SQLCODE パラメータを指定しても指定しなくても、SQLSTATE と SQLCODE 特別値を常にプロシージャコールのすぐ後でチェックして、プロシージャのリターンステータスをテストすることができます。
- SQLSTATE と SQLCODE 特別値は、その次の SQL 文によって修正されます。SQLSTATE と SQLCODE をプロシージャ引数として与えると、リターンコードは変数の中に格納されます。

- **CREATE** - 新しいプロシージャを作成します。
- **OR REPLACE** - 既存のプロシージャを同じ名前でも置き換えます。この句によって、プロシージャの定義は変更されますが、既存のパーミッションは保持されます。

OR REPLACE 句をテンポラリプロシージャで使用することはできません。また、置き換えられるプロシージャがすでに使用されている場合は、エラーが返されます。

- **TEMPORARY** - 作成した接続でのみ参照できるストアドプロシージャになり、接続を削除するとストアドプロシージャも自動的に削除されます。テンポラリストアドプロシージャは明示的に削除することもできます。テンポラリストアドプロシージャに対して **ALTER**、**GRANT**、または **REVOKE** は実行できません。また他のストアドプロシージャとは異なり、テンポラリストアドプロシージャはカタログやトランザクションログに記録されません。

テンポラリプロシージャは、作成者 (現在のユーザ) または指定された所有者のパーミッションで実行されます。次の場合に、テンポラリプロシージャの所有者を指定できます。

- テンポラリプロシージャが永続的なストアドプロシージャ内に作成されている。
- テンポラリプロシージャと永続的なプロシージャの所有者が同じである。

テンポラリプロシージャの所有者を削除するには、最初にテンポラリプロシージャを削除します。

テンポラリストアドプロシージャを作成および削除できるのは、読み込み専用データベースに接続されている場合、つまり、外部プロシージャにすることができない場合です。

たとえば、次のテンポラリプロシージャは CustRank というテーブルがあればそれを削除します。この例では、テーブル名がユニークであり、プロシージャの作成者がテーブルの所有者を指定しなくても参照できると想定しています。

```
CREATE TEMPORARY PROCEDURE drop_table( IN @TableName char(128) )
BEGIN
  IF EXISTS ( SELECT 1 FROM SYS.SYSTAB WHERE
    table_name = @TableName )
  THEN EXECUTE IMMEDIATE
    'DROP TABLE "' || @TableName || "'';
  MESSAGE 'Table "' || @TableName ||
    '" dropped' to client;
  END IF;
END;
CALL drop_table( 'CustRank' )
```

- **RESULT** – 結果セットのカラムの数と型を宣言します。RESULT キーワードに続くカッコで囲まれたリストは、結果カラムの名前と型を定義します。CALL 文が記述されている場合、この情報は Embedded SQL DESCRIBE または ODBC SQLDescribeCol によって返されます。使用可能なデータ型については、『リファレンス：ビルディングブロック、テーブル、およびプロシージャ』の「SQL データ型」を参照してください。

一部のプロシージャでは、実行方法によっては複数の結果セットが生成される場合があります。たとえば、次のプロシージャは、2 カラムを返す場合も、1 カラムを返す場合もあります。

```
CREATE PROCEDURE names( IN formal char(1) )
BEGIN
  IF formal = 'n' THEN
    SELECT GivenName
    FROM Employees
  ELSE
    SELECT Surname, GivenName
    FROM Employees
  END IF
END
```

これらの結果セットプロシージャは RESULT 句を指定しないで記述するか、Transact-SQL で記述します。これらの使用には、次の制約があります。

- **Embedded SQL** – 正しい形式の結果セットを取得するには、結果セットのカーソルが開かれてからローが返されるまでの間に、プロシージャコールを記述 (DESCRIBE) する必要があります。DESCRIBE 文の CURSOR *cursor-name* 句は必須です。
- **ODBC、OLE DB、ADO.NET** – ODBC アプリケーションでは、変数結果セットプロシージャを使用できます。結果セットの記述は、ドライバまたはプロバイダによって実行されます。

- **Open Client アプリケーション** – 変数結果セットプロシージャは Open Client アプリケーションで使用できます。

プロシージャが返す結果セットが1つだけの場合は、**RESULT** 句を使用します。この句を使用すると、カーソルが開いた後で ODBC と Open Client アプリケーションが結果セットを記述し直すのを防ぐことができます。

複数の結果セットを処理するには、ODBC はプロシージャが定義した結果セットではなく、現在実行中のカーソルを記述する必要があります。したがって、ODBC はいつもプロシージャ定義の **RESULT** 句内で定義されているカラム名を記述するわけではありません。この問題を回避するには、結果セットを生成する **SELECT** 文でカラムエイリアスを使用します。

- **NO RESULT SET** – プロシージャが結果セットを返さないことを宣言します。この句は、プロシージャが結果セットを返さないことを外部環境から知る必要がある場合に役立ちます。
- **SQL SECURITY – INVOKER** (プロシージャを呼び出しているユーザ) または **DEFINER** (プロシージャを所有しているユーザ) のどちらとしてプロシージャが実行されるかを定義します。デフォルトは **DEFINER** です。

SQL SECURITY INVOKER を指定すると、プロシージャを呼び出す各ユーザに対して注釈を付ける必要があるため、メモリの使用量が増加します。また、名前解決も **INVOKER** として実行されます。そのため、適切な所有者名で、すべてのオブジェクト名 (テーブル、プロシージャなど) を修飾します。たとえば、`user1` が次のプロシージャを作成したとします。

```
CREATE PROCEDURE user1.myProcedure ()
  RESULT( columnA INT )
  SQL SECURITY INVOKER
  BEGIN
    SELECT columnA FROM table1;
  END;
```

`user2` がこのプロシージャを実行しようとしたときにテーブル `user2.table1` が存在しない場合、テーブルルックアップエラーが発生します。さらに、`user2.table1` が存在する場合は、目的の `user1.table1` の代わりにそのテーブルが使用されます。このような状況を防ぐには、文中でテーブル参照を修飾してください (単に `table1` とするのではなく、`user1.table1` とします)。

- **ON EXCEPTION RESUME** – プロシージャは **ON_TSQL_ERROR** オプションの設定に応じたアクションを実行します。**ON_TSQL_ERROR** オプションを **CONDITIONAL** (デフォルト) に設定すると、次の文がエラーを処理する場合は実行が継続され、そうでない場合は終了します。

エラー処理文には次の文があります。

- IF
- SELECT @variable =
- CASE
- LOOP
- LEAVE
- CONTINUE
- CALL
- EXECUTE
- SIGNAL
- RESIGNAL
- DECLARE
- SET VARIABLE

ON EXCEPTION RESUME 句では、明示的なエラー処理コードを使用しないでください。

「ON_TSQL_ERROR オプション [TSQL]」を参照してください。

- **AT location-string** – *location-string* に指定されたりモートプロシージャのプロキシストアプロシージャを現在のデータベース上に作成します。AT 句では、*location-string* 内のフィールドデリミタとしてセミコロン (;) を使用できます。セミコロンがない場合は、ピリオドがフィールドデリミタです。セミコロンを使用すると、データベースと所有者の各フィールドにファイル名と拡張子を使用できます。

例

(先頭に戻る) (210 ページ)

- **例 1** – CASE 文を使用してクエリ結果を分類します。

```
CREATE PROCEDURE ProductType (IN product_id INT, OUT type
CHAR(10))
BEGIN
    DECLARE prod_name CHAR(20) ;
    SELECT name INTO prod_name FROM "GROUP0"."Products"
    WHERE ID = product_id;
    CASE prod_name
    WHEN 'Tee Shirt' THEN
        SET type = 'Shirt'
    WHEN 'Sweatshirt' THEN
        SET type = 'Shirt'
    WHEN 'Baseball Cap' THEN
        SET type = 'Hat'
    WHEN 'Visor' THEN
        SET type = 'Hat'
    WHEN 'Shorts' THEN
        SET type = 'Shorts'
    ELSE
```

```

    SET type = 'UNKNOWN'
  END CASE ;
END

```

- **例2** – カーソルのロー上でカーソルとループを使用して、単一の値を返します。

```

CREATE PROCEDURE TopCustomer (OUT TopCompany CHAR(35), OUT
TopValue INT)
BEGIN
  DECLARE err_notfound EXCEPTION
    FOR SQLSTATE '02000' ;
  DECLARE curThisCust CURSOR FOR
  SELECT CompanyName, CAST(      sum(SalesOrderItems.Quantity *
  Products.UnitPrice) AS INTEGER) VALUE
  FROM Customers
  LEFT OUTER JOIN SalesOrders
  LEFT OUTER JOIN SalesorderItems
  LEFT OUTER JOIN Products
  GROUP BY CompanyName ;

  DECLARE ThisValue INT ;
  DECLARE ThisCompany CHAR(35) ;
  SET TopValue = 0 ;
  OPEN curThisCust ;
  CustomerLoop:
  LOOP
    FETCH NEXT curThisCust
    INTO ThisCompany, ThisValue ;
    IF SQLSTATE = err_notfound THEN
      LEAVE CustomerLoop ;
    END IF ;
    IF ThisValue > TopValue THEN
      SET TopValue = ThisValue ;
      SET TopCompany = ThisCompany ;
    END IF ;
  END LOOP CustomerLoop ;
  CLOSE curThisCust ;
END

```

使用法

(先頭に戻る) (210 ページ)

CREATE PROCEDURE はデータベースにプロシージャを作成します。プロシージャは **CALL** 文で呼び出します。永続的なストアードプロシージャまたはテンポラリ (TEMPORARY) ストアドプロシージャを作成できます。PROC は PROCEDURE の同意語として使用できます。

注意：ストアードプロシージャは、ISO/ANSI SQL と T-SQL という 2 とおりの方法で作成できます。たとえば、**CREATE PROCEDURE** 構文を使用する場合、**BEGIN TRANSACTION** は T-SQL でしか使用できません。ストアードプロシージャを作成する

際は、2種類の構文を混在させないでください。「CREATE PROCEDURE 文 [T-SQL]」を参照してください。

CALL を使用してプロシージャを実行する場合、必ずしもすべてのパラメータを指定する必要はありません。**CREATE PROCEDURE** 文の中にデフォルト値がある場合、パラメータが指定されていないと、デフォルト値が割り当てられます。**CALL** 文で引数が指定されておらず、デフォルトも設定されていない場合は、エラーが発生します。

リモートプロシージャが出力変数で返すことのできる文字数は最大 254 です。

すべての場合に返されるのではなくても、リモートプロシージャで結果セットが返される可能性がある場合は、ローカルプロシージャ定義に **RESULT** 句を含める必要があります。

リモートサーバについては、「CREATE SERVER 文」を参照してください。

関連する動作：

- オートコミット

標準

(先頭に戻る) (210 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - Transact-SQL の **CREATE PROCEDURE** 文は異なります。
- SQLJ - Java 結果セットの構文拡張は、推奨される SQLJ1 標準に指定されています。

パーミッション

(先頭に戻る) (210 ページ)

自分が所有する Watcom SQL プロシージャまたは Transact SQL プロシージャの場合、**CREATE PROCEDURE** システム権限が必要です。

任意のユーザが所有する Watcom SQL プロシージャまたは Transact SQL プロシージャの場合、次のいずれかが必要です。

- **CREATE ANY PROCEDURE** システム権限
- **CREATE ANY OBJECT** システム権限

自分が所有するリモートプロシージャの場合、次のすべてが必要です。

- **CREATE EXTERNAL REFERENCE** システム権限

- CREATE PROCEDURE システム権限

任意のユーザが所有するリモートプロシージャの場合、CREATE EXTERNAL REFERENCE システム権限が必要です。次のいずれかも必要です。

- CREATE ANY PROCEDURE システム権限
- CREATE ANY OBJECT システム権限

参照：

- プロシージャ内でのテンポラリテーブルの参照 (217 ページ)
- BEGIN ... END 文 (108 ページ)
- CALL 文 (115 ページ)
- CREATE PROCEDURE 文 [T-SQL] (218 ページ)
- CREATE PROCEDURE 文 (外部プロシージャ) (221 ページ)
- CREATE SERVER 文 (241 ページ)
- DROP 文 (319 ページ)
- EXECUTE IMMEDIATE 文 [ESQL] [SP] (352 ページ)
- GRANT EXECUTE 文 (391 ページ)
- RAISERROR 文 [T-SQL] (476 ページ)
- ON_TSQL_ERROR オプション [TSQL] (737 ページ)

プロシージャ内でのテンポラリテーブルの参照

プロシージャ間でテンポラリテーブルを共有すると、テーブル定義に不整合がある場合に問題が発生することがあります。

たとえば、procA と procB という 2 つのプロシージャがあり、その両方がテンポラリテーブル temp_table を定義して、sharedProc という名前の別のプロシージャを呼び出すとします。procA と procB のどちらもまだ呼び出されていないため、テンポラリテーブルはまだ存在しません。

ここで、procA の temp_table の定義が procB の定義と少し異なるとします。両方とも同じカラム名と型を使用していますが、カラムの順序が異なります。

procA を呼び出すと、予期した結果が返されます。しかし、procB を呼び出すと、異なる結果が返されます。

これは、procA が呼び出されたときに temp_table が作成され、その後 sharedProc が呼び出されたためです。sharedProc が呼び出されると、内部の **SELECT** 文が解析および検証され、その後、別の **SELECT** 文が実行されたときに使用できるように、解析された文の表現がキャッシュされます。キャッシュされたバージョンは、procA のテーブル定義のカラムの順序を反映しています。

SQL 文

procB を呼び出すと temp_table が再作成されますが、カラムの順序が異なります。procB が sharedProc を呼び出すと、データベースサーバは **SELECT** 文のキャッシュされた表現を使用します。そのため、結果が異なります。

次のいずれかを実行すると、このような状況の発生を防ぐことができます。

- このような方法で使用されるテンポラリテーブルは、一致するように定義する
- グローバルテンポラリテーブルの使用を検討する

CREATE PROCEDURE 文 [T-SQL]

Adaptive Server と互換性がある新しいプロシージャを作成します。

クイックリンク：

「パラメータ」 (218 ページ)

「使用法」 (218 ページ)

「標準」 (220 ページ)

「パーミッション」 (220 ページ)

構文

次の Transact-SQL の **CREATE PROCEDURE** 文のサブセットは SAP Sybase IQ でサポートされています。

```
CREATE [ OR REPLACE ] PROCEDURE [ owner.]procedure_name
... [ [ ( ] @parameter_name data-type [ = default ] [ OUTPUT ] [ , ... ]
[ ) ] ]
...[ WITH RECOMPILE ]
... AS
... statement-list
```

パラメータ

(先頭に戻る) (218 ページ)

- **CREATE** – 新しいプロシージャを作成します。
- **OR REPLACE** – 既存のプロシージャを同じ名前置き換えます。この句によって、プロシージャの定義は変更されますが、既存のパーミッションは保持されます。

使用法

(先頭に戻る) (218 ページ)

Transact-SQL の文と SAP Sybase IQ SQL の文の相違点は次のとおりです。

- **@をプレフィクスとする変数名** – "@" は Transact-SQL 変数名であることを示します。SAP Sybase IQ では、有効な識別子であればどれでも使用でき、@ プレフィクスは任意指定です。
- **入出力パラメータ** – SAP Sybase IQ のプロシージャパラメータは IN、OUT、または INOUT で指定されますが、Transact-SQL のプロシージャパラメータはデフォルトの INPUT パラメータか、OUTPUT パラメータで指定されます。SAP Sybase IQ で INOUT または OUT として宣言されるパラメータは、Transact-SQL では OUTPUT として宣言してください。
- **パラメータのデフォルト値** – SAP Sybase IQ のプロシージャパラメータには、キーワード DEFAULT を使用してデフォルト値を設定します。Transact-SQL では等号記号 (=) を使用してデフォルト値を設定します。
- **返される結果セット** – SAP Sybase IQ では、RESULT 句を使用して、返される結果セットを指定します。Transact-SQL プロシージャでは、最初のクエリのカラム名またはエイリアス名が呼び出し元の環境に返されます。

```
CREATE PROCEDURE showdept @deptname varchar(30)
AS
    SELECT Employees.Surname, Employees.givenName
    FROM Departments, Employees
    WHERE Departments.DepartmentName = @deptname
    AND Departments.DepartmentID =
        Employees.DepartmentID
```

対応する SAP Sybase IQ のプロシージャは次のとおりです。

```
CREATE PROCEDURE showdept(in deptname
    varchar(30) )
RESULT ( lastname char(20), firstname char(20))
ON EXCEPTION RESUME
BEGIN
    SELECT Employees.SurName, Employees.GivenName
    FROM Departments, Employees
    WHERE Departments.DepartmentName = deptname
    AND Departments.DepartmentID =
        Employees.DepartmentID
END
```

- **プロシージャ本体** – Transact-SQL プロシージャの本体は、AS キーワードをプレフィクスとして付けた Transact-SQL 文のリストです。SAP Sybase IQ プロシージャの本体は、BEGIN と END キーワードで囲まれた複合文です。

注意： ストアドプロシージャは、T-SQL と SQL/92 という 2 つの方法で作成できます。たとえば、**CREATE PROCEDURE** 構文を使用する場合、**BEGIN TRANSACTION** は T-SQL でしか使用できません。ストアドプロシージャを作成する際は、2 種類の構文を混在させないでください。

関連する動作：

SQL 文

- オートコミット。

標準

(先頭に戻る) (218 ページ)

- SQL - ISO/ANSI SQL 文法の Transact-SQL 拡張。
- SAP Sybase Database 製品 - SAP Sybase IQ は、Adaptive Server の **CREATE PROCEDURE** 文の構文のサブセットをサポートしています。
Transact-SQL の **WITH RECOMPILE** オプション句は、指定されていても無視されます。SQL Anywhere は、常にデータベース起動後に初めて実行されたプロシージャを再コンパイルし、コンパイルされたプロシージャをデータベースが停止するまで保管します。
プロシージャのグループはサポートされません。

パーミッション

(先頭に戻る) (218 ページ)

自分が所有する Watcom SQL プロシージャまたは Transact SQL プロシージャの場合、CREATE PROCEDURE システム権限が必要です。

任意のユーザが所有する Watcom SQL プロシージャまたは Transact SQL プロシージャの場合、次のいずれかが必要です。

- CREATE ANY PROCEDURE システム権限
- CREATE ANY OBJECT システム権限

自分が所有するリモートプロシージャの場合、次のすべてが必要です。

- CREATE EXTERNAL REFERENCE システム権限
- CREATE PROCEDURE システム権限

任意のユーザが所有するリモートプロシージャの場合、CREATE EXTERNAL REFERENCE システム権限が必要です。次のいずれかも必要です。

- CREATE ANY PROCEDURE システム権限
- CREATE ANY OBJECT システム権限

参照：

- CREATE PROCEDURE 文 (210 ページ)

CREATE PROCEDURE 文 (外部プロシージャ)

ネイティブプロシージャまたは外部プロシージャへのインタフェースを作成します。

Java UDF 向けの **CREATE PROCEDURE** 文のリファレンス情報については、「CREATE PROCEDURE 文 (Java UDF)」を参照してください。テーブル UDF 向けの **CREATE PROCEDURE** 文のリファレンス情報については、「CREATE PROCEDURE 文 (テーブル UDF)」を参照してください。

クイックリンク：

「パラメータ」 (222 ページ)

「使用法」 (227 ページ)

「標準」 (228 ページ)

「パーミッション」 (228 ページ)

構文

```
CREATE [ OR REPLACE ] PROCEDURE [ owner.]procedure-name ( [ parameter,
... ] )
```

```
[ SQL SECURITY { INVOKER | DEFINER } ]
[ RESULT ( result-column, ...) | NO RESULT SET ]
[ DYNAMIC RESULT SETS integer-expression ]
[ EXTERNAL NAME 'native-call'
  | EXTERNAL NAME 'c-call' LANGUAGE { C_ESQL32 | C_ESQL64 | C_ODBC32
  | C_ODBC64 }
  | EXTERNAL NAME 'perl-call' LANGUAGE PERL
  | EXTERNAL NAME 'php-call' LANGUAGE PHP
  | EXTERNAL NAME 'java-call' LANGUAGE JAVA ]
```

parameter - (構文に戻る)

```
parameter_mode parameter-name data-type [ DEFAULT expression ] |
SQLCODE | SQLSTATE
```

parameter_mode - (back to parameter)

```
IN | OUT | INOUT
```

result-column - (構文に戻る)

```
column-name data-type
```

native-call - (構文に戻る)

```
[ operating-system : ] function-name@library
```

c-call - (構文に戻る)

```
[ operating-system : ] function-name@library; ...
```

```

perl-call - (構文に戻る)
  <file=perl-file> $sa_perl_return = perl-subroutine( $sa_perl_arg0[, ... ] )

php-call - (構文に戻る)
  <file=php-file> print php-func( $argv[1][, ... ] )

java-call - (構文に戻る)
  [package-name.] class-name.method-name method-signature

operating-system - (back to native-call) or (back to c-call)
  Unix

method-signature - (back to java-call)
  ( [ field-descriptor, ... ] ) return-descriptor

field-descriptor and return-descriptor - (back to method-signature)
  { Z
    | B
    | S
    | I
    | J
    | F
    | D
    | C
    | V
    | [descriptor]
    | Lclass-name;
  }

```

パラメータ

(先頭に戻る) (221 ページ)

- **CREATE** – 新しいプロシージャを作成します。
- **OR REPLACE** – 既存のプロシージャを同じ名前で置き換えます。この句によって、プロシージャの定義は変更されますが、既存のパーミッションは保持されます。
- **parameter** – パラメータ名は、カラム名などの他のデータベース識別子に関するルールに従って付ける必要があります。また、有効な SQL データ型でなければなりません。キーワードには次の意味があります。

パラメータは、IN、OUT、INOUT のキーワードのいずれかをプレフィクスにできます。キーワードを指定しない場合、パラメータはデフォルトで INOUT になります。キーワードには次の意味があります。

- **IN** – パラメータは、プロシージャに値を渡す式です。
- **OUT** – パラメータは、プロシージャから値を与えられる変数です。

- **INOUT** – パラメータはプロシージャに値を渡し、プロシージャから新しい値を与えられる変数です。

注意： TABLE パラメータを INOUT または OUT として宣言することはできません。「CREATE PROCEDURE 文 (テーブル UDF)」を参照してください。

CALL を使用してプロシージャを実行する場合、必ずしもすべてのパラメータを指定する必要はありません。**CREATE PROCEDURE** 文の中にデフォルト値がある場合、パラメータが指定されていないと、デフォルト値が割り当てられます。**CALL** 文で引数が指定されておらず、デフォルトも設定されていない場合は、エラーが発生します。

注意： テーブル UDF を **CALL** で呼び出すことはできません。**CREATE PROCEDURE** 文を使用してください。

- **RESULT** – 結果セットのカラムの数と型を宣言します。RESULT キーワードに続くカッコで囲まれたリストは、結果カラムの名前と型を定義します。**CALL** 文が記述されている場合、この情報は Embedded SQL DESCRIBE または ODBC SQLDescribeCol によって返されます。使用可能なデータ型については、『リファレンス：ビルディングブロック、テーブル、およびプロシージャ』の「SQL データ型」を参照してください。

Embedded SQL (LANGUAGE C_ESQL32、LANGUAGE C_ESQL64) 外部プロシージャまたは ODBC (LANGUAGE C_ODBC32、LANGUAGE C_ODBC64) 外部プロシージャは、結果セットを返さないか、1つの結果セットを返すことができます。

Perl または PHP (LANGUAGE PERL、LANGUAGE PHP) 外部プロシージャは、結果セットを返すことができません。データベースサーバによってロードされるネイティブ関数を呼び出すプロシージャは、結果セットを返すことができません。

CLR または Java (LANGUAGE CLR、LANGUAGE JAVA) 外部プロシージャは、結果セットを返さないか、1つ以上の結果セットを返すことができます。

- **NO RESULT SET** – プロシージャが結果セットを返さないことを宣言します。この句は、プロシージャが結果セットを返さないことを外部環境から知る必要がある場合に役立ちます。
- **DYNAMIC RESULT SETS** – この句は、LANGUAGE CLR 呼び出しまたは LANGUAGE JAVA 呼び出しで使用します。この句は、LANGUAGE が指定されている場合にのみ有効です。RESULT 句を指定すると、DYNAMIC RESULT SETS がデフォルトで 1 に設定されます。RESULT 句を指定しないと、DYNAMIC RESULT SETS はデフォルトで 0 に設定されます。Perl または PHP (LANGUAGE PERL、LANGUAGE PHP) 外部関数を呼び出すプロシージャは結果セットを返すことができません。データベースサーバによってロードされる

ネイティブ関数を呼び出すプロシージャは、結果セットを返すことができません。

C_ESQL32、C_ESQL64、C_ODBC32、および C_ODBC64 外部環境も結果セットを返します (CLR や JAVA と同様) が、これらは 1 つの動的結果セットに制限されます。

Perl または PHP (LANGUAGE PERL、LANGUAGE PHP) 外部関数を呼び出すプロシージャは、結果セットを返すことができません。データベースサーバによってロードされるネイティブ関数を呼び出すプロシージャは、結果セットを返すことができません。

- **SQL SECURITY – INVOKER** (プロシージャを呼び出しているユーザ) または **DEFINER** (プロシージャを所有しているユーザ) のどちらとしてプロシージャが実行されるかを定義します。デフォルトは **DEFINER** です。外部呼び出しの場合、この句は、外部環境での修飾されていないオブジェクト参照に対して所有者のコンテキストを確立します。

SQL SECURITY INVOKER を指定すると、プロシージャを呼び出す各ユーザに対して注釈を付ける必要があるため、メモリの使用量が増加します。また、名前解決も **INVOKER** として実行されます。そのため、適切な所有者名で、すべてのオブジェクト名 (テーブル、プロシージャなど) を修飾します。たとえば、`user1` が次のプロシージャを作成したとします。

```
CREATE PROCEDURE user1.myProcedure ()
  RESULT( columnA INT )
  SQL SECURITY INVOKER
  BEGIN
    SELECT columnA FROM table1;
  END;
```

`user2` がこのプロシージャを実行しようとしたときにテーブル `user2.table1` が存在しない場合、テーブルルックアップエラーが発生します。さらに、`user2.table1` が存在する場合は、目的の `user1.table1` の代わりにそのテーブルが使用されます。このような状況を防ぐには、文中でテーブル参照を修飾してください (単に `table1` とするのではなく、`user1.table1` とします)。

- **EXTERNAL NAME – LANGUAGE** 属性を指定せずに **EXTERNAL NAME** 句を使用したプロシージャでは、C などのプログラミング言語で記述されたネイティブ関数へのインタフェースが定義されます。ネイティブ関数は、データベースサーバによってそのアドレス空間にロードされます。

ライブラリ名にはファイル拡張子が付く場合があります。この拡張子は通常、Windows では `.dll`、UNIX では `.so` です。拡張子がない場合、ライブラリに対するプラットフォーム固有のデフォルトのファイル拡張子が追加されます。正式に記述した例を次に示します。

```
CREATE PROCEDURE mystring( IN instr LONG VARCHAR )
EXTERNAL NAME
'mystring@mylib.dll;Unix:mystring@mylib.so';
```

プラットフォーム固有のデフォルトを使用して、前述の EXTERNAL NAME 句をより簡単に記述する方法を次に示します。

```
CREATE PROCEDURE mystring( IN instr LONG VARCHAR )
EXTERNAL NAME 'mystring@mylib';
```

関数が含まれているライブラリは、呼び出されると、データベースサーバのアドレス空間にロードされます。ネイティブ関数は、サーバの一部として実行されます。この場合、関数によって障害が引き起こされると、データベースサーバは終了します。したがって、外部環境での関数のロードと実行には、LANGUAGE 属性を使用することをおすすめします。外部環境で関数によって障害が引き起こされた場合でも、データベースサーバは動作し続けます。

- **EXTERNAL NAME c-call LANGUAGE { C_ESQL32 | C_ESQL64 | C_ODBC32 | C_ODBC64 }:** – コンパイルされたネイティブ C 関数をデータベースサーバ内ではなく外部環境で呼び出すには、EXTERNAL NAME 句の後に C_ESQL32、C_ESQL64、C_ODBC32、または C_ODBC64 のいずれかを指定した LANGUAGE 属性を続けて、ストアードプロシージャまたは関数を定義します。

LANGUAGE 属性が指定されると、その関数を含むライブラリが外部プロセスによってロードされ、外部関数とその外部プロセスの一部として実行されます。この場合、関数が原因で障害が発生しても、データベースサーバは実行し続けます。

次に、プロシージャ定義の例を示します。

```
CREATE PROCEDURE ODBCinsert(
  IN ProductName CHAR(30),
  IN ProductDescription CHAR(50)
)
NO RESULT SET
EXTERNAL NAME 'ODBCexternalInsert@extodbc.dll'
LANGUAGE C_ODBC32;
```

- **EXTERNAL NAME perl-call LANGUAGE CLR** – Perl 関数を外部環境で呼び出すには、プロシージャインタフェースを EXTERNAL NAME 句で定義し、それに続いて LANGUAGE PERL 属性を指定します。

Perl のストアードプロシージャまたは関数は、SQL のストアードプロシージャまたは関数と同じように動作します。ただし、例外として、プロシージャまたは関数のコードは Perl で記述され、プロシージャまたは関数の実行はデータベースサーバの外部 (つまり、Perl 実行プログラムインスタンス内) で行われます。

プロシージャ定義の例：

```
CREATE PROCEDURE PerlWriteToConsole( IN str LONG VARCHAR)
NO RESULT SET
```

```
EXTERNAL NAME '<file=PerlConsoleExample>
WriteToServerConsole( $sa_perl_arg0 )'
LANGUAGE PERL;
```

- **EXTERNAL NAME perl-call LANGUAGE PHP** – PHP 関数を外部環境で呼び出すには、プロシージャインタフェースを EXTERNAL NAME 句で定義し、それに続いて LANGUAGE PHP 属性を指定します。

PHP のストアードプロシージャまたは関数は、SQL のストアードプロシージャまたは関数と同じように動作します。ただし、例外として、プロシージャまたは関数のコードは PHP で記述され、プロシージャまたは関数の実行はデータベースサーバの外部（つまり、PHP 実行プログラムインスタンス内）で行われます。

プロシージャ定義の例：

```
CREATE PROCEDURE PHPPopulateTable()
NO RESULT SET
EXTERNAL NAME '<file=ServerSidePHPExample>
ServerSidePHPSub()'
LANGUAGE PHP;
```

- **EXTERNAL NAME java-call LANGUAGE JAVA** – Java メソッドシグネチャは、パラメータの型と戻り値の型を簡潔に文字で表現したものです。

Java メソッドを外部環境で呼び出すには、プロシージャインタフェースを **EXTERNAL NAME** 句で定義し、それに続いて LANGUAGE JAVA 属性を指定します。

Java インタフェースのストアードプロシージャまたは関数は、SQL のストアードプロシージャまたは関数と同じように動作します。ただし、例外として、プロシージャまたは関数のコードは Java で記述され、プロシージャまたは関数の実行はデータベースサーバの外部（つまり、Java 仮想マシン内）で行われます。

プロシージャ定義の例：

```
CREATE PROCEDURE HelloDemo( IN
name LONG VARCHAR )
NO RESULT SET
EXTERNAL NAME 'Hello.main([Ljava/lang/String;)V'
LANGUAGE JAVA;
```

Java メソッドの引数と戻り値の記述子には次の意味があります。

フィールドタイプ	Java データ型
B	byte
C	char
D	double
F	float

フィールドタイプ	Java データ型
I	int
J	long
L <i>class-name</i> ;	<i>class-name</i> クラスのインスタンス。クラス名は、完全に修飾された名前、ドットをスラッシュに置き換えたものとする。例: java/lang/String
S	short
V	void
Z	boolean
[配列の各次元ごとに 1 つ使用

次に例を示します。

```
double some_method(
    boolean a,
    int b,
    java.math.BigDecimal c,
    byte [][] d,
    java.sql.ResultSet[] d ) {
}
```

この例では、次のシグネチャを得られます。

```
'(ZILjava/math/BigDecimal; [[B[Ljava/sql/ResultSet;)]D'
```

使用法

(先頭に戻る) (221 ページ)

プロシージャの本体は複合文で構成されます。複合文の詳細については、「BEGIN ... END 文」を参照してください。

注意：ストアードプロシージャは、ISO/ANSI SQL と T-SQL という 2 とおりの方法で作成できます。たとえば、**CREATE PROCEDURE** 構文を使用する場合、**BEGIN TRANSACTION** は T-SQL でしか使用できません。ストアードプロシージャを作成する際は、2 種類の構文を混在させないでください。「CREATE PROCEDURE 文 [T-SQL]」を参照してください。

ストアードプロシージャが結果セットを返す場合、出力パラメータを設定したり戻り値を返したりすることはできません。

TEMPORARY 外部呼び出しプロシージャは作成できません。

複数のプロシージャからテンポラリテーブルを参照するとき、テンポラリテーブルの定義が矛盾していて、テーブルを参照している文がキャッシュされていると、問題が発生する可能性があります。

SQL 文

外部プロシージャまたはネイティブプロシージャを呼び出す永続的なストアードプロシージャの作成には、さまざまなプログラミング言語を使用できます。**PROC** を **PROCEDURE** の同意語として使用することができます。

関連する動作：

- オートコミット

標準

(先頭に戻る) (221 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - Transact-SQL の **CREATE PROCEDURE** 文は異なります。
- SQLJ - Java 結果セットの構文拡張は、推奨される SQLJ1 標準に指定されています。

パーミッション

(先頭に戻る) (221 ページ)

自分が所有する外部プロシージャの場合、次の権限が必要です。

- CREATE EXTERNAL REFERENCE システム権限
- CREATE PROCEDURE システム権限

任意のユーザが所有する外部プロシージャの場合、CREATE EXTERNAL REFERENCE システム権限が必要です。次のいずれかも必要です。

- CREATE ANY PROCEDURE システム権限
- CREATE ANY OBJECT システム権限

参照：

- ALTER PROCEDURE 文 (48 ページ)
- BEGIN ... END 文 (108 ページ)
- CALL 文 (115 ページ)
- CREATE PROCEDURE 文 (210 ページ)
- CREATE PROCEDURE 文 [T-SQL] (218 ページ)
- DROP 文 (319 ページ)
- EXECUTE IMMEDIATE 文 [ESQL] [SP] (352 ページ)
- GRANT EXECUTE 文 (391 ページ)

CREATE PROCEDURE 文 (Java UDF)

外部 Java テーブル UDF へのインタフェースを作成します。

外部プロシージャ向けの **CREATE PROCEDURE** 文のリファレンス情報については、「CREATE PROCEDURE 文 (外部プロシージャ)」を参照してください。テーブル UDF 向けの **CREATE PROCEDURE** 文のリファレンス情報については、「CREATE PROCEDURE 文 (テーブル UDF)」を参照してください。

クイックリンク：

「パラメータ」 (230 ページ)

「使用法」 (230 ページ)

「標準」 (230 ページ)

「パーミッション」 (230 ページ)

構文

構文 1 – 少なくとも 1 つの SAP Sybase IQ テーブルを参照するクエリの場合は次のとおりです。

```
CREATE[ OR REPLACE ] PROCEDURE
  [ owner.]procedure-name ( [ parameter, ...] )
  [ RESULT (result-column, ...)]
  [ SQL SECURITY { INVOKER | DEFINER } ]
  EXTERNAL NAME `java-call' [ LANGUAGE java ] }
```

構文 2 – カタログストアテーブルのみを参照するクエリの場合は次のとおりです。

```
CREATE[ OR REPLACE ] PROCEDURE
  [ owner.]procedure-name ( [ parameter, ...] )
  [ RESULT (result-column, ...)]
  | NO RESULT SET
  [ DYNAMIC RESULT SETS integer-expression ]
  [ SQL SECURITY { INVOKER | DEFINER } ]
  EXTERNAL NAME `java-call' [ LANGUAGE java ] }
```

parameter – (back to Syntax 1) or (back to Syntax 2)

```
[ IN parameter_mode parameter-name data-type
  [ DEFAULT expression ]
```

result-column – (back to Syntax 1) or (back to Syntax 2)

```
column-name data-type
```

java-call – (back to Syntax 1) or (back to Syntax 2)

```
'[ package-name.]class-name.method-name method-signature'
```

java – (back to Syntax 1) or (back to Syntax 2)

```
[ ALLOW | DISALLOW SERVER SIDE REQUESTS ]
```

パラメータ

(先頭に戻る) (229 ページ)

- **java** - DISALLOW がデフォルトです。ALLOW はサーバ側接続を許可することを示します。

注意： ALLOW は、必要な場合以外は指定しないでください。ALLOW を設定すると、特定の種類の SAP Sybase IQ テーブルジョインの速度が低下します。プロシージャ定義を ALLOW から DISALLOW に (またはその逆に) 変更した場合、新しい接続を確立するまではその変更は認識されません。

UDF を使用するとき **ALLOW SERVER SIDE REQUESTS** と **DISALLOW SERVER SIDE REQUESTS** の両方を同じクエリの中で指定しないでください。

使用法

(先頭に戻る) (229 ページ)

クエリが SAP Sybase IQ テーブルを参照する場合、カタログストアテーブルのみを参照するクエリとは構文とパラメータが異なることに注意します。

Java テーブル UDF は FROM 句でのみサポートされています。

Java テーブル関数では 1 つの結果セットのみを使用できます。Java テーブル関数を SAP Sybase IQ テーブルとジョインする場合や、SAP Sybase IQ テーブルのカラムが Java テーブル関数の引数である場合、単一の結果セットのみがサポートされます。

FROM 句の項目が Java テーブル関数のみの場合は、N 個の結果セットを使用できます。

標準

(先頭に戻る) (229 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - Transact-SQL の **CREATE PROCEDURE** 文は異なります。
- SQLJ - Java 結果セットの構文拡張は、推奨される SQLJ1 規格に指定されています。

パーミッション

(先頭に戻る) (229 ページ)

テンポラリプロシージャを作成する場合を除き、自分用のプロシージャを作成するには **CREATE PROCEDURE** システム権限を持っている必要があります。他の

ユーザの UDF プロシージャを作成するには、所有者を指定すること、および CREATE ANY PROCEDURES システム権限または CREATE ANY OBJECT システム権限を持っていることが必要です。プロシージャに外部参照を含む場合は、プロシージャの所有者がどうかに関係なく、前述のシステム権限に加えて CREATE EXTERNAL REFERENCE システム権限も必要です。

CREATE PROCEDURE 文 (テーブル UDF)

外部のテーブルユーザ定義関数 (テーブル UDF) に対するインタフェースを作成します。テーブル UDF を使用するための正規のライセンスを取得している必要があります。

外部プロシージャ向けの **CREATE PROCEDURE** 文のリファレンス情報については、「CREATE PROCEDURE 文 (外部プロシージャ)」を参照してください。Java UDF 向けの **CREATE PROCEDURE** 文のリファレンス情報については、「CREATE PROCEDURE 文 (Java UDF)」を参照してください。

クイックリンク：

「パラメータ」 (231 ページ)

「使用法」 (233 ページ)

「標準」 (234 ページ)

「パーミッション」 (234 ページ)

構文

```
CREATE [ OR REPLACE ] PROCEDURE
[ owner. ] procedure-name ( [ parameter [ , ... ] ] )
| RESULT result-column [ , ... ] )
[ SQL SECURITY { INVOKER | DEFINER } ]
EXTERNAL NAME 'external-call'
```

parameter - (構文に戻る)
 [IN] parameter-name data-type [DEFAULT expression]
 | [IN] parameter-name table-type

table-type - (back to parameter)
 TABLE (column-name data-type [, ...])

external-call - (構文に戻る)
 [column-name:] function-name@library; ...

パラメータ

(先頭に戻る) (231 ページ)

- **IN** – このパラメータは、スカラパラメータの値、または TABLE パラメータの値セットを UDF に指定するオブジェクトです。

注意： TABLE パラメータを INOUT または OUT として宣言することはできません。TABLE パラメータは 1 つのみ使用できます (位置は重要ではありません)。

- **OR REPLACE – OR REPLACE (CREATE OR REPLACE PROCEDURE)** を指定すると、新しいプロシージャが作成されるか、既存のプロシージャが同じ名前で置き換えられます。この句によって、プロシージャの定義は変更されますが、既存のパーミッションは維持されます。使用中のプロシージャを置き換えようとすると、エラーが返されます。
- **RESULT** – 外部 UDF の結果セットのカラムの名前とデータ型を宣言します。カラムのデータ型は有効な SQL データ型である必要があります (たとえば、結果セットのカラムでは TABLE をデータ型として使用できません)。結果に含まれる一連のデータは暗黙的に TABLE を意味します。外部 UDF ではデータ型 TABLE の結果セットを 1 つのみ使用できます。

注意： TABLE は出力値ではありません。テーブル UDF では結果セットに LONG VARBINARY および LONG VARCHAR データ型は使用できませんが、テーブルパラメータ関数 (TPF) では結果セットにラージオブジェクト (LOB) データを使用できます。

TPF では、LOB データを生成することはできませんが、結果セットに LOB データ型のカラムを持つことはできます。ただし、出力に LOB データ含めるには、入力テーブルのカラムを出力テーブルに渡す方法しかありません。これは describe の属性

EXTFNAPIV4_DESCRIBE_COL_VALUES_SUBSET_OF_INPUT を使用することで実現できます。例をサンプルファイル tpf_blob.cxx に示します。

- **SQL SECURITY** – プロシージャが INVOKER (UDF を呼び出しているユーザ) として実行されているか、または DEFINER (UDF を所有しているユーザ) として実行されているかを指定します。デフォルトは DEFINER です。

SQL SECURITY INVOKER が指定されている場合は、プロシージャを呼び出す各ユーザに対して注釈を付ける必要があるため、メモリ使用量が増えます。また、SQL SECURITY INVOKER が指定されている場合は、名前解決も INVOKER として実行されます。したがって、適切な所有者を使用して、すべてのオブジェクト名 (テーブルやプロシージャなど) を修飾するように注意してください。たとえば、user1 が次のプロシージャを作成したとします。

```
CREATE PROCEDURE user1.myProcedure ()
  RESULT( columnA INT )
  SQL SECURITY INVOKER
```

```
BEGIN
  SELECT columnA FROM table1;
END;
```

user2がこのプロシージャを実行しようとして、user2.table1 テーブルが存在しない場合、テーブルのロックアップでエラーが発生します。さらに、user2.table1 が存在する場合は、意図した user1.table1 ではなくそのテーブルが使用されます。このような状況を防ぐには、文においてテーブル参照を修飾します(単なる table1 ではなく、user1.table1 とします)。

- **EXTERNAL NAME** – 外部 UDF には、C などのプログラミング言語で記述された関数へのインタフェースを定義する **EXTERNAL NAME** 句が必要です。この関数は、データベースサーバによってそのアドレス空間にロードされます。

ライブラリ名にはファイル拡張子を付けることができます。通常、Windows では .dll、UNIX では .so です。拡張子がない場合、ライブラリに対するプラットフォーム固有のデフォルトのファイル拡張子が追加されます。形式の例を次に示します。

```
CREATE PROCEDURE mystring( IN instr CHAR(255),
  IN input_table TABLE(A INT) )
  RESULT (CHAR(255))
EXTERNAL NAME
'mystring@mylib.dll;Unix:mystring@mylib.so'
```

プラットフォーム固有のデフォルトを使用して、前述の **EXTERNAL NAME** 句をより簡単に記述する方法を次に示します。

```
CREATE PROCEDURE mystring( IN instr CHAR(255),
  IN input_table TABLE(A INT) )
  RESULT (CHAR(255))
EXTERNAL NAME 'mystring@mylib'
```

使用法

(先頭に戻る) (231 ページ)

テーブル UDF の定義には a_v4_extfn API を使用します。a_v3_extfn API および a_v4_extfn API を使用しない外部プロシージャ向けの **CREATE PROCEDURE** 文のリファレンス情報については、別のトピックを参照してください。Java UDF 向けの **CREATE PROCEDURE** 文のリファレンス情報については、別のトピックを参照してください。

CREATE PROCEDURE 文はデータベースにプロシージャを作成します。自分用のプロシージャを作成するには、**CREATE PROCEDURE** システム権限が必要です。他のユーザのプロシージャを作成するには、プロシージャの所有者を指定すること、および **CREATE ANY PROCEDURES** システム権限または **CREATE ANY OBJECT** システム権限を持っていることが必要です。プロシージャに外部参照が含まれる場合は、プロシージャの所有者が誰かに関係なく、前述のシステム権限に加えて **CREATE EXTERNAL REFERENCE** システム権限も必要です。

ストアドプロシージャが結果セットを返す場合、出力パラメータを設定したり戻り値を返したりすることはできません。

複数のプロシージャからテンポラリテーブルを参照するとき、テンポラリテーブルの定義が矛盾していて、テーブルを参照している文がキャッシュされていると、問題が発生する可能性があります。プロシージャ内でテンポラリテーブルを参照するときは注意してください。

CREATE PROCEDURE 文を使用して、SQL とは異なるプログラミング言語で実装された外部のテーブル UDF を作成できます。ただし、外部 UDF を作成する前に、テーブル UDF の制限について確認してください。

スカラパラメータ、結果カラム、および TABLE パラメータのカラムのデータ型は、有効な SQL データ型であることが必要です。

パラメータ名は、カラム名など他のデータベース識別子に関するルールに従っている必要があります。これらは、有効な SQL データ型である必要があります。

TPF では、複数のスカラパラメータの混在と、単一の TABLE パラメータがサポートされます。TABLE パラメータでは、UDF が処理する入力ローセットのスキーマを定義する必要があります。TABLE パラメータの定義にはカラム名とカラムのデータ型が含まれます。

```
TABLE (c1 INT, c2 CHAR(20))
```

上の例では、INT 型と CHAR(20) 型の 2 つのカラム c1 と c2 を使用してスキーマを定義しています。UDF が処理する各ローは 2 つの値の組である必要があります。スカラパラメータと違い、TABLE パラメータにデフォルト値を割り当てることはできません。

標準

(先頭に戻る) (231 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - Transact-SQL の **CREATE PROCEDURE** 文は異なります。
- SQLJ - Java 結果セットの構文拡張は、推奨される SQLJ1 規格に指定されていません。

パーミッション

(先頭に戻る) (231 ページ)

テンポラリプロシージャを作成する場合を除き、自分用の UDF を作成するには **CREATE PROCEDURE** システム権限が必要です。他のユーザの UDF を作成するには、プロシージャの所有者の指定、および **CREATE ANY PROCEDURE** システム権

限または CREATE ANY OBJECT システム権限が必要です。プロシージャに外部参照が含まれる場合は、前述のシステム権限に加えて CREATE EXTERNAL REFERENCE システム権限も必要です。

CREATE ROLE 文

新しいロールを作成したり、既存ユーザをロールとして使用できるように拡張したり、任意のロールのロール管理者を管理したりできます。

クイックリンク：

「パラメータ」 (235 ページ)

「例」 (236 ページ)

「使用法」 (237 ページ)

「標準」 (237 ページ)

「パーミッション」 (237 ページ)

構文

```
CREATE [ OR REPLACE ] ROLE { role_name | FOR USER userID }  
[ WITH ADMIN [ ONLY ] admin_name [...], [ SYS_MANAGE_ROLES_ROLE ]
```

パラメータ

(先頭に戻る) (235 ページ)

- **role_name** – OR REPLACE 句を使用する場合を除き、*role_name* をデータベースに既存の名前にはできません。
- **OR REPLACE** – *role_name* はデータベースにすでに存在している必要があります。*role_name* がまだ存在していない場合、新しいユーザ定義ロールが作成されます。現在のすべての管理者が、以下のように *admin_name* [...] 句に指定されている管理者で置換されます。
 - **WITH ADMIN OPTION** を付与され、新しいロール管理者リストに指定されていない既存のロール管理者はすべて、ロールに対する管理権限を持たないロールのメンバーになります。
 - **WITH ADMIN ONLY OPTION** を付与され、新しいロール管理者リストに指定されていない既存のロール管理者はすべて、ロールのメンバーから削除されます。

OR REPLACE 句を使用する際、新しいロール管理者リストに含まれている既存のロール管理者は、元の管理権限が置換権限より上位である場合、元の管理権限を保持します。たとえば、ユーザ A が、そのロールに対する WITH ADMIN

権限を付与されている既存のロール管理者であるとしします。新しいロール管理者には WITH ADMIN ONLY 権限が付与されます。ユーザ A がこのリストに含まれている場合、ユーザ A は上位である WITH ADMIN 権限を保持します。

- **FOR USER – OR REPLACE** なしで FOR USER 句を使用する場合、*userID* は、ロールとして現在機能できない既存ユーザの名前である必要があります。
- **admin_name** – ロールの管理者として指定するユーザのリスト。
- **WITH ADMIN** – 指定の各 *admin_name* に、そのロールと、その基礎となるすべてのシステム権限に対する管理権限が付与されます。SYS_MANAGE_ROLES_ROLE がリストに含まれている場合は、WITH ADMIN 句は無効になります。
- **WITH ADMIN ONLY** – 指定の各 *admin_name* に、基礎となるシステム権限ではなく、そのロールに対する管理権限が付与されます。
- **SYS_MANAGE_ROLES_ROLE** – グローバルロール管理者がロールを管理できるようにします。WITH ADMIN ONLY 句とともに指定できます。

例

(先頭に戻る) (235 ページ)

- **例 1** – ロール Sales を作成します。グローバルロール管理者のみがロールを管理できます。

```
CREATE ROLE Sales
```

- **例 2** – 既存ユーザ Jane をロールとして機能できるように拡張します。

```
CREATE OR REPLACE ROLE FOR USER Jane
```

- **例 3** – Mary と Jeff をロールの管理権限を持つロール管理者として、ロール Finance を作成します。グローバルロール管理者はこのロールを管理できません。

```
CREATE ROLE Finance
WITH ADMIN Mary, Jeff
```

- **例 3** – Mary と Jeff をロール管理者として、ロール Marketing を作成します。グローバルロール管理者もこのロールを管理できます。

```
CREATE ROLE Finance
WITH ADMIN ONLY Mary, Jeff, SYS_MANAGE_ROLES_ROLE
```

- **例 4** – Finance は Harry と Susan が管理権限を持つロール管理者である既存のロールです。Susan は管理者のままにして、Harry は置換し、グローバルロール管理者を追加します。新しいロール管理者は、管理権限のみを保有します。

次の文は、Susan を管理者として保持しますが、Susan はもともと付与されていた管理権限の方が高いためにロールに対する管理権限を保持します。Harry は管理権限のみの Bob と Sarah に置換され、グローバル管理者が追加されています。Harry は引き続きロールメンバーではありますが、管理権限は失います。

```
CREATE OR REPLACE ROLE Finance
WITH ADMIN ONLY Susan, Bob, Sarah, SYS_MANAGE_ROLE_ROLE
```

使用法

(先頭に戻る) (235 ページ)

ロール管理者 (*admin_name*) を指定し、グローバルロール管理者 (SYS_MANAGE_ROLES_ROLE) を含めない場合、グローバルロール管理者は新しいロールを管理できなくなります。したがって、作成時にはロール管理者を指定しないことをおすすめします。後で、OR REPLACE 句を使用して追加するようにしてください。

ADMIN 句を指定しない場合、デフォルトの WITH ADMIN ONLY 句が使用され、デフォルトの管理者はグローバルロール管理者 (SYS_MANAGE_ROLES_ROLE) になります。

ロール管理者を置換する際、ロールにグローバルロール管理者がいれば、新しいロール管理者リストにその管理者を含める必要があります。このようにしないと、ロールから削除されます。

ただし、WITH ADMIN 句を使用してロール管理者を付与する場合、この句はグローバルロール管理者に対しては無効なので、GRANT ROLE 文を使用して、グローバルロール管理者 (SYS_MANAGE_RILES_ROLE) をロールに追加しなおす必要があります。この付与に失敗すると、グローバルロール管理者はロールを管理できなくなります。

標準

(先頭に戻る) (235 ページ)

ANSI SQL – 準拠レベル：Transact-SQL 拡張。

パーミッション

(先頭に戻る) (235 ページ)

- 新しいロールを作成する場合 – MANAGE ROLES システム権限が必要です。
- OR REPLACE 句を使用する場合 – 置換するロールに対する管理権限とともに MANAGE ROLES システム権限が必要です。

CREATE SCHEMA 文

データベースユーザ用のテーブル、ビュー、パーミッションとそれらに関連するパーミッションのコレクションであるスキーマを作成します。

クイックリンク：

「[使用法](#)」 (238 ページ)

「[標準](#)」 (239 ページ)

「[パーミッション](#)」 (239 ページ)

構文

```
CREATE SCHEMA AUTHORIZATION userid
... [ { create-table-statement
      | create-view-statement
      | grant-statement } ] ...
```

使用法

([先頭に戻る](#)) (238 ページ)

userid には、現在の接続のユーザ ID を指定する必要があります。別のユーザに対するスキーマを作成することはできません。ユーザ ID の大文字小文字は区別されません。

CREATE SCHEMA 文に含まれるいずれかの文にエラーが発生すると、**CREATE SCHEMA** 文全体がロールバックされます。

CREATE SCHEMA 文を使用すると、個別の **CREATE** 文と **GRANT** 文を 1 つにまとめて一度に処理できます。データベース内に SCHEMA データベースオブジェクトは作成されません。オブジェクトを削除するには、個別の **DROP TABLE** 文または **DROP VIEW** 文を使用します。パーミッションを取り消すには、付与されているそれぞれのパーミッションに対して **REVOKE** 文を使用します。

注意： アクティブなマルチプレックスでは、**CREATE SCHEMA** 文は無効です。

個々の **CREATE** 文または **GRANT** 文は、文デリミタで区切りません。文デリミタは **CREATE SCHEMA** 文自身の末尾を区切ります。

それぞれの **CREATE** 文または **GRANT** 文は、まずオブジェクトを作成してから、それにパーミッションを付与するという順番に並べます。

1 ユーザに対して複数のスキーマを作成することはおすすめできません。また、今後のバージョンではサポートされなくなる可能性があります。

関連する動作：

- オートコミット。

標準

(先頭に戻る) (238 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - SAP Sybase IQ では、**CREATE SCHEMA** 文内での **REVOKE** 文の使用はサポートされていません。また、Transact-SQL のバッチまたはプロシージャ内での使用も許可されていません。

パーミッション

(先頭に戻る) (238 ページ)

CREATE ANY OBJECT システム権限が必要です。

参照：

- CREATE TABLE 文 (257 ページ)
- CREATE VIEW 文 (292 ページ)
- GRANT CREATE 文 (388 ページ)

CREATE SEQUENCE 文

複数のテーブル間でユニークな値となるプライマリキー値、およびテーブルのデフォルト値を生成するために使用できるシーケンスを作成します。この文は、SAP Sybase IQ のカタログストアテーブルのみに適用されます。

構文

```
CREATE [ OR REPLACE ] SEQUENCE [ owner. ] sequence-name
[ INCREMENT BY signed-integer ]
[ START WITH signed-integer ]
[ MINVALUE signed-integer | NO MINVALUE ]
[ MAXVALUE signed-integer | NO MAXVALUE ]
[ CACHE integer | NO CACHE ]
[ CYCLE | NO CYCLE ]
```

パラメータ

OR REPLACE 句 – OR REPLACE を指定すると、新しいシーケンスが作成されるか、同じ名前の既存のシーケンスが置き換えられます。OR REPLACE 句を使用しなかった場合、現在のユーザに対してすでに存在するシーケンスの名前を指定すると、エラーが返されます。

INCREMENT BY 句 – 最後に割り当てられた値から次のシーケンス値までの増分量を定義します。デフォルトは 1 です。負の値を指定すると、降順のシーケンスが生成されます。INCREMENT BY 値が 0 の場合は、エラーが返されます。

START WITH 句 – シーケンスの開始値を定義します。START WITH 句に値を指定しない場合、昇順シーケンスでは MINVALUE が使用され、降順シーケンスでは MAXVALUE が使用されます。START WITH の値が MINVALUE または MAXVALUE で指定された範囲外の場合は、エラーが返されます。

MINVALUE 句 – シーケンスで生成される最小値を定義します。デフォルトは 1 です。MINVALUE が $(2^{63}-1)$ より大きいか $-(2^{63}-1)$ より小さい場合は、エラーが返されます。また、MINVALUE が MAXVALUE より大きい場合も、エラーが返されます。

MAXVALUE 句 – シーケンスで生成される最大値を定義します。デフォルトは -1 です。MAXVALUE が $2^{63}-1$ より大きいか $-(2^{63}-1)$ より小さい場合は、エラーが返されます。

CACHE 句 – より速くアクセスできるようにメモリに保持される、事前に割り付けられたシーケンス値の数を指定します。キャッシュが不足すると、シーケンスキャッシュが再移植され、対応するエントリがトランザクションログに書き込まれます。キャッシュの現在の値は、チェックポイントの時点で ISYSEQUENCE システムテーブルに転送されます。デフォルトは 100 です。

CYCLE 句 – 最大値または最小値に達した後に、値の生成を継続するかどうかを指定します。

デフォルトは NO CYCLE です。この場合、最大値または最小値に達すると、エラーが返されます。

備考

シーケンスは、数値の自動生成が可能なデータベースオブジェクトです。シーケンスは特定またはユニークなテーブルカラムにバインドされず、シーケンスが適用されるテーブルカラム経由でのみアクセスできます。

シーケンスでは、次のいずれかの方法で値を生成できます。

単調な増加または減少を無制限に行う

ユーザ定義の制限まで単調な増加または減少を行い、停止する

ユーザ定義の制限まで単調な増加または減少を行い、循環して始めに返り、再び開始する

CYCLE 句を使用して、シーケンスで値を使い果たした場合の動作を制御します。

シーケンスが増加して MAXVALUE を上回り、CYCLE が指定されている場合には、次のシーケンス値として MINVALUE が使用されます。シーケンスが減少して MINVALUE を下回り、CYCLE が指定されている場合には、次のシーケンス値と

して MAXVALUE が使用されます。CYCLE が指定されていない場合は、エラーが返されます。

ビュー定義またはマテリアライズドビュー定義では、シーケンス値を使用できません。

権限

シーケンスを作成するには、CREATE ANY SEQUENCE または CREATE ANY OBJECT システム権限が必要です。

関連する動作

なし

標準と互換性

- **SQL/2008** – シーケンスは、SQL/2008 言語機能 T176 を構成します。SAP Sybase IQ では、シーケンスのデータ型にオプションの指定はできません。これを行うには、シーケンスで CAST を使用します。

また、以下はベンダー拡張です。

CACHE 句

OR REPLACE 構文

CURRVAL 式

DEFAULT 式でのシーケンスの使用

例

次の例は、4 で開始して 2 ずつ増加し、循環せず、15 個の値を一度にキャッシュする、Test という名前のシーケンスを作成します。

```
CREATE SEQUENCE Test
START WITH 4
INCREMENT BY 2
NO MAXVALUE
NO CYCLE
CACHE 15;
```

CREATE SERVER 文

サーバを ISYSSERVER テーブルに追加します。

クイックリンク：

「パラメータ」 (242 ページ)

「例」 (243 ページ)

「使用法」 (243 ページ)

「標準」 (243 ページ)

「パーミッション」 (243 ページ)

構文

```
CREATE SERVER server-name
  CLASS 'server-class'
  USING 'connection-info'
  [ READ ONLY ]

server-class - (構文に戻る)
{ ASAJDBC
  | ASEJDBC
  | SAODBC
  | ASEODBC
  | DB2ODBC
  | MSSODBC
  | ORAODBC
  | ODBC }
```

```
connection-info - (構文に戻る)
{ machine-name:port-number [ /dbname ] | data-source-name }
```

パラメータ

(先頭に戻る) (241 ページ)

- **USING** – JDBC ベースのサーバクラスを使用する場合、USING 句は *hostname:port-number[/dbname]* となります。
 - **hostname** – リモートサーバを実行するマシン。
 - **portnumber** – リモートサーバが受信する TCP/IP ポート番号。SAP Sybase IQ および SAP Sybase SQL Anywhere® のデフォルトのポート番号は 2638 です。
 - **dbname** – SQL Anywhere リモートサーバでは、*dbname* を指定しない場合、デフォルトのデータベースが使用されます。Adaptive Server の場合は、デフォルトが master データベースです。*dbname* を使用しない場合は、他の方法 (**FORWARD TO** 文など) を使って別のデータベースを指定します。

ODBC ベースのサーバクラスを使用する場合、USING 句は *data-source-name* になります。これは ODBC データソース名です。

- **READ ONLY** – リモートサーバが読み込み専用データソースであることを指定します。更新の要求は SAP Sybase IQ によって拒否されます。

例

(先頭に戻る) (241 ページ)

- **例 1** – JDBC ベースの Adaptive Server サーバ用のリモートサーバを `ase_prod` という名前で作成します。このマシン名は "banana" でポート番号は 3025 です。

```
CREATE SERVER ase_prod
CLASS 'asejdbc'
USING 'banana:3025'
```

- **例 2** – "apple" というマシンに、名前が `testasa` で、ポート番号 2638 で受信する SQL Anywhere のリモートサーバを作成します。

```
CREATE SERVER testasa
CLASS 'asajdbc'
USING 'apple:2638'
```

- **例 3** – Oracle サーバ用の `oracle723` という名前のリモートサーバを作成します。この ODBC データソース名は "oracle723" です。

```
CREATE SERVER oracle723
CLASS 'oraodbc'
USING 'oracle723'
```

使用法

(先頭に戻る) (241 ページ)

CREATE SERVER 文は、SAP Sybase IQ カタログからリモートサーバを定義します。

関連する動作

- オートコミット

標準

(先頭に戻る) (241 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - Open Client/Open Server でサポートされています。

パーミッション

(先頭に戻る) (241 ページ)

SERVER OPERATOR システム権限が必要です。

参照：

- ALTER SERVER 文 (54 ページ)
- DROP SERVER 文 (337 ページ)

CREATE SERVICE 文

データベースサーバが Web サーバとして機能することを許可します。

クイックリンク：

「パラメータ」 (244 ページ)

「例」 (246 ページ)

「使用法」 (246 ページ)

「標準」 (247 ページ)

「パーミッション」 (247 ページ)

構文

```
CREATE SERVICE service-name-string
  TYPE service-type-string
  [ attributes ] [
  AS statement ]
```

service-type-string - (構文に戻る)

```
{ 'RAW '
| 'HTML '
| 'XML '
| 'SOAP '
| 'DISH ' }
```

attributes - (構文に戻る)

```
[ AUTHORIZATION { ON | OFF } ]
[ SECURE { ON | OFF } ]
[ USER { user-name | NULL } ]
[ URL [ PATH/ ] { ON | OFF | ELEMENTS } ]
[ USING { SOAP-prefix | NULL } ]
```

パラメータ

(先頭に戻る) (244 ページ)

- **service-name-string** – Web サービス名には、英数字または "/"、"-","_",".","!","~","*","'", "("、")" を自由に並べて指定できます。ただし、最初の文字をスラッシュ (/) にしないでください。また、名前にスラッシュを 2 つ以上続けて使用することはできません。
- **AUTHORIZATION** – ユーザがサービスに接続する際、ユーザ名とパスワードの指定が必要かどうかを定義します。デフォルト値は ON です。

- 認証が OFF の場合、AS 句が必要となり、USER 句によって 1 人のユーザが識別されます。要求はすべて、そのユーザのアカウントとパーミッションを使用して実行されます。
- ON の場合は、すべてのユーザにユーザ名とパスワードの入力が必要になります。必要に応じて、USER 句を使用しユーザ名またはロール名を指定することにより、ユーザに対しサービスの使用許可を制限することもできます。ユーザ名が NULL の場合、既知のユーザはすべてサービスにアクセスできます。

運用システムは、認証を ON にして実行します。ユーザをロールに追加することにより、サービスを使用するためのパーミッションを付与します。

- **SECURE** – 安全でない接続を受け入れるかどうかを指定します。ON は、HTTPS 接続のみ受け入れることを意味します。HTTP ポートで受信したサービス要求は、HTTPS ポートへ自動的にリダイレクトされます。OFF に設定すると、HTTP と HTTPS の接続をどちらも受け入れます。デフォルト値は OFF です。
- **USER – AUTHORIZATION** が OFF の場合、このパラメータは必須となり、すべてのサービス要求の実行に使用するユーザ ID が指定されます。AUTHORIZATION が ON (デフォルト) であれば、この句はオプションとなり、サービスへのアクセスが許可されたユーザまたはロールが識別されます。デフォルト値は NULL で、すべてのユーザにアクセス許可が与えられます。
- **URL** – URI パスを受け入れるかどうか、また、受け入れる場合はどのように処理されるかを定義します。OFF の場合、URI 要求のサービス名の後に何も指定することができません。ON の場合は、URI の残りの部分が、*url* という名前の変数の値として解釈されます。ELEMENTS は、URI パスの残りの部分がスラッシュ文字で区切られて、最大 10 個の要素のリストで構成されることを指定します。値は、*url* の末尾に 1 から 10 までの数字を付けた名前を持つ変数に割り当てられます。たとえば、3 つ目までの変数の名前は *url1*、*url2*、*url3* です。指定された値が 10 に満たなければ、残りの変数は NULL に設定されます。サービス名が / の文字で終わっている場合、URL は OFF に設定する必要があります。デフォルト値は OFF です。
- **USING – DISH** サービスにだけ適用されます。パラメータには名前のプレフィックスを指定します。このプレフィックスで始まる名前の SOAP サービスだけが処理されます。
- **service-type-string** – サービスのタイプを識別します。タイプはリストされたサービスタイプのいずれかにする必要があります。デフォルト値はありません。

- **RAW** – 結果セットをそのままクライアントに送信します。プロシージャ内で必要なタグを明示的に生成することで、フォーマットされたドキュメントを作成できます。
- **HTML** – 文またはプロシージャの結果セットをテーブルを格納する HTML ドキュメントにフォーマットします。
- **XML** – 結果セットを XML フォーマットとして扱います。XML フォーマットではない場合は、自動的に XML RAW フォーマットに変換されます。
- **SOAP** – 結果セットを Simple Object Access Protocol (SOAP) 応答としてフォーマットします。有効な SOAP 要求を使用する必要があります。SOAP 標準の詳細については、www.w3.org/TR/SOAP を参照してください。
- **DISH** – SOAP ハンドラ、つまり DISH を指定します。このサービスは、1 つ以上の SOAP サービスのプロキシとして機能します。使用中は、複数の SOAP サービスへのアクセスを保持し、アクセス可能にするコンテナとして機能します。DISH に含まれる SOAP サービスのそれぞれに、Web Services Description Language (WSDL) ファイルが自動的に生成されます。内部の SOAP サービスは、共通のプレフィクスで識別されます。この識別子は、**USING** 句で指定する必要があります。
- **statement** – statement が NULL の場合、実行する文を URI に指定する必要があります。それ以外の場合は、指定された SQL 文だけがそのサービス内で実行可能になります。statement は SOAP サービスには必須ですが、DISH サービスでは無視されます。デフォルト値は NULL です。

運用システム内で実行されるすべてのサービスで statement を定義する必要があります。AUTHORIZATION が ON である場合のみ、statement を NULL にできます。

例

(先頭に戻る) (244 ページ)

- **例 1** – Web サーバをすばやく設定するには、`-xs` スイッチを指定してデータベースサーバを起動し、次の文を実行します。

```
CREATE SERVICE tables TYPE 'HTML'
AUTHORIZATION OFF USER DBA
AS SELECT * FROM SYS.ISYSTAB
```

この文を実行したら、Web ブラウザを使用して URL `http://localhost/tables` を開きます。

使用法

(先頭に戻る) (244 ページ)

この **CREATE SERVICE** 文を実行すると、データベースサーバが Web サーバとして動作するようになります。新しいエントリが `SYSWEBSERVICE` システムテーブルに作成されます。

マルチプレックスでは、Web サーバとして動作するコーディネータノードとセカンドリノードの両方で **CREATE SERVICE** を実行します。

標準

(先頭に戻る) (244 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - Adaptive Server ではサポートされていません。

パーミッション

(先頭に戻る) (244 ページ)

MANAGE ANY WEB SERVICE システム権限が必要です。

参照：

- ALTER SERVICE 文 (57 ページ)
- DROP SERVICE 文 (338 ページ)

CREATE SPATIAL REFERENCE SYSTEM 文

空間参照系を作成するか、置き換えます。

クイックリンク：

「パラメータ」 (248 ページ)

「例」 (253 ページ)

「使用法」 (254 ページ)

「標準」 (254 ページ)

「パーミッション」 (255 ページ)

構文

```
{ CREATE [ OR REPLACE ] SPATIAL REFERENCE SYSTEM
  | CREATE SPATIAL REFERENCE SYSTEM IF NOT EXISTS }
  srs-name
  [ srs-attribute ] [ srs-attribute ... ]
```

srs-attribute - (構文に戻る)

```

SRID srs-id
| DEFINITION { definition-string | NULL }
| ORGANIZATION { organization-name IDENTIFIED BY organization-srs-id
| NULL }
| TRANSFORM DEFINITION { transform-definition-string | NULL }
| LINEAR UNIT OF MEASURE linear-unit-name
| ANGULAR UNIT OF MEASURE { angular-unit-name | NULL }
| TYPE { ROUND EARTH | PLANAR }
| COORDINATE coordinate-name { UNBOUNDED | BETWEEN low-number
AND high-number }
| ELLIPSOID SEMI MAJOR AXIS semi-major-axis-length { SEMI MINOR AXIS
semi-minor-axis-length
| INVERSE FLATTENING inverse-flattening-ratio }
| TOLERANCE { tolerance-distance | DEFAULT }
| SNAP TO GRID { grid-size | DEFAULT }
| AXIS ORDER axis-order
| POLYGON FORMAT polygon-format
| STORAGE FORMAT storage-format

grid-size - (back to srs-attribute)
DOUBLE : usually between 0 and 1

axis-order - (back to srs-attribute)
{ 'x/y/z/m' | 'long/lat/z/m' | 'lat/long/z/m' }

polygon-format - (back to srs-attribute)
{ 'CounterClockWise' | 'Clockwise' | 'EvenOdd' }

storage-format - (back to srs-attribute)
{ 'Internal' | 'Original' | 'Mixed' }

```

パラメータ

(先頭に戻る) (247 ページ)

- **OR REPLACE** – OR REPLACE を指定すると、空間参照系がデータベースにまだ存在しない場合は空間参照系が作成され、存在する場合は置き換えられます。使用中の空間参照系を置き換えようとする、エラーが返されます。OR REPLACE 句を指定しないでデータベースにすでに存在する空間参照系を置き換えようとしても、エラーが返されます。
- **IF NOT EXISTS** – CREATE SPATIAL REFERENCE IF NOT EXISTS を指定すると、指定された名前の空間参照系がデータベースにすでに存在するかどうかチェックされます。存在しない場合は、データベースサーバによって空間参照系が作成されます。存在する場合は、追加のアクションは行われず、エラーも返されません。
- **IDENTIFIED BY** – 空間参照系の SRID (*srs-id*)。組織で *organization-srs-id* を指定して空間参照系を定義した場合は、*srs-id* をその値に設定してください。

IDENTIFIED BY 句が指定されていない場合は、デフォルトにより、ORGANIZATION 句または DEFINITION 句で定義された *organization-srs-id* に

SRID が設定されます。デフォルトの SRID として使用できる *organization-srs-id* がどちらの句にも指定されていない場合は、エラーが返されます。

空間参照系が周知の座標系に基づいておらず、異なる測地線解釈がされている場合は、1000000000 (10 億) に周知の値を加えた値に *srs-id* 値を設定します。たとえば、測地線の空間参照系 WGS 84 (ID 4326) の平面解釈の SRID は、1000004326 になります。

SAP Sybase IQ で提供される周知の座標系に基づいていない空間参照系には、SRID 0 を除き 2000000000 (20 億) 以上の SRID が付与されます。2000000000 ~ 2147483647 の範囲の SRID 値は SAP Sybase IQ で予約されているため、この範囲の SRID は作成しないでください。

OGC などの定義機関または他のベンダで予約されている SRID を選択する可能性を低減するため、0 ~ 32767 (EPSG で予約済み) または 2147483547 ~ 2147483647 の範囲の SRID は選択しないでください。

また、SRID は 32 ビットの符号付き整数として格納されるため、 $2^{31}-1$ または 2147483647 を超える数値は指定できません。

- **DEFINITION** – 座標系のデフォルトを設定または上書きします。DEFINITION 句以外の句で属性が設定されている場合は、DEFINITION 句の設定内容に関係なく、他の句で指定された値が優先されます。

definition-string は、SQL/MM と OGC で定義されるような、空間参照系の Well Known Text 構文の文字列です。たとえば、次のクエリは WGS 84 の定義を返します。

```
SELECT ST_SpatialRefSys::ST_FormatWKT( definition )
FROM ST_SPATIAL_REFERENCE_SYSTEMS
WHERE srs_id=4326;
```

Interactive SQL で戻り値をダブルクリックすると、読みやすい形で値が表示されます。

DEFINITION 句が指定されている場合は、*definition-string* が解析され、属性のデフォルト値の選択に使用されます。たとえば、*definition-string* に、*organization-name* と *organization-srs-id* を定義する AUTHORITY 要素が含まれる場合があります。

definition-string のパラメータ値は、SQL 文の句で明示的に設定した値で上書きされます。たとえば、ORGANIZATION 句が指定されていると、*definition-string* の ORGANIZATION の値が上書きされます。

- **ORGANIZATION** – 新しい空間参照系の基となる空間参照系を作成した組織に関する情報。

- **TRANSFORM DEFINITION** – 空間参照系に使用する変換の説明。現時点では、PROJ.4 変換のみがサポートされています。変換定義は、空間参照系間でデータを変換するときに ST_Transform メソッドで使用されます。一部の变換は、transform-definition-string が定義されていなくても使用できます。
- **LINEAR UNIT OF MEASURE** – 空間参照系の線形測定単位。指定する値は、ST_UNITS_OF_MEASURE システムビューで定義された線形測定単位と一致している必要があります。

この句が指定されておらず、DEFINITION 句で定義されていない場合、デフォルトは METRE です。事前に定義された測定単位をデータベースに追加するには、sa_install_feature システムプロシージャを使用します。

カスタム測定単位をデータベースに追加するには、CREATE SPATIAL UNIT OF MEASURE 文を使用します。

注意： METRE と METER はいずれも有効なスペルですが、SQL/MM 標準に準拠している METRE を使用することをおすすめします。

- **ANGULAR UNIT OF MEASURE** – 空間参照系の角度測定単位。指定する値は、ST_UNITS_OF_MEASURE システムテーブルで定義された角度測定単位と一致している必要があります。

この句が指定されておらず、DEFINITION 句で定義されていない場合、デフォルトは、地理的空間参照系の場合は DEGREE、非地理的空間参照系の場合は NULL です。

角度測定単位は、地理的空間参照系の場合は NULL 以外、非地理的空間参照系の場合は NULL にしてください。

角度測定単位は、地理的空間参照系の場合は NULL 以外、非地理的空間参照系の場合は NULL にしてください。事前に定義された測定単位をデータベースに追加するには、sa_install_feature システムプロシージャを使用します。

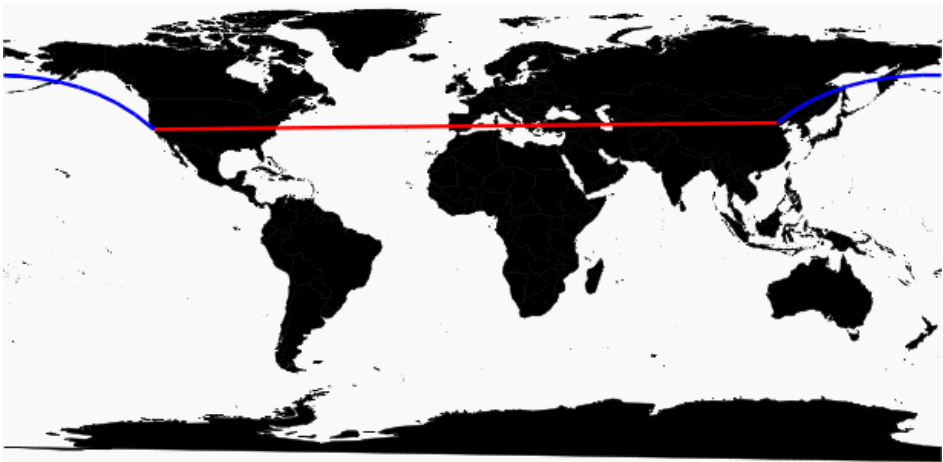
カスタム測定単位をデータベースに追加するには、CREATE SPATIAL UNIT OF MEASURE 文を使用します。

- **TYPE** – SRS で点と点を結ぶ線を解釈する方法を制御します。地理的空間参照系では、TYPE 句で ROUND EARTH (デフォルト) または PLANAR のいずれかを指定できます。ROUND EARTH モデルでは、点と点を結ぶ線が大楕円弧として解釈されます。地表面の 2 点を指定し、この 2 点および地球の中心と交差するように平面を選択したとします。この平面は地球と交差し、2 点を結ぶ線はこの交差に沿った最短距離になります。

2 点がまったく反対側にある場合、2 点および地球の中心と交差する単一でユニークな平面は存在しません。このような正反対の点を結ぶ線セグメントは無効であり、ROUND EARTH モデルでエラーになります。

ROUND EARTH モデルでは、地球が回転楕円体と見なされ、地球の湾曲に沿った線が選択されます。場合によっては、2 点間の線が x = 経度、 y = 緯度の正距円筒図法における直線として解釈される平面モデルの使用が必要となる場合があります。

次の例では、青い線は ROUND EARTH モデルで使用される線の解釈を示し、赤い線は対応する PLANAR モデルを示します。



PLANAR モデルは、他の製品で使用される解釈と一致させるために使用できます。また、ROUND EARTH モデルでサポートされていないメソッド (ST_Area、ST_ConvexHull など) には制限があり、一部のメソッドは部分的にのみサポートされている (ST_Distance はポイントジオメトリでのみサポートされる) ため、PLANAR モデルも便利です。円ストリングに基づいたジオメトリは、ROUND EARTH 空間参照系ではサポートされていません。

非地理的 SRS の場合には、タイプを PLANAR にしてください (これが、TYPE 句が指定されておらず、DEFINITION 句が指定されていないか非地理的定義を使用している場合のデフォルトです)。

- **COORDINATE** – 空間参照系の次元の境界。coordinate-name は、空間参照系で使用される座標系の名前です。非地理的座標系の場合は、coordinate-name に x 、 y 、または m を使用できます。地理的座標系の場合は、coordinate-name に LATITUDE、LONGITUDE、 z 、または m を使用できます。

次元で境界を配置しない場合は、UNBOUNDED を指定します。境界の上限と下限を設定するには、BETWEEN 句を使用します。

X 座標と Y 座標には、関連する境界が必要です。地理的座標系の場合は、これらの設定を COORDINATE 句で上書きしないかぎり、デフォルトにより経度座標は -180 ~ 180 度の間、緯度座標は -90 ~ 90 の間になります。非地理的座標系の場合は、X 座標と Y 座標の両方について CREATE 文で境界を指定してください。

LATITUDE と LONGITUDE は、地理的座標系に使用されます。LATITUDE と LONGITUDE の境界は、指定がなければデフォルトにより地球全体になります。

- **ELLIPSOID** – タイプ ROUND EARTH の空間参照系の楕円として、地球を表現するときに使用する値。DEFINITION 句が指定されている場合は、この句で楕円定義を指定できます。ELLIPSOID 句が指定されていると、このデフォルトの楕円が上書きされます。

地球の自転が引き起こす扁平化により地球の中心から北極または南極までの距離が赤道までの距離より短くなるため、地球は完全な球体ではありません。このため、地球は、半長径 (中心から赤道までの距離) と半短径 (中心から極までの距離) に異なる値を使用する楕円としてモデル化されます。半長径と逆扁平率を使用して楕円を定義するのが最も一般的ですが、代わりに半短径を使用して指定することもできます (たとえば、完全な球形を使用して地球を近似させる場合には、この方法を使用します)。半長径と半短径は空間参照系の線形単位で定義され、逆扁平率 (1/f) は次の式で求められる比率です。

$$1/f = (\text{semi-major-axis}) / (\text{semi-major-axis} - \text{semi-minor-axis})$$

product-name では、地理的空間参照系で距離を計算するときに楕円定義が使用されます。

- **SNAP TO GRID** – 平らな地球 (平面) の空間参照系では、SNAP TO GRID 句を使用して、SAP Sybase IQ で計算に使用されるグリッドサイズを定義します。デフォルトでは、X と Y の空間の境界におけるすべての点で 12 有効桁数を格納できるように SAP Sybase IQ でグリッドサイズが選択されます。たとえば、空間参照系の境界が X は -180 ~ 180 の間、Y は -90 ~ 90 の間の場合、グリッドサイズ 0.000000001 (1E-9) が選択されます。
- **TOLERANCE** – 平らな地球 (平面) の空間参照系では、TOLERANCE 句を使用して、点を比較するときに使用する精度を指定します。2 点間の距離が tolerance-distance より小さい場合、その 2 点は同じと見なされます。tolerance-distance を指定することにより、入力データまたは制限された内部精度の不正確さに対する許容度を制御できます。デフォルトでは、tolerance-distance は grid-size と等しくなるように設定されます。

0 に設定すると、2 点が完全に等しい場合にのみ同じであると見なされます。

曲面の空間参照系では、TOLERANCE を 0 に設定してください。

- **POLYGON FORMAT** – SAP Sybase IQ では、まず、多角形を構成するリングの方向によって多角形が解釈されます。定義された点の順序でリングを移動した場合、多角形の内側がリングの左側になります。平面と曲面の空間参照系では、同じルールが適用されます。

SAP Sybase IQ で使用される解釈は一般的ですが、汎用的な解釈ではありません。正反対の方向を使用する製品もあれば、多角形の解釈にリングの方向を使用しない製品もあります。POLYGON FORMAT 句を使用して、必要に応じて入力データと一致する多角形の解釈を選択できます。次の値がサポートされます。

- **CounterClockwise** – 入力 は SAP Sybase IQ の内部解釈に従います。リングの方向に従って、多角形の内側が左側になります。
- **Clockwise** – 入力 は SAP Sybase IQ の内部解釈と反対になります。リングの方向に従って、多角形の内側が右側になります。
- **EvenOdd** – (デフォルト) リングの方向は無視され、多角形の内部は代わりにリングのネストによって決定されます。外部のリングは最大のリングとなり、内部のリングはこのリングの内側の小さなリングとなります。斜線は、リング内の点から、すべてのリングを交差して外側に向かってトレースされます。交差するリングの数が偶数の場合は、外部リングです。奇数の場合は、内部リングです。
- **STORAGE FORMAT** – 空間データをデータベースにロードするときの格納内容を制御します。考えられる値は、次のとおりです。
 - **Internal** – SAP Sybase IQ は正規化された表現のみを格納します。元の入力特性を再現する必要がない場合に指定します。これは、平面 (TYPE PLANAR) の空間参照系のデフォルトです。
 - **Original** – SAP Sybase IQ は元の表現のみを格納します。元の入力特性を再現できますが、格納された値のすべての操作に対して正規化の手順を繰り返す必要があり、データの操作が遅くなる可能性があります。
 - **Mixed** – SAP Sybase IQ は内部バージョンを格納します。元のバージョンと異なる場合、SAP Sybase SQL Anywhere® は元のバージョンも格納します。両方のバージョンを格納すると、元の表現特性を再現でき、格納された値の操作に対して正規化の手順を繰り返す必要はありません。ただし、ジオメトリごとに 2 つの表現が格納される可能性があるため、記憶領域の要件が大幅に増加します。Mixed は、曲面の空間参照系 (TYPE ROUND EARTH) のデフォルトフォーマットです。

例

(先頭に戻る) (247 ページ)

- **例 1** – 空間参照系 mySpatialRS を作成します。

```
CREATE SPATIAL REFERENCE SYSTEM "mySpatialRS"
IDENTIFIED BY 1000026980
LINEAR UNIT OF MEASURE "metre"
TYPE PLANAR
COORDINATE X BETWEEN 171266.736269555 AND 831044.757769222
COORDINATE Y BETWEEN 524881.608973277 AND 691571.125115319
DEFINITION 'PROJCS["NAD83 / Kentucky South",
GEOGCS["NAD83",
DATUM["North American Datum 1983",
SPHEROID["GRS 1980",
6378137,298.257222101,AUTHORITY["EPSG","7019"]],
AUTHORITY["EPSG","6269"]],
PRIMEM["Greenwich",0,AUTHORITY["EPSG","8901"]],
UNIT["degree",0.01745329251994328,AUTHORITY["EPSG","9122"]],
AUTHORITY["EPSG","4269"]],
UNIT["metre",1,AUTHORITY["EPSG","9001"]],
PROJECTION["Lambert Conformal Conic 2SP"],
PARAMETER["standard_parallel_1",37.93333333333333],
PARAMETER["standard_parallel_2",36.73333333333333],
PARAMETER["latitude_of_origin",36.33333333333334],
PARAMETER["central_meridian",-85.75],
PARAMETER["false_easting",500000],
PARAMETER["false_northing",500000],
AUTHORITY["EPSG","26980"],
AXIS["X",EAST],
AXIS["Y",NORTH]]'
TRANSFORM DEFINITION '+proj=lcc
+lat_1=37.93333333333333+lat_2=36.73333333333333+lat_0=36.33333333333334+lon_0=-85.75+x_0=500000+y_0=500000+ellps=GRS80+datum=NAD83+units=m+no_defs';
```

使用法

(先頭に戻る) (247 ページ)

地理空間参照系では、LINEAR と ANGULAR の両方の測定単位を指定できます。非地理空間参照系では、LINEAR 測定単位のみを指定できます。LINEAR 測定単位は、点と領域間の距離を計算するために使用されます。ANGULAR 測定単位は、角度の緯度と経度を解釈する方法を示し、投影座標系の場合は NULL、地理的座標系の場合は NULL 以外になります。

操作から返される派生ジオメトリはすべて正規化されます。

SQL Anywhere 以外のデータベースで同期されるデータを操作する場合は、データの元の特性を保持できるようにするため、STORAGE FORMAT を 'Original' または 'Mixed' のいずれかに設定してください。

標準

(先頭に戻る) (247 ページ)

ANSI SQL – 準拠レベル：Transact-SQL® 拡張。

パーミッション

(先頭に戻る) (247 ページ)

次のいずれかが必要です。

- **MANAGE ANY SPATIAL OBJECT** システム権限
- **CREATE ANY OBJECT** システム権限

CREATE SPATIAL UNIT OF MEASURE 文

空間測定単位を作成するか、置き換えます。

クイックリンク：

「パラメータ」 (255 ページ)

「例」 (256 ページ)

「使用法」 (256 ページ)

「標準」 (256 ページ)

「パーミッション」 (256 ページ)

構文

```
CREATE [ OR REPLACE ] SPATIAL UNIT OF MEASURE identifier
    TYPE { LINEAR | ANGULAR }
    [ CONVERT USING number ]
```

パラメータ

(先頭に戻る) (255 ページ)

- **OR REPLACE** – OR REPLACE の指定は、空間測定単位を作成するか、既存の空間測定単位を同じ名前で置き換えます。この句は、現在の権限を保持します。使用中の空間単位を置き換えようとすると、エラーが返されます。
- **TYPE** – 角度 (ANGULAR) または距離 (LINEAR) のどちらに測定単位を使用するかを定義します。
- **CONVERT USING** – ベース単位に関する空間単位の換算係数。線形単位の場合、ベース単位は METRE です。角度単位の場合、ベース単位は RADIAN です。

例

(先頭に戻る) (255 ページ)

- **例 1** – Test という空間測定単位を作成します。

```
CREATE SPATIAL UNIT OF MEASURE Test
TYPE LINEAR
CONVERT USING 15;
```

使用法

(先頭に戻る) (255 ページ)

CONVERT USING 句は、定義された測定単位の値をベース測定単位 (ラジアンまたはメートル) に換算する方法を定義するために使用します。指定された換算係数を測定値に乗算して、ベース測定単位の値を取得します。たとえば、測定値 512 ミリメートルに換算係数 0.001 を乗算して、測定値 0.512 メートルを取得します。

距離 (ST_Distance または ST_Length) または領域を計算する場合、空間参照系では常に線形測定単位が使用されます。たとえば、空間参照系の線形測定単位がマイルの場合、使用する領域単位は平方マイルになります。場合によっては、線形測定単位を使用するよう指定するオプションパラメータが空間メソッドで受け入れられることがあります。たとえば、空間参照系の線形測定単位がマイルの場合、オプションパラメータ 'metre' を使用すると、2つのジオメトリ間の距離をメートルで取得できます。

投影座標系の場合は、空間参照系の線形単位で X 座標と Y 座標が指定されます。地理的座標系の場合は、空間参照系に関連する角度測定単位で緯度と経度が指定されます。多くの場合、この角度測定単位は度ですが、有効な任意の角度測定単位を使用できます。

sa_install_feature システムプロシージャを使用して、事前に定義された測定単位をデータベースに追加できます。

標準

(先頭に戻る) (255 ページ)

ANSI SQL – 準拠レベル： Transact-SQL 拡張。

パーミッション

(先頭に戻る) (255 ページ)

次のいずれかが必要です。

- MANAGE ANY SPATIAL OBJECT システム権限

- CREATE ANY OBJECT システム権限

CREATE TABLE 文

データベースまたはリモートサーバに新しいテーブルを作成します。

クイックリンク：

「パラメータ」 (259 ページ)

「例」 (269 ページ)

「使用法」 (273 ページ)

「標準」 (274 ページ)

「パーミッション」 (275 ページ)

構文

```
CREATE [ { GLOBAL | LOCAL } TEMPORARY ] TABLE
  [ IF NOT EXISTS ] [ owner. ] table-name
  ... ( column-definition [ column-constraint ] ...
  [ , column-definition [ column-constraint ] ... ]
  [ , table-constraint ] ... )
  | { ENABLE | DISABLE } RLV STORE

...[ IN dbspace-name ]
...[ ON COMMIT { DELETE | PRESERVE } ROWS ]
[ AT location-string ]
[ PARTITION BY
  range-partitioning-scheme
  | hash-partitioning-scheme
  | composite-partitioning-scheme ]

column-definition - (構文に戻る)
column-name data-type
[ [ NOT ] NULL ]
[ DEFAULT default-value | IDENTITY ]
[ PARTITION | SUBPARTITION ( partition-name IN dbspace-name
[ , ... ] ) ]

default-value - (back to column-definition)
special-value
| string
| global variable
| [ - ] number
| ( constant-expression )
| built-in-function( constant-expression )
| AUTOINCREMENT
| CURRENT DATABASE
| CURRENT REMOTE USER
| NULL
```

```

| TIMESTAMP
| LAST USER

special-value - (back to default value)
CURRENT
{ DATE
| TIME
| TIMESTAMP
| USER
| PUBLISHER }
| USER

column-constraint - (構文に戻る)
[ CONSTRAINT constraint-name ] {
  { UNIQUE
    | PRIMARY KEY
    | REFERENCES table-name [ ( column-name ) ] [ action ]
  }
  [ IN dbspace-name ]
  | CHECK ( condition )
  | IQ UNIQUE ( integer )
}

table-constraint - (構文に戻る)
[ CONSTRAINT constraint-name ]
{ { UNIQUE ( column-name [ , column-name ] ... )
  | PRIMARY KEY ( column-name [ , column-name ] ... )
}
[ IN dbspace-name ]
| foreign-key-constraint
| CHECK ( condition )
| IQ UNIQUE ( integer )
}

foreign-key-constraint - (back to table-constraint)
FOREIGN KEY [ role-name ] [ ( column-name [ , column-name ] ... ) ]
...REFERENCES table-name [ ( column-name [ , column-name ] ... ) ]
...[ actions ] [ IN dbspace-name ]

actions - (back to foreign-key-constraint)
[ ON { UPDATE | DELETE } RESTRICT ]

location-string - (構文に戻る) or (back to composite-partitioning-scheme)
{ remote-server-name. [ db-name ].[ owner ].object-name
| remote-server-name; [ db-name ]; [ owner ];object-name }

range-partitioning-scheme - (構文に戻る)
RANGE ( partition-key ) ( range-partition-decl [ , range-partition-decl ... ] )

partition-key - (back to range-partitioning-scheme) or (back to hash-
partitioning-scheme)
column-name

range-partition-decl - (back to range-partitioning-scheme)
VALUES <= ( { constant-expr
```

```

| MAX } [ , { constant-expr
| MAX } ]... )
[ IN dbspace-name ]

```

hash-partitioning-scheme - (構文に戻る) or (back to composite-partitioning-scheme)

```

HASH ( partition-key [ , partition-key, ... ] )

```

composite-partitioning-scheme - (構文に戻る)

```

hash-partitioning-scheme SUBPARTITION range-partitioning-scheme

```

パラメータ

(先頭に戻る) (257 ページ)

- **IN** - column-definition 句、column-constraint 句、table-constraint 句、foreign-key 句、および partition-decl 句で、オブジェクトが作成される DB 領域を指定するために使用します。IN 句を省略した場合、SAP Sybase IQ はテーブルが割り当てられている DB 領域にオブジェクトを作成します。

この句で SYSTEM を指定し、永久テーブルまたはテンポラリテーブルをカタログストアに置くことができます。IQ_SYSTEM_TEMP を指定すると、テンポラリなユーザオブジェクト (テーブル、パーティション、またはテーブルインデックス) を IQ_SYSTEM_TEMP に格納できます。または、

TEMP_DATA_IN_SHARED_TEMP オプションが 'ON' に設定されており、IQ_SHARED_TEMP DB 領域に RW ファイルが含まれている場合は、IQ_SHARED_TEMP に格納できます (IN 句を IQ_SHARED_TEMP とともに指定することはできません)。これ以外の IN 句の使用はすべて無視されます。デフォルトでは、すべての永久テーブルはメイン IQ ストアに、すべてのテンポラリテーブルはテンポラリ IQ ストアに配置されます。グローバルテンポラリテーブルとローカルテンポラリテーブルを IQ ストアに置くことは絶対にできません。

以下の構文はサポートされていません。

```

CREATE LOCAL TEMPORARY TABLE tab1(c1 int) IN IQ_SHARED_TEMP

```

BIT データ型のカラムは DB 領域に明示的に配置することはできません。以下は BIT データ型に対してサポートされていません。

```

CREATE TABLE t1(c1_bit bit IN iq_main);

```

- **ON COMMIT** - テンポラリテーブルに対してのみ使用できます。デフォルトで、テンポラリテーブルのローは COMMIT のときに削除されます。
- **AT** - location-string 句で指定されたりリモートロケーションにマップするプロキシテーブルを作成します。プロキシテーブル名は最大で 30 文字までです。AT 句は、デリミタとしてセミコロン (;) をサポートします。セミコロンが location-

string 句のどこかにある場合、そのセミコロンはフィールドデリミタです。セミコロンがない場合は、ピリオドがフィールドデリミタです。これにより、データベースフィールドと所有者フィールドでファイル名と拡張子を使用できます。

セミコロンのフィールドデリミタは、現在サポートされていないサーバクラスで主に使用されていますが、ピリオドもフィールドデリミタとして機能する状況ではセミコロンも使用できます。たとえば、次の文は、テーブル proxy_a をリモートサーバ myasa の SQL Anywhere データベース mydb にマッピングします。

```
CREATE TABLE proxy_a1
AT 'myasa;mydb;;a1'
```

外部キー定義は、リモートテーブルでは無視されます。リモートテーブルを参照するローカルテーブルの外部キー定義も無視されます。プライマリキー定義は、サーバがプライマリキーをサポートする場合、リモートサーバに送信されます。

シンプレックス環境では、同じノード上でリモートテーブルを参照するプロキシテーブルを作成することはできません。マルチプレックス環境では、マルチプレックス内で定義されたリモートテーブルを参照するプロキシテーブルを作成することはできません。

- **IF NOT EXISTS** – 指定したオブジェクトがすでに存在する場合、変更は行われず、エラーは返されません。
- **{ ENABLE | DISABLE } RLV STORE** – このテーブルをインメモリリアルタイム更新の対象として RLV ストアに登録します。IQ テンポラリテーブルはサポートされていません。この値は、データベースオプション **BASE_TABLES_IN_RLV** の値よりも優先されます。この値を ENABLE に設定するには、CREATE TABLE システム権限と、RLV ストアの DB 領域に対する CREATE パーミッションが必要です。
- **column-definition** – テーブルカラムを定義します。使用可能なデータ型については、『リファレンス：ビルディングブロック、テーブル、およびプロシージャ』の「SQL データ型」を参照してください。同じテーブル内の 2 つのカラムが同じ名前を持つことはできません。最大 45,000 のカラムを作成可能ですが、1 つのテーブルに 10,000 を超えるカラムを作成すると、パフォーマンスの低下を招くおそれがあります。
- **[NOT] NULL** – NULL 値を含めるか、除外するかを設定します。NOT NULL を指定した場合や、カラムが UNIQUE 制約または PRIMARY KEY 制約を受ける場合は、カラムに NULL 値を含めることができません。NULL を許可するカラム数のテーブルごとの制限は、約 $8 * (\text{database-page-size} - 30)$ です。

- **DEFAULT default-value** – CREATE TABLE (および ALTER TABLE) 文の DEFAULT キーワードでカラムのデフォルト値を指定します。DEFAULT 値は、カラムの値を指定しない INSERT (または LOAD) 文でカラムの値として使用されます。
- **DEFAULT AUTOINCREMENT** – DEFAULT AUTOINCREMENT カラムの値は、テーブル内の各ローをユニークに識別します。この種のカラムは、Adaptive Server との互換性を考慮して IDENTITY カラムとも呼ばれます。IDENTITY/DEFAULT AUTOINCREMENT カラムには、挿入や更新の際に自動的に生成される連続した数値が格納されます。IDENTITY/DEFAULT AUTOINCREMENT を使用する場合、カラムは整数データ型のいずれか、または真数型で、位取りを 0 にする必要があります。カラムの値は NULL でもかまいません。テーブル名は、所有者の名前で修飾して指定する必要があります。

ON でテーブルへの挿入を行います。IDENTITY/DEFAULT

AUTOINCREMENT カラムへの値が指定されなければ、カラム内のどの値よりも大きいユニークな値が生成されます。カラムへ格納する値を INSERT に指定すれば、その値が使用されます。指定した値がそのカラムの現在の最大値よりも小さい場合、指定した値は後続の挿入に対して生成する値の起点として使用されます。

ローを削除しても IDENTITY/AUTOINCREMENT カウンタはデクリメントされません。ローの削除によって作成されたギャップは、挿入を行うときに明示的に割り当てることによってのみ埋めることができます。

IDENTITY/AUTOINCREMENT カラムへの挿入を実行するには、データベースオプション IDENTITY_INSERT をテーブル名に設定する必要があります。

以下に、IDENTITY カラムを持つテーブルを作成し、データを明示的に追加する例を示します。

```
CREATE TABLE mytable(c1 INT IDENTITY);
SET TEMPORARY OPTION IDENTITY_INSERT = "DBA".mytable;
INSERT INTO mytable VALUES(5);
```

最大値よりも小さいロー番号を明示的に挿入すると、後続のローで明示的に割り当てなくても、その最大値より 1 大きい値に自動的にインクリメントされます。

カラムに直前に挿入された値は、グローバル変数 @@identity を調べることによって確認できます。

- **IDENTITY – AUTOINCREMENT** デフォルトを使用する代わりに Transact-SQL® 互換の代替手段です。SAP Sybase IQ では、IDENTITY 句または DEFAULT AUTOINCREMENT 句のどちらかを使用して IDENTITY カラムを作成できます。

- **table-constraint** – データベース内のデータの整合性を保証します。整合性制約には次の 4 つのタイプがあります。

- **UNIQUE** – 1 つまたは複数のカラムによってテーブル内の各ローがユニークに識別されるように指定します。テーブル内の 2 つのローは、指定されたすべてのカラム中に同じ値を持つことはできません。1 つのテーブルに複数の一意性制約が存在することがあります。
- **PRIMARY KEY** – UNIQUE 制約と同じですが、プライマリキー制約はテーブルに 1 つしか作成できない点が違います。プライマリキー制約と一意性制約を同じカラムに指定することはできません。プライマリキーは通常、特定のローにとって最適な識別子を識別します。たとえば、顧客番号は顧客テーブルのプライマリキーです。
- **FOREIGN KEY** – 一方のカラムセットに対する値を制限し、他方のテーブルのプライマリキーまたは一意性制約の値と一致させます。たとえば、外部キー制約を使用して、請求書テーブルの顧客番号が顧客テーブルの顧客番号と確実に一致するようにできます。

ローカルテンポラリテーブルに対しては外部キー制約を作成できません。グローバルテンポラリテーブルは ON COMMIT PRESERVE ROWS で作成してください。

- **CHECK** – 任意の条件を検証できます。たとえば、検査制約を使用して Gender カラムに Male と Female の値しか含まれないようにすることができます。テーブル内のどのローも、制約に違反することは許されません。**INSERT** 文または **UPDATE** 文によってローが制約に違反する場合、操作は許可されず、この文の結果は取り消されます。

カラムの検査制約に記述され、先頭に '@' の記号が付くカラム識別子は、実際のカラム名のプレースホルダです。したがって、次のように記述された文は、

```
CREATE TABLE t1(c1 INTEGER CHECK (@foo < 5))
```

次の文とまったく同じになります。

```
CREATE TABLE t1(c1 INTEGER CHECK (c1 < 5))
```

テーブルの検査制約に記述され、先頭に '@' の記号が付くカラム識別子はプレースホルダではありません。

ある文によって整合性制約に違反するデータベースへの変更が生じた場合、その文は事実上実行されず、エラーがレポートされます「事実上」とは、エラーが検出されるより前にこの文が行った変更がすべて取り消されることを示します。

SAP Sybase IQ はそのカラムの HG インデックスを作成することにより、シングルカラム UNIQUE 制約を強制します。

注意： BIT データ型のカラムに、UNIQUE 制約または PRIMARY KEY 制約を定義することはできません。また、BIT データ型のカラムのデフォルトでは NULL 値を使用できませんが、カラムを明示的に定義して、NULL 値を使用できるように変更できます。

- **column-constraint** – カラムに格納可能な値を制限します。カラム制約とテーブル制約によってデータベース内のデータの整合性が保証されます。整合性制約の違反を起こす文は、実行が完了しません。このような文がエラー検出の前に行った変更は取り消され、エラーがレポートされます。カラム制約は、対応するテーブル制約の省略形です。たとえば、次の文は同じです。

```
CREATE TABLE Products (
    product_num integer UNIQUE
)
CREATE TABLE Products (
    product_num integer,
    UNIQUE ( product_num )
)
```

通常、カラム制約を使用するのは、制約がテーブル内で複数のカラムを参照しない場合です。複数のカラムを参照する場合は、テーブル制約を使用する必要があります。

- **IQ UNIQUE** – カラムの予期されるカーディナリティを定義し、カラムをフラット FP または NBit FP のどちらとしてロードするかを決定します。IQ UNIQUE(n) の値を明示的に 0 に設定すると、カラムはフラット FP としてロードされます。IQ UNIQUE 制約のないカラムは、FP_NBIT_AUTOSIZE_LIMIT、FP_NBIT_LOOKUP_MB、FP_NBIT_ROLLOVER_MAX_MB の各オプションで定義された上限まで暗黙的に NBit としてロードされます。
 - FP_NBIT_AUTOSIZE_LIMIT は、NBit としてロードする、重複しない値の数を制限します。
 - FP_NBIT_LOOKUP_MB は、NBit ディクショナリの合計サイズのスレッシュホールドを設定します。
 - FP_NBIT_ROLLOVER_MAX_MB は、NBit からフラット FP への暗黙的な NBit ロールオーバーで使用するディクショナリサイズを設定します。
 - FP_NBIT_ENFORCE_LIMITS は、NBit ディクショナリのサイズ制限を強制します。このオプションはデフォルトで OFF になっています。

IQ UNIQUE を FP_NBIT_AUTOSIZE_LIMIT 未満の n 値とともに使用する必要はありません。自動サイズ機能によって、カーディナリティが低いか中程度のカラムはすべて NBit としてサイズ決定されます。カラムをフラット FP としてロードする場合や、重複しない値の数が

FP_NBIT_AUTOSIZE_LIMIT を超えるときにカラムを NBit としてロードする場合は、IQ UNIQUE を使用します。

注意：

- 高い IQ UNIQUE 値を指定する際はメモリ使用率を考慮します。マシンリソースに制限がある場合、FP_NBIT_ENFORCE_LIMITS='OFF' (デフォルト) でロードしないでください。
SAP Sybase IQ 16.0 以前は、16777216 を超える IQ UNIQUE *n* 値はフラット FP にロールオーバーされていました。16.0 では、より大きい IQ UNIQUE 値がトークン化でサポートされていますが、カーディナリティとカラム幅によっては、大量のメモリリソースが必要になる場合があります。
- BIT、BLOB、および CLOB の各データ型は NBit ディクショナリ圧縮をサポートしません。FP_NBIT_IQ15_COMPATIBILITY='OFF' である場合、これらのデータ型を含む CREATE TABLE 文または ALTER TABLE 文でゼロ以外の IQ UNIQUE カラムを指定すると、エラーが返されます。

-
- **column-constraint 句と table-constraint 句** – カラム制約とテーブル制約によってデータベース内のデータの整合性が保証されます。
 - **PRIMARY KEY または PRIMARY KEY (column-name, ...)** – テーブルのプライマリキーは、リストしたカラムで構成されます。プライマリキーに指定されたどのカラムにも NULL 値を格納することはできません。SAP Sybase IQ では、テーブル内の各ローが必ず、ユニークなプライマリキー値を持ちます。1 つのテーブルが持てる PRIMARY KEY は 1 つだけです。

2 番目の形式 (PRIMARY KEY の後にカラムリストが続く形式) を使用する場合、カラムのリスト順ではなく定義順にカラムを含めるプライマリキーが作成されます。

カラムを PRIMARY KEY、FOREIGN KEY、または UNIQUE として指定すると、SAP Sybase IQ によってそのカラムに自動的に High_Group インデックスが作成されます。マルチカラムプライマリキーの場合、このインデックスは個々のカラムではなく、プライマリキーに対して作成されます。パフォーマンスを高めるため、各カラムに HG または LF インデックスを個別に作成してください。

- **REFERENCES primary-table-name [(primary-column-name)]** – カラムを、プライマリテーブルのプライマリキーまたは一意性制約に対する外部キーとして定義します。通常、外部キーは一意性制約のためのものではなく、プライマリキーのためのものです。プライマリカラム名を指定する場合、この名前は一意性制約またはプライマリキーの制約を受けるプライマリテーブルのカラム名と一致する必要があります。また、この制約は、その 1 カラムだけで構成される必要があります。それ以外の場合、外部キーは第 2 のテーブルのプライマリキーを参照します。プライマリキーと外部キーは、

データ型と精度、位取り、符号が同じであることが必要です。シングルカラムの外部キーには、ユニークでないシングルカラムの HG インデックスだけが作成されます。マルチカラム外部キーに対しては、SAP Sybase IQ はユニークでない複合 HG インデックスを作成します。ユニークまたはユニークでない HG インデックスのマルチカラム複合キーの最大幅は 1KB です。

テンポラリテーブルは、ベーステーブルを参照する外部キーを持つことはできません。また、ベーステーブルも、テンポラリテーブルを参照する外部キーを持つことはできません。ローカルテンポラリテーブルは、外部キーを持つことも、外部キーによって参照されることもできません。

- **FOREIGN KEY [role-name] [(...)] REFERENCES primary-table-name [(...)]**—別のテーブルのプライマリキーまたは一意性制約を参照する外部キーを定義します。通常、外部キーは一意性制約のためのものではなく、プライマリキーのためのものです (この説明では、この他方のテーブルをプライマリテーブルと呼びます)。

プライマリテーブルのカラム名が指定されていない場合、プライマリテーブルのカラムは、テーブルのプライマリキーの中のカラムになります。外部キーカラム名が指定されていない場合、外部キーカラムは、プライマリテーブルの中のカラムと同じ名前になります。外部キーカラム名が指定されている場合は、プライマリキーのカラム名を指定する必要があります。これらのカラム名は、リスト内の位置に応じて一対一になります。

プライマリテーブルと外部キーテーブルが同じでない場合、参照されるキーには一意性制約またはプライマリキー制約を定義する必要があります。参照されるキーと外部キーは、カラムの数、データ型、符号、精度、位取りが同じでなくてはなりません。

ローの外部キーの値は、外部キー内の 1 つまたは複数のカラムで、null を許可する外部キーカラムに null が格納されている場合を除き、プライマリテーブルのいずれかのローに格納された候補キー値として出現する必要があります。

明示的に定義されない外部キーのカラムは、プライマリテーブルの対応するカラムと同じデータ型で自動的に作成されます。これらの自動的に作成されたカラムは外部テーブルのプライマリキーの一部にはなりません。したがって、プライマリキーと外部キーの両方の中で使われるカラムは、明示的に作成する必要があります。

role-name は外部キーの名前です。*role-name* の主な機能は、同じテーブルに対する 2 つの外部キーを区別することです。*role-name* が指定されていない場合、*role-name* は次のように割り当てられます。

1. テーブル名と同じ *role-name* を含む外部キーが存在しない場合、テーブル名が *role-name* として割り当てられます。

2. テーブル名がすでに使用されている場合、*role-name* は、0 が埋め込まれた、テーブルに固有の 3桁の数字と結合されたテーブル名になります。

参照整合性アクションは、データベース内で外部キー関係を維持するために取られるアクションを定義します。データベーステーブルからプライマリキー値が変更されたり削除されると、それに対応して何らかの修正が必要となる外部キー値が他のテーブルに存在する可能性があります。ON DELETE 句の後に続けて RESTRICT 句を指定できます。

- **RESTRICT** – データベースのどこかに対応する外部キーがあるにもかかわらず、プライマリキー値を更新または削除しようとする、エラーになります。外部キーを更新して、候補キーに一致しない新しい値を作成しようとする、エラーになります。この動作はデフォルトです。ただし、オプションで参照整合性に違反するローを拒否するように LOAD を指定した場合を除きます。これにより、文レベルでの参照整合性が確保されます。

アクションを何も指定しないで CHECK ON COMMIT を使用すると、RESTRICT は DELETE のアクションに対して適用されます。SAP Sybase IQ は CHECK ON COMMIT をサポートしません。

グローバルテナポラリテーブルは、ベーステーブルを参照する外部キーを持つことはできません。また、ベーステーブルも、グローバルテナポラリテーブルを参照する外部キーを持つことはできません。ローカルテナポラリテーブルは、外部キーを持つことも、外部キーによって参照されることもできません。

- **CHECK (条件)** – 条件に合わないローは許可されません。INSERT 文によってローがこの条件を満たさなくなる場合、操作は許可されず、この文の効果は取り消されます。

変更は、条件が FALSE の場合にのみ拒否されます。条件が UNKNOWN の場合、変更は許可されます。CHECK 条件は SAP Sybase IQ では強制されません。

注意： 孤立した外部キーがないと確実にわかっている場合を除き、できる限り、SAP Sybase IQ に参照整合性(すなわち、外部キーとプライマリキーの関係)を定義しないでください。

- **リモートテーブル** – 外部キー定義は、リモートテーブルでは無視されます。リモートテーブルを参照するローカルテーブルの外部キー定義も無視されます。プライマリキー定義は、サーバがプライマリキーをサポートする場合、リモートサーバに送信されます。
- **PARTITION BY** – 大きなテーブルを、より小さく管理しやすいストレージオブジェクトに分割します。各パーティションは親テーブルと同じ論理属性を共有しますが、別々の DB 領域に配置して個別に管理できます。SAP Sybase IQ は、次のような複数のテーブル分割スキームをサポートしています。

- ハッシュパーティション
- 範囲パーティション
- 複合パーティション

partition-key は、テーブル分割キーが格納されている 1 つまたは複数のカラムです。分割キーには、NULL 値 および DEFAULT 値を含めることができますが、次のカラムを含めることはできません。

- LOB (BLOB または CLOB) カラム
- BINARY または VARBINARY カラム
- 長さが 255 バイトを超える CHAR または VARCHAR カラム
- BIT カラム
- FLOAT/DOUBLE/REAL カラム
- **PARTITION BY RANGE** – 分割カラム内の値の範囲によってローを分割します。範囲分割は、単一の分割キーカラムおよび最大 1024 パーティションまでに制限されています。range-partitioning-scheme 内の partition-key は、テーブル分割キーが格納されているカラムです。

```
range-partition-decl:
  partition-name VALUES <= ( {constant-expr | MAX } [ ,
  { constant-expr | MAX } ]... )
  [ IN dbspace-name ]
```

partition-name は、テーブルローが格納される新しいパーティションの名前です。パーティション名は、テーブル上にあるパーティションセット内でユニークである必要があります。パーティション名は必須です。

- **VALUE** – 各パーティションの包括的な上限を (昇順に) 指定します。ユーザは、各ローが 1 つのパーティションのみに分配されるように、各範囲分割の分割基準を指定する必要があります。NULL は分割カラムに使用でき、NULL を分割キー値に含んだローは最初のテーブル分割に属します。ただし、NULL をバインド値に指定することはできません。

最初のパーティションには、下限 (MIN 値) は設定されていません。分割キーの最初のカラムにある NULL セルのローは、最初のパーティションに移動します。最後のパーティションでは、包括的な上限または MAX を指定できます。最後のパーティションの上限値が MAX でない場合は、最後のパーティションの上限値よりも大きい分割キーの値を含んだローをロードまたは挿入すると、エラーが生成されます。

- **MAX** – 無制限の上限を示し、最後のパーティションに対してのみ指定できます。
- **IN** – partition-decl でパーティションのローが存在する DB 領域を指定します。

次の制限を設定すると、範囲分割されたテーブルの分割キーとバインド値がその影響を受けます。

- パーティションバインドは定数式でなく、定数として指定する必要があります。
- パーティションバインドは、パーティションの作成順に応じて、昇順で指定する必要があります。つまり、2 番目のパーティションの上限は最初のパーティションよりも高く指定する必要があります、というようになります。さらに、パーティションバインドの値は、対応する分割キーカラムのデータ型と互換性がなければなりません。たとえば、VARCHAR は CHAR と互換性があります。
- バインド値に対応する分割キーのカラムとは異なるデータ型が指定されていると、SAP Sybase IQ はバインド値を分割キーのカラムのデータ型に変換します。ただし、次の場合は例外となります。
- 明示的な変換は使用できません。この例では、INT から VARCHAR に明示的に変換しようとしてエラーが生成されます。

```
CREATE TABLE Employees(emp_name VARCHAR(20))
PARTITION BY RANGE (emp_name)
(p1 VALUES <= (CAST (1 AS VARCHAR(20))),
p2 VALUES <= (CAST (10 AS VARCHAR(20)))
```

- データロスにつながる暗黙的な変換は使用できません。この例では、パーティションバインドは分割キー型と互換性がありません。丸めを前提で処理を行うとデータロスにつながる可能性があり、エラーが生成されます。
- この例では、パーティションバインドと分割キーのデータ型の間には互換性があります。バインド値は FLOAT 値に直接変換されます。丸め処理は必要なく、変換はサポートされています。

```
CREATE TABLE emp_id (id INT) PARTITION BY RANGE (id) (p1 VALUES
<= (10.5), p2 VALUES <= (100.5))
```

- 非バイナリデータ型からバイナリデータ型に変換することはできません。たとえば、次の変換は実行できずに、エラーが返されます。

```
CREATE TABLE newemp (name BINARY)
PARTITION BY RANGE (name)
(p1 VALUES <= ("Maarten"),
p2 VALUES <= ("Zymerman"))
```

- NULL を範囲分割テーブルで境界として使用することはできません。
- 分割キーの最初のカラムのセル値が NULL と評価された場合、ローは最初のパーティションに挿入されます。SAP Sybase IQ は、1つのカラムの分割キーのみをサポートしているため、分割キー内に NULL が含まれていると、ローは最初のパーティションに分配されます。
- **PARTITION BY HASH** – 内部ハッシュ関数によって処理された分割キーの値に基づいて、データをパーティションにマップします。ハッシュ分割キーは最大 8 カラムで、組み合わせた宣言カラム幅が 5300 バイト以下に制限されていま

す。ハッシュパーティションの場合、テーブル作成者は分割キーカラムのみを決定します。パーティションの数と位置は内部的に決定されます。

hash-partitioning 宣言内の partition-key は、1つのカラムまたはカラムのグループです。その複合値によってデータの各ローが格納されるパーティションが決まります。

```
hash-partitioning-scheme:
HASH ( partition-key [ , partition-key, ... ] )
```

• 制限事項 –

- ハッシュ分割できるのはベーステーブルのみです。グローバルテンポラリテーブルやローカルテンポラリテーブルを分割しようとすると、エラーが発生します。
 - ハッシュパーティションの追加、削除、マージ、分割はできません。
 - カラムをハッシュ分割キーから追加または削除することはできません。
- **PARTITION BY HASH RANGE** – ハッシュ分割されたテーブルを範囲によってさらに分割します。hash-range-partitioning-scheme 宣言内の SUBPARTITION BY RANGE 句は、新しい範囲サブパーティションを既存のハッシュ範囲分割テーブルに追加します。

```
hash-range-partitioning-scheme:
PARTITION BY HASH ( partition-key [ , partition-key, ... ] )
  [ SUBPARTITION BY RANGE ( range-partition-decl [ , range-
partition-decl ... ] ) ]
```

ハッシュパーティションはデータの論理的な配分および配置方法を指定するのに対して、範囲サブパーティションはデータの物理的な配置方法を指定します。新しい範囲サブパーティションは、既存のハッシュ範囲分割テーブルと同じハッシュ分割キーを持つハッシュによって論理的に分割されます。範囲サブパーティションは1つのカラムに制限されています。

• 制限事項 –

- ハッシュ分割できるのはベーステーブルのみです。グローバルテンポラリテーブルやローカルテンポラリテーブルを分割しようとすると、エラーが発生します。
- ハッシュパーティションの追加、削除、マージ、分割はできません。
- カラムをハッシュ分割キーから追加または削除することはできません。

注意： 範囲パーティションと複合分割スキームは、ハッシュ範囲パーティションと同様、個別にライセンスが必要な VLDB Management オプションを必要とします。

例

(先頭に戻る) (257 ページ)

- **例 1**–5つのカラムを持つ SalesOrders2 という名前のテーブルを作成します。カラム FinancialCode、OrderDate、ID のデータページは DB 領域 Dsp3 にあります。整数カラム CustomerID のデータページは DB 領域 Dsp1 にあります。CLOB カラム History のデータページは DB 領域 Dsp2 にあります。プライマリキー (ID の HG) のデータページは DB 領域 Dsp4 にあります。

```
CREATE TABLE SalesOrders2 (
  FinancialCode CHAR(2),
  CustomerID int IN Dsp1,
  History CLOB IN Dsp2,
  OrderDate TIMESTAMP,
  ID BIGINT,
  PRIMARY KEY(ID) IN Dsp4
) IN Dsp3
```

- **例 2**–4つのカラムを持つ fin_code2 というテーブルを作成します。カラム code、type、id のデータページはデフォルトの DB 領域にあり、データベースオプション DEFAULT_DBSPACE の値によって決まります。CLOB カラム description のデータページは DB 領域 Dsp2 にあります。外部キー fk1 (c1 の HG) のデータページは DB 領域 Dsp4 にあります。

```
CREATE TABLE fin_code2 (
  code INT,
  type CHAR(10),
  description CLOB IN Dsp2,
  id BIGINT,
  FOREIGN KEY fk1(id) REFERENCES SalesOrders(ID) IN Dsp4
)
```

- **例 3**–テーブル t1 を作成します。ここで、パーティション p1 はパーティション p2 に、パーティション p2 はパーティション p3 に隣接します。

```
CREATE TABLE t1 (c1 INT, c2 INT)
PARTITION BY RANGE(c1)
(p1 VALUES <= (0), p2 VALUES <= (10), p3 VALUES <= (100))
```

- **例 4**–6つのカラムと3つのパーティションを持つ、RANGE 分割されたテーブル bar を作成し、データを日付に基づいたパーティションにマッピングします。

```
CREATE TABLE bar (
  c1 INT IQ UNIQUE(65500),
  c2 VARCHAR(20),
  c3 CLOB PARTITION (P1 IN Dsp11, P2 IN Dsp12,
    P3 IN Dsp13),
  c4 DATE,
  c5 BIGINT,
  c6 VARCHAR(500) PARTITION (P1 IN Dsp21,
    P2 IN Dsp22),
  PRIMARY KEY (c5) IN Dsp2) IN Dsp1
PARTITION BY RANGE (c4)
(P1 VALUES <= ('2006/03/31') IN Dsp31,
```

```
P2 VALUES <= ('2006/06/30') IN Dsp32,
P3 VALUES <= ('2006/09/30') IN Dsp33
) ;
```

各パーティションのデータページ割り付け：

パーティション	DB 領域	カラム
P1	Dsp31	c1、c2、c4、c5
P1	Dsp11	c3
P1	Dsp21	c6
P2	Dsp32	c1、c2、c4、c5
P2	Dsp12	c3
P2	Dsp22	c6
P3	Dsp33	c1、c2、c4、c5、c6
P3	Dsp13	c3
P1、P2、P3	Dsp1	c1 および他の共有データのルックアップストア
P1、P2、P3	Dsp2	プライマリキー (c5 の HG)

- **例 5** – HASH 分割された (table tbl142) を作成します。このテーブルは、PRIMARY KEY (カラム c1) と HASH PARTITION KEY (カラム c4 および c3) を含みます。

```
CREATE TABLE tbl142 (
  c1 BIGINT NOT NULL,
  c2 CHAR(2) IQ UNIQUE(50),
  c3 DATE IQ UNIQUE(36524),
  c4 VARCHAR(200),
  PRIMARY KEY (c1)
)
PARTITION BY HASH ( c4, c3 )
```

- **例 6** – PRIMARY KEY (カラム c1)、ハッシュ分割キー (カラム c4 および c2)、および範囲サブ分割キー (カラム c3) を指定して、ハッシュ範囲分割されたテーブルを作成します。

```
CREATE TABLE tbl142 (
  c1 BIGINT NOT NULL,
  c2 CHAR(2) IQ UNIQUE(50),
  c3 DATE,
  c4 VARCHAR(200),
  PRIMARY KEY (c1)) IN Dsp1
```

```
PARTITION BY HASH ( c4, c2 )
SUBPARTITION BY RANGE ( c3 )
( P1 VALUES <= (2011/03/31) IN Dsp31,
  P2 VALUES <= (2011/06/30) IN Dsp32,
  P3 VALUES <= (2011/09/30) IN Dsp33) ;
```

- **例 7** – 図書データベース用にテーブルを作成し、貸出された図書の情報を保持します。

```
CREATE TABLE borrowed_book (
  date_borrowed DATE NOT NULL,
  date_returned DATE,
  book          CHAR(20)
                REFERENCES library_books (isbn),
  CHECK( date_returned >= date_borrowed )
)
```

- **例 8** – リモートサーバ SERVER_A にテーブル t1 を作成し、リモートテーブルにマッピングされるプロキシテーブル t1 を作成します。

```
CREATE TABLE t1
( a INT,
  b CHAR(10))
AT 'SERVER_A.db1.joe.t1'
```

- **例 9** – カラム c1 に特殊定数 LAST USER のデフォルト値を含むテーブル tab1 を作成します。

```
CREATE TABLE tab1(c1 CHAR(20) DEFAULT LAST USER)
```

- **例 10** – カラム c1 を持つローカルテンポラリテーブル tab1 を作成します。

```
CREATE LOCAL TEMPORARY TABLE tab1(c1 int) IN IQ_SYSTEM_TEMP
```

この例では、次の場合、tab1 が IQ_SYSTEM_TEMP DB 領域に作成されます。

- DQP_ENABLED 論理サーバポリシーオプションが ON に設定されているが、読み取り／書き込み可能なファイルが IQ_SHARED_TEMP に存在しない場合
- DQP_ENABLED オプションが OFF、TEMP_DATA_IN_SHARED_TEMP 論理サーバポリシーオプションが ON に設定されているが、読み取り／書き込み可能なファイルが IQ_SHARED_TEMP に存在しない場合
- DQP_ENABLED オプションと TEMP_DATA_IN_SHARED_TEMP オプションが両方とも OFF に設定されている場合

この例では、次の場合、同じテーブル tab1 が IQ_SHARED_TEMP DB 領域に作成されます。

- DQP_ENABLED が ON で、読み取り／書き込み可能なファイルが IQ_SHARED_TEMP に存在する場合

- DQP_ENABLED が OFF、TEMP_DATA_IN_SHARED_TEMP が ON で、読み取り／書き込み可能なファイルが IQ_SHARED_TEMP に存在する場合
- **例 11** – インメモリ RLV ストアでローレベルのバージョン管理とリアルタイムストレージを使用できるようにしたテーブル tab1 を作成します。

```
CREATE TABLE tab1 ( c1 INT, c2 CHAR(25) ) ENABLE RLV STORE
```

使用法

(先頭に戻る) (257 ページ)

所有者名を指定することにより、別のユーザが使用するテーブルを作成できます。GLOBAL TEMPORARY または LOCAL TEMPORARY が指定されていない場合、テーブルはベーステーブルと呼ばれます。指定すると、テーブルはテンポラリテーブルとなります。

作成されたグローバルテンポラリテーブルは、ベーステーブルと同様にデータベース内に存在し、DROP TABLE 文によって明示的に削除されるまでデータベース内に残ります。テンポラリテーブル内のローは、ローを挿入した接続だけが参照できます。同じ、または異なるアプリケーションからの複数の接続が、同じテンポラリテーブルを同時に使用することもできます。このとき、それぞれの接続で参照できるのは自身のローだけです。この接続は、最初にグローバルテンポラリテーブルを参照し、存在していれば、そのテーブルのスキーマを継承します。接続が終了すると、テンポラリテーブルのローは削除されます。

ローカルテンポラリテーブルを作成するときは、所有者を指定しないでください。たとえば、テンポラリテーブルの作成時に、CREATE TABLE dbo.#temp(coll int) のように所有者を指定すると、ベーステーブルが間違っって作成されます。

その接続に同じ名前のローカルテンポラリテーブルがある場合、ベーステーブルまたはグローバルテンポラリテーブルを作成しようとしても失敗します。これは、新しいテーブルを owner.table がユニークに識別できないからです。

ただし、既存のベーステーブルまたはグローバルテンポラリテーブルとしてなら、ローカルテンポラリテーブルを同じ名前で作成できます。テーブル名への参照は、ローカルテンポラリテーブルにアクセスします。ローカルテンポラリテーブルが最初に解決されるからです。

次のシーケンス例を見てみましょう。

```
CREATE TABLE t1 (c1 int);
INSERT t1 VALUES (9);

CREATE LOCAL TEMPORARY TABLE t1 (c1 int);
INSERT t1 VALUES (8);

SELECT * FROM t1;
```

返される結果は 8 です。ローカルテンポラリテーブルが接続により削除されるまで、`t1` に対する参照はいずれも、ローカルテンポラリテーブル `t1` を参照します。

プロシージャの完了後も保持されるテーブルを作成する場合、`DECLARE LOCAL TEMPORARY TABLE` 文ではなく `CREATE LOCAL TEMPORARY TABLE` 文をプロシージャに使用します。`CREATE LOCAL TEMPORARY TABLE` 文を使用して作成されたローカルテンポラリテーブルは、明示的に削除するか接続が終了するまで保持されます。

`CREATE LOCAL TEMPORARY TABLE` を使用して `IF` 文で作成されたローカルテンポラリテーブルは、`IF` 文が完了した後も保持されます。

SAP Sybase IQ では、SAP Sybase IQ テーブルのテーブルレベルの暗号化の `CREATE TABLE ENCRYPTED` 句はサポートされていません。ただし、`CREATE TABLE ENCRYPTED` 句は、SAP Sybase IQ データベースの SQL Anywhere テーブルではサポートされています。

関連する動作：

- オートコミット

標準

(先頭に戻る) (257 ページ)

- SQL – ISO/ANSI SQL 文法のベンダ拡張。
次にベンダ拡張を示します。
 - `{ IN | ON } dbspace-name` 句
 - **ON COMMIT** 句
 - いくつかのデフォルト値
- SAP Sybase Database 製品 - Adaptive Server でサポートされていますが、いくつか相違があります。
 - **テンポラリテーブル** – ポンド記号 (#) を持つ **CREATE TABLE** 文でテーブル名に先行して、テンポラリテーブルを作成することができます。このようなテンポラリテーブルは、SAP Sybase IQ が宣言するテンポラリテーブルであり、現在の接続でしか使用できません。宣言されたテンポラリテーブルの詳細については、「`DECLARE LOCAL TEMPORARY TABLE` 文」を参照してください。
 - **物理的配置** – テーブルの物理的配置は SAP Sybase IQ と Adaptive Server では方法が異なります。Adaptive Server によってサポートされている **ON segment-name** 句は SAP Sybase IQ でもサポートされていますが、*segment-name* は IQ の DB 領域を参照します。
 - **制約** – SAP Sybase IQ は名前付き制約と名前付きデフォルトをサポートしませんが、制約とデフォルトの定義をデータ型定義にカプセル化できるユー

ザ定義データ型はサポートします。また、**CREATE TABLE** 文の明示的なデフォルトと **CHECK** 条件もサポートします。

- **NULL** – (デフォルト) デフォルトで、Adaptive Server のカラムは、NOT NULL にデフォルト設定されていますが、SAP Sybase IQ でのデフォルト設定は NULL であり、NULL 値が許可されています。この設定は、ALLOW_NULLS_BY_DEFAULT オプションを使用して制御できます。「ALLOW_NULLS_BY_DEFAULT オプション [TSQL]」を参照してください。データ定義文を転送可能にするには、NULL または NOT NULL を明示的に指定します。

パーミッション

(先頭に戻る) (257 ページ)

テーブルタイプ	必要な権限
IQ メインストア内のベーステーブル	<p>自分が所有するテーブル – テーブルが作成される DB 領域に対する CREATE 権限が必要。次のいずれかも必要。</p> <ul style="list-style-type: none"> • CREATE TABLE システム権限 • CREATE ANY OBJECT システム権限 <p>任意のユーザが所有するテーブル – テーブルが作成される DB 領域に対する CREATE 権限が必要。次のいずれかも必要。</p> <ul style="list-style-type: none"> • CREATE ANY TABLE システム権限 • CREATE ANY OBJECT システム権限
グローバルテンポラリテーブル	<p>自分が所有するテーブル – 次のいずれかが必要。</p> <ul style="list-style-type: none"> • CREATE TABLE システム権限 • CREATE ANY OBJECT システム権限 <p>任意のユーザが所有するテーブル – 次のいずれかが必要。</p> <ul style="list-style-type: none"> • CREATE ANY TABLE システム権限 • CREATE ANY OBJECT システム権限

テーブルタイプ	必要な権限
プロキシ テーブル	<p>自分が所有するテーブル – 次のいずれかが必要。</p> <ul style="list-style-type: none"> • CREATE PROXY TABLE システム権限 • CREATE ANY TABLE システム権限 • CREATE ANY OBJECT システム権限 <p>任意のユーザが所有するテーブル – 次のいずれかが必要。</p> <ul style="list-style-type: none"> • CREATE ANY TABLE システム権限 • CREATE ANY OBJECT システム権限

参照：

- ALLOW_NULLS_BY_DEFAULT オプション [TSQL] (603 ページ)
- ALTER TABLE 文 (67 ページ)
- CREATE DBSPACE 文 (150 ページ)
- CREATE INDEX 文 (179 ページ)
- DECLARE LOCAL TEMPORARY TABLE 文 (306 ページ)
- DROP 文 (319 ページ)
- FP_NBIT_AUTOSIZE_LIMIT オプション (667 ページ)
- FP_NBIT_ENFORCE_LIMITS オプション (668 ページ)
- FP_NBIT_LOOKUP_MB オプション (672 ページ)
- FP_NBIT_ROLLOVER_MAX_MB オプション (673 ページ)

CREATE TEXT CONFIGURATION 文

テキスト設定オブジェクトを作成します。

注意： この文は、非構造化データ分析 (IQ_UDA) ライセンスを必要とします。。

クイックリンク：

「パラメータ」 (277 ページ)

「例」 (277 ページ)

「使用法」 (277 ページ)

「パーミッション」 (277 ページ)

構文

```
CREATE TEXT CONFIGURATION [ owner.]new-config-name
FROM [ owner.]existing-config-name
```

パラメータ

(先頭に戻る) (276 ページ)

- **FROM** – 新しいテキスト設定オブジェクトを作成するためのテンプレートとして使用するテキスト設定オブジェクトの名前を指定します。デフォルトのテキスト設定オブジェクトの名前は、**DEFAULT_CHAR** および **DEFAULT_NCHAR** です。**DEFAULT_CHAR** は SAP Sybase IQ テーブルのみでサポートされており、**DEFAULT_NCHAR** は SQL Anywhere テーブルのみでサポートされています。

例

(先頭に戻る) (276 ページ)

- **例 1** – `default_char` テキスト設定オブジェクトを使用してテキスト設定オブジェクト `max_term_sixteen` を作成し、**ALTER TEXT CONFIGURATION** を使用して `max_term_sixteen` の単語の最大長を 16 に変更します。

```
CREATE TEXT CONFIGURATION max_term_sixteen FROM default_char;
```

```
ALTER TEXT CONFIGURATION max_term_sixteen MAXIMUM TERM LENGTH 16;
```

使用法

(先頭に戻る) (276 ページ)

別のテキスト設定オブジェクトをテンプレートとして使用してテキスト設定オブジェクトを作成し、必要に応じて **ALTER TEXT CONFIGURATION** 文を使用してオプションを変更します。

データベース内のすべてのテキスト設定オブジェクトとその設定のリストを表示するには、**SYSTEXTCONFIG** システムビューに対してクエリを実行します。

関連する動作：

- オートコミット。

パーミッション

(先頭に戻る) (276 ページ)

自分が所有するテキスト設定オブジェクトの場合、次のものがが必要です。

- **CREATE TEXT CONFIGURATION** システム権限

任意のユーザが所有するテキスト設定オブジェクトの場合、次のいずれかが必要です。

- CREATE ANY TEXT CONFIGURATION システム権限
- CREATE ANY OBJECT システム権限

すべてのテキスト設定オブジェクトに PUBLIC アクセス権があります。TEXT インデックスを作成する権限を持つユーザは、任意のテキスト設定オブジェクトを使用できます。

CREATE TEXT INDEX 文

TEXT インデックスを作成して、使用するテキスト設定オブジェクトを指定します。

注意：この文は、非構造化データ分析 (IQ_UDA) ライセンスを必要とします。。

クイックリンク：

「パラメータ」 (278 ページ)

「例」 (279 ページ)

「使用法」 (279 ページ)

「パーミッション」 (279 ページ)

構文

```
CREATE TEXT INDEX text-index-name
  ON [ owner. ] table-name ( column-name, ... )
  [ IN dbspace-name ]
  [ CONFIGURATION [ owner. ] text-configuration-name ]
  [ IMMEDIATE REFRESH ]
```

パラメータ

(先頭に戻る) (278 ページ)

- **ON** – TEXT インデックスを構築するテーブルとカラムを指定します。
- **IN** – TEXT インデックスが格納される DB 領域を指定します。この句を指定しない場合、TEXT インデックスは、基本となるテーブルと同じ DB 領域に作成されます。
- **CONFIGURATION** – TEXT インデックスの作成時に使用するテキスト設定オブジェクトを指定します。この句を指定しない場合、default_char テキスト設定オブジェクトが使用されます。

- **IMMEDIATE REFRESH** – (デフォルト) 基本となるテーブルでの変更によって TEXT インデックスのデータが影響を受けるたびに TEXT インデックスを更新します。SAP Sybase IQ メインストアのテーブルでは、この値のみが許可されます。作成後に IMMEDIATE REFRESH 句を変更することはできません。

例

(先頭に戻る) (278 ページ)

- **例 1** – max_term_sixteen テキスト設定オブジェクトを使用して、iqdemo データベース内の Customers テーブルの CompanyName カラムに TEXT インデックス myTxtIdx を作成します。

```
CREATE TEXT INDEX myTxtIdx ON Customers (CompanyName );
CONFIGURATION max_term_sixteen;
```

使用法

(先頭に戻る) (278 ページ)

ビューやテンポラリテーブル、および IN SYSTEM マテリアライズドビューに TEXT インデックスを作成することはできません。BEGIN PARALLEL IQ...END PARALLEL IQ 文では CREATE TEXT INDEX はサポートされていません。

関連する動作：

- オートコミット。

パーミッション

(先頭に戻る) (278 ページ)

次のいずれかが必要です。

- CREATE ANY INDEX システム権限に加え、インデックスが作成される DB 領域に対する CREATE 権限
- CREATE ANY OBJECT システム権限

CREATE TRIGGER 文

テーブル内にトリガを作成します。

構文

```
CREATE [ OR REPLACE ] TRIGGER trigger-name trigger-type
{ trigger-event-list | UPDATE OF column-list }
```

```
[ ORDER integer ] ON table-name
[ REFERENCING [ OLD AS old-name ]
  [ NEW AS new-name ]
  [ REMOTE AS remote-name ] ]
[ FOR EACH { ROW | STATEMENT } ]
[ WHEN ( search-condition ) ]
trigger-body
```

```
column-list : column-name[, ...]
```

```
trigger-type :
BEFORE
| AFTER
| INSTEAD OF
| RESOLVE
```

```
trigger-event-list : trigger-event[, ... ]
```

```
trigger-event :
DELETE
| INSERT
| UPDATE ( column-name )
| UPDATING [ ( column-name-string ) ]
```

```
trigger-body :BEGIN 文。
```

パラメータ

OR REPLACE 句 – OR REPLACE を指定すると、新しいトリガが作成されるか、同じ名前前の既存のトリガが置き換えられます。

trigger-event – トリガは次のイベントによって起動できます。DELETE、INSERT、または UPDATE イベントに複数のトリガを定義するか、UPDATE OF *column-list* イベントに 1 つのトリガを定義できます。

- **DELETE 句** – 関連するテーブルのローが削除されると、呼び出されます。
- **INSERT 句** – このトリガに関連するテーブルに新しいローが挿入されると、呼び出されます。
- **UPDATE 句** – 関連するテーブルのローが更新されると、呼び出されます。

UPDATE 句を指定する場合は、REFERENCING 句も指定して構文エラーを回避する必要があります。

- **UPDATE OF *column-list* 句** – 関連するテーブルのローが更新され、*column-list* のカラムが修正されると、呼び出されます。このタイプのトリガイベントは、*trigger-event-list* では使用できません。このトリガ用に定義されたトリガイベントであることが必要です。この句は、INSTEAD OF トリガでは使用できません。

処理が必要なイベントごとにトリガを個別に作成できます。または、共有するアクションや、イベントに応じたアクションが複数ある場合は、すべてのイベ

ントに対して1つのトリガを作成し、IF文を使用して実行するアクションを区別できます。

- **UPDATING 句** – UPDATING の引数は、引用符付きの文字列です (たとえば、UPDATING ('mycolumn'))。UPDATE の引数は、識別子です (たとえば、UPDATE (mycolumn))。この2つのバージョンは相互運用可能で、他のベンダが提供する DBMS の SQL ダイアレクトとの互換性のために用意されています。

UPDATING 句を指定する場合は、REFERENCING 句も指定して構文エラーを回避する必要があります。

trigger-type – ローレベルのトリガを定義して、挿入、更新、または削除の前 (BEFORE)、後 (AFTER) または代わり (INSTEAD OF) に実行できます。文レベルのトリガは、文の INSTEAD OF または AFTER の実行を定義できます。

BEFORE UPDATE トリガは、ローを対象に UPDATE が実行されるたび、新しい値が古い値と異なるかどうかに関係なく起動されます。つまり、BEFORE UPDATE トリガに *column-list* が指定されている場合は、UPDATE 文の SET 句に *column-list* 内のカラムがあると、トリガが起動します。AFTER UPDATE トリガに *column-list* が指定されている場合は、*column-list* 内のいずれかのカラムの値が UPDATE 文によって変更された場合にのみ、トリガが起動します。

INSTEAD OF トリガは、通常のビューに定義できるトリガの唯一の形式です。INSTEAD OF トリガは、他の動作を、トリガ動作で置換します。INSTEAD OF トリガが起動すると、トリガ動作はスキップされ、指定された動作が実行されます。INSTEAD OF トリガは、ローレベルまたは文レベルのトリガとして定義できます。文レベルの INSTEAD OF トリガは、ローレベルのすべての処理を含め、文全体を置換します。文レベルの INSTEAD OF トリガが起動すると、その文の結果としてローレベルのトリガが起動することはありません。ただし、文レベルのトリガの本文が、その他の処理を実行するため、結果として、その他のローレベルのトリガが実行されます。

INSTEAD OF トリガを定義する場合は、UPDATE OF *column-list* 句、ORDER 句、または WHEN 句は使用できません。

ORDER 句 – 同じタイミング (before、after、resolve) で起動する同じタイプ (insert、update、delete) の追加トリガを定義する場合は、ORDER 句を指定してトリガを起動する順序をデータベースサーバに指示します。同じタイミングで起動するように設定された同じタイプのトリガの間では、順序番号をユニークにします。ユニークでない順序番号を指定すると、エラーが返されます。順序番号は、連続した順番にする必要はありません (たとえば、1、12、30 と指定できます)。データベースサーバは、最も小さい番号が付いたトリガから順に起動します。

ORDER 句を省略するか、0 を指定すると、データベースサーバは順序番号 1 を割り当てます。ただし、すでに同じタイプの別のトリガが 1 に設定されている場合は、エラーが返されます。

追加トリガを追加する場合は、トリガの動作が相互に影響するかどうかによって、イベントの同じタイプの既存トリガを修正する必要がある可能性があります。相互に影響しない場合、新しいトリガには既存のトリガより大きな ORDER 値が必要です。相互に影響する場合は、他のトリガの動作を検討する必要があり、これらのトリガが起動する順序の変更が必要となる可能性があります。

ORDER 句は、INSTEAD OF トリガではサポートされません。テーブルまたはビューに定義されたタイプ (insert、update、delete) ごとに、1 つの INSTEAD OF トリガだけを使用できます。

REFERENCING 句 – REFERENCING OLD 句と REFERENCING NEW 句を使用すると、挿入、削除、または更新されたローを参照できます。この句では、UPDATE は削除とそれに続く挿入として取り扱われます。

INSERT は REFERENCING NEW 句を使います。これは挿入されたローを表します。REFERENCING OLD 句はありません。

DELETE は REFERENCING OLD 句を使います。これは削除されたローを表します。REFERENCING NEW 句はありません。

UPDATE は、REFERENCING OLD 句を使うときは更新前のローを表し、REFERENCING NEW 句を使うときは更新後のローを表します。

REFERENCING OLD と REFERENCING NEW の意味は、トリガがローレベルのトリガなのか、文レベルのトリガなのかによって異なります。ローレベルのトリガの場合、REFERENCING OLD 句を使うと、更新または削除する前のローの値を参照できます。また、REFERENCING NEW 句を使用すると、挿入または更新された値を参照できます。OLD ローと NEW ローは、BEFORE と AFTER トリガの中で参照できます。REFERENCING NEW 句を使うと、BEFORE トリガの新しいローを修正してから、挿入または更新操作を行うことができます。

文レベルのトリガの場合、REFERENCING OLD と REFERENCING NEW 句は、ローの古い値と新しい値を保持している宣言されたテンポラリテーブルを参照します。

FOR EACH 句 – トリガをローレベルのトリガとして宣言するには、FOR EACH ROW 句を使用します。トリガを文レベルのトリガとして宣言するには、FOR EACH STATEMENT 句を使用するか、または FOR EACH 句を省略します。文レベルのトリガを宣言する場合は、わかりやすくするために、FOR EACH STATEMENT 句を指定することをおすすめします。

WHEN 句 – 探索条件が真と評価されたローに対してだけ、トリガが起動されます。WHEN 句は、ローレベルトリガと一緒にだけ使用できます。この句は、INSTEAD OF トリガでは使用できません。

trigger-body – トリガの本文には、トリガアクションが生じたときに実行されるアクションが含まれます。これは、BEGIN 文で構成されます。

BEGIN 文にトリガオペレーション条件を指定できます。トリガオペレーション条件は、トリガを起動したトリガイベントに従って動作を行います。たとえば、トリガが更新と削除の両方のために起動するように定義されている場合は、2つの条件に異なる動作を指定できます。

備考

CREATE TRIGGER 文は、データベースのテーブルに関連するトリガを作成し、データベースにトリガを格納します。

マテリアライズドビューにトリガを定義することはできません。定義すると、SQLE_INVALID_TRIGGER_MATVIEW エラーが返されます。

トリガをローレベルのトリガとして宣言すると、各ローを修正する前または後にトリガが実行されます。また、トリガを文レベルのトリガとして宣言すると、トリガ文全体が完了してから、トリガが実行されます。

CREATE TRIGGER はテーブルにテーブルロックを設定して、テーブルを排他的に使用します。

権限

CREATE ANY TRIGGER または CREATE ANY OBJECT のシステム権限が必要です。さらに、そのトリガが作成されるテーブルの所有者であるか、または次のいずれかの権限を持っていることも必要です。

そのテーブルに対する ALTER 権限

ALTER ANY TABLE システム権限

ALTER ANY OBJECT システム権限

別のユーザが所有するビューのトリガを作成するには、ALTER ANY TRIGGER または ALTER ANY OBJECT システム権限のどちらかを持っていて、ALTER ANY VIEW または ALTER ANY OBJECT システム権限のどちらかを持っている必要があります。

関連する動作

オートコミット。

標準と互換性

- **SQL/2008** – CREATE TRIGGER は、SQL/2008 標準のオプションの SQL 言語機能 T211、"Basic trigger capability" の一部です。ROW トリガはオプションの SQL 言語機能 T212 であり、INSTEAD OF トリガはオプションの SQL 言語機能 T213 です。

SAP Sybase IQ トリガの一部の機能は、ベンダー拡張です。これらのコードを以下に示します。

- オプションの OR REPLACE 構文。既存のトリガを置き換える場合、新しいトリガインスタンスの作成の認証は省略されます。
- ORDER 句。SQL/2008 では、トリガは作成順に起動されます。
- RESOLVE トリガはベンダー拡張です。
- **Transact-SQL** – ROW トリガと RESOLVE トリガは、Adaptive Server Enterprise でサポートされていません。Transact-SQL の INSTEAD OF トリガは、Adaptive Server Enterprise ではサポートされていますが、SAP Sybase IQ の Transact-SQL ダイアレクトではサポートされていません。Transact-SQL トリガは、異なる構文で定義されます。

例

この例は、文レベルのトリガを作成します。最初に、この CREATE TABLE 文に示されているようにテーブルを作成します (CREATE TABLE システム権限が必要)。

```
CREATE TABLE t0
( id INTEGER NOT NULL,
  times TIMESTAMP NULL DEFAULT CURRENT_TIMESTAMP,
  remarks TEXT NULL,
  PRIMARY KEY ( id )
);
```

次に、このテーブルについて文レベルのトリガを作成します。

```
CREATE TRIGGER myTrig AFTER INSERT ORDER 4 ON t0
REFERENCING NEW AS new_name
FOR EACH STATEMENT
BEGIN
  DECLARE @id1 INTEGER;
  DECLARE @times1 TIMESTAMP;
  DECLARE @remarks1 LONG VARCHAR;
  DECLARE @err_notfound EXCEPTION FOR SQLSTATE VALUE '02000';
  //declare a cursor for table new_name
  DECLARE new1 CURSOR FOR
    SELECT id, times, remarks FROM new_name;
  OPEN new1;
  //Open the cursor, and get the value
  LoopGetRow:
  LOOP
    FETCH NEXT new1 INTO @id1, @times1, @remarks1;
    IF SQLSTATE = @err_notfound THEN
```



```

        LEAVE LoopGetRow
      END IF;
      //print the value or for other use
      PRINT (@remarks1);
    END LOOP LoopGetRow;
  CLOSE new1
END;

```

次の例は、前の例で作成された myTrig トリガを置き換えます。

```

CREATE OR REPLACE TRIGGER myTrig AFTER INSERT ORDER 4 ON t0
REFERENCING NEW AS new_name
FOR EACH STATEMENT
BEGIN
  FOR l1 AS new1 CURSOR FOR
    SELECT id, times, remarks FROM new_name
  DO
    //print the value or for other use
    PRINT (@remarks1);
  END FOR;
END;

```

次の例は、BEFORE UPDATE トリガで REFERENCING NEW を使用方法を示します。次の例では、新規 Employees テーブルの郵便番号が大文字になるようにしています。この文を実行するには、GROUPO.Employees に対する SELECT、ALTER、UPDATE 権限が必要です。

```

CREATE TRIGGER emp_upper_postal_code
BEFORE UPDATE OF PostalCode
ON GROUPO.Employees
REFERENCING NEW AS new_emp
FOR EACH ROW
WHEN ( ISNUMERIC( new_emp.PostalCode ) = 0 )
BEGIN
  -- Ensure postal code is uppercase (employee might be
  -- in Canada where postal codes contain letters)
  SET new_emp.PostalCode = UPPER(new_emp.PostalCode)
END;

UPDATE GROUPO.Employees SET state='ON', PostalCode='n2x 4y7' WHERE
EmployeeID=191;
SELECT PostalCode FROM GROUPO.Employees WHERE EmployeeID = 191;

```

次の例は、BEFORE DELETE トリガで REFERENCING OLD を使用方法を示します。この例は、Employees テーブルから、退職していない従業員が削除されないようにします。

```

CREATE TRIGGER TR_check_delete_employee
BEFORE DELETE
ON Employees
REFERENCING OLD AS current_employee
FOR EACH ROW WHEN ( current_employee.Terminate IS NULL )
BEGIN
  RAISERROR 30001 'You cannot delete an employee who has not been

```

```
fired';
END;
```

次の例は、**BEFORE UPDATE** トリガで **REFERENCING NEW** と **REFERENCING OLD** を使用する方法を示します。この例は、従業員の給料に減給が生じないようにします。

```
CREATE TRIGGER TR_check_salary_decrease
  BEFORE UPDATE
  ON GROUPO.Employees
  REFERENCING OLD AS before_update
  NEW AS after_update
FOR EACH ROW
BEGIN
  IF after_update.salary < before_update.salary THEN
    RAISERROR 30002 'You cannot decrease a salary';
  END IF;
END;
```

次の例は、**BEFORE INSERT** トリガと **UPDATE** トリガで **REFERENCING NEW** を使用する方法を示します。この例では、**SalesOrderItems** テーブルのローで挿入や更新が行われる前に起動するトリガを作成します。

```
CREATE TRIGGER TR_update_date
  BEFORE INSERT, UPDATE
  ON GROUPO.SalesOrderItems
  REFERENCING NEW AS new_row
FOR EACH ROW
BEGIN
  SET new_row.ShipDate = CURRENT_TIMESTAMP;
END;
```

次のトリガでは、トリガが起動されるきっかけとなったアクションを示すメッセージが、**Interactive SQL** の [結果] ウィンドウ枠の [メッセージ] タブに表示されます。

```
CREATE TRIGGER tr BEFORE INSERT, UPDATE, DELETE
ON sample_table
REFERENCING OLD AS t1old
FOR EACH ROW
BEGIN
  DECLARE msg varchar(255);
  SET msg = 'This trigger was fired by an ';
  IF INSERTING THEN
    SET msg = msg || 'insert'
  ELSEIF DELETING THEN
    set msg = msg || 'delete'
  ELSEIF UPDATING THEN
    set msg = msg || 'update'
  END IF;
  MESSAGE msg TO CLIENT
END;
```

CREATE USER 文

ユーザを作成します。

クイックリンク：

「パラメータ」 (287 ページ)

「例」 (288 ページ)

「使用法」 (288 ページ)

「標準」 (288 ページ)

「パーミッション」 (289 ページ)

構文

```
CREATE USER user-name [ IDENTIFIED BY password ]  
[ LOGIN POLICY policy-name ]  
[ FORCE PASSWORD CHANGE { ON | OFF } ]
```

パラメータ

(先頭に戻る) (287 ページ)

- **user-name** – ユーザの名前。
- **IDENTIFIED BY** – ユーザのパスワード。
- **policy-name** – ユーザを割り当てるログインポリシーの名前。ログインポリシーを指定しないと、変更は行われません。
- **FORCE PASSWORD CHANGE** – ログイン時にユーザが新しいパスワードを指定する必要があるかどうかを制御します。この設定は、ユーザのログインポリシーの **PASSWORD_EXPIRY_ON_NEXT_LOGIN** オプションの設定を上書きします。

注意： この機能は現在、SAP Control Center へのログイン時に実装されなくなっています。ユーザは、パスワードの変更を要求されません。ただし、SAP Control Center 外から (たとえば Interactive SQL を使用して) SAP Sybase IQ にログインする際には要求されます。

- **password** – ユーザにパスワードを指定する必要はありません。パスワードのないユーザは、データベースに接続できません。これは、ロールを作成して、そのロールユーザ ID を使用したユーザをデータベースに接続させないようにする場合に便利です。ユーザ ID には、有効な識別子を使用します。ユーザ ID とパスワードで禁止されていることは次のとおりです。

- 最初の文字をスペース、一重引用符または二重引用符にする
- 最後の文字をスペースにする
- セミコロンを含める

パスワードには有効な識別子、または一重引用符で囲まれた文字列 (最大 255 文字) を指定できます。パスワードでは大文字と小文字を区別します。パスワードには 7 ビット ASCII 文字で使用してください。それ以外の文字を使用すると、データベースサーバがクライアントの文字セットを UTF-8 に変換できない場合に、パスワードが正しく機能しないことがあります。

VERIFY_PASSWORD_FUNCTION オプションを使用して、パスワードルール (パスワードには 1 つ以上の数字が含まれている必要があるなど) を実装する関数を指定できます。パスワード検証関数を使用する場合は、**GRANT CONNECT** 文に複数のユーザ ID とパスワードを指定することはできません。

ユーザパスワードのハッシュに使用される暗号化アルゴリズムは、FIPS 認定の暗号化サポートです。

- DLL は、dbfips10.dll という名前です。
- HASH 関数では、アルゴリズム SHA1_FIPS と SHA256_FIPS を使用できません。
- -fips サーバオプションを指定したときに FIPS 認定でないアルゴリズムを HASH 関数に指定すると、データベースサーバでは SHA1 の代わりに SHA1_FIPS が、SHA256 の代わりに SHA256_FIPS が使用されます。また、MD5 を使用した場合はエラーが返されます (MD5 は FIPS 認定のアルゴリズムではありません)。
- -fips オプションを指定した場合は、パスワードハッシュ処理に SHA256_FIPS が使用されます。

例

(先頭に戻る) (287 ページ)

- **例 1** - SQLTester という名前のユーザを作成し、パスワードを welcome に設定します。SQLTester ユーザは Test1 ログインポリシーに割り当てられ、パスワードは次回ログイン時に有効期限切れになります。

```
CREATE USER SQLTester IDENTIFIED BY welcome
LOGIN POLICY Test1
FORCE PASSWORD CHANGE ON;
```

標準

(先頭に戻る) (287 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Adaptive Server ではサポートされていません。

パーミッション

(先頭に戻る) (287 ページ)

MANAGE ANY USER システム権限が必要です。

参照：

- COMMENT 文 (123 ページ)
- CREATE LOGIN POLICY 文 (194 ページ)
- DROP LOGIN POLICY 文 (330 ページ)
- DROP USER 文 (346 ページ)
- GRANT ROLE 文 (394 ページ)
- GRANT システム権限文 (402 ページ)
- VERIFY_PASSWORD_FUNCTION オプション (807 ページ)
- ALTER LOGIN POLICY 文 (32 ページ)

CREATE VARIABLE 文

指定したデータ型の新しい変数を作成します。

クイックリンク：

「パラメータ」 (290 ページ)

「例」 (290 ページ)

「使用法」 (291 ページ)

「標準」 (291 ページ)

「パーミッション」 (291 ページ)

構文

```
CREATE [OR REPLACE] VARIABLE identifier data-type
    [{= | DEFAULT} initial-value]
```

initial-value - (構文に戻る)

special-value

```
| string
| [ - ] number
| ( constant-expression )
| built-in-function ( constant-expression )
```

```

| NULL
special-value - (back to initial-value)
CURRENT
{ DATABASE
  | DATE
  | PUBLISHER
  | TIME
  | TIMESTAMP
  | USER
  | UTC TIMESTAMP }
| USER

```

パラメータ

(先頭に戻る) (289 ページ)

- **OR REPLACE** – OR REPLACE 句を指定すると、指定された変数がすでに存在する場合は、その変数が削除され、定義が置き換えられます。OR REPLACE 句は、SQL スクリプトの **VAREXISTS** 関数の代わりに使用できます。
- **initial-value** – データ型は、*data-type* で定義された型と一致する必要があります。

initial-value を指定すると、変数はその値に設定されます。*initial-value* を指定しないと、変数に NULL 値が設定されます。この値は、**SET** 文によって別の値が割り当てられるまで変わりません。

例

(先頭に戻る) (289 ページ)

- **例 1** – 次のコードフラグメントは、データベースに大きなテキスト値を挿入します。

```

EXEC SQL BEGIN DECLARE SECTION;
char buffer[5000];
EXEC SQL END DECLARE SECTION;
EXEC SQL CREATE VARIABLE hold_blob VARCHAR;
EXEC SQL SET hold_blob = '';
for(;;) {
    /* read some data into buffer ... */
    size = fread( buffer, 1, 5000, fp );
    if( size <= 0 ) break;
    /* add data to blob using concatenation
    Note that concatenation works for binary
    data too! */
    EXEC SQL SET hold_blob = hold_blob || :buffer;
}
EXEC SQL INSERT INTO some_table VALUES ( 1, hold_blob );
EXEC SQL DROP VARIABLE hold_blob;

```

使用法

(先頭に戻る) (289 ページ)

変数は、カラム名を使用できる場所なら SQL 式のどこでも使うことができます。変数と同じ名前のカラム名がある場合は、変数値が使用されます。

変数は、カラム名を使用できる場所なら SQL 式のどこでも使うことができます。名前の決定は次のように実行されます。

- クエリの SELECT リストで指定した任意のエイリアスと一致。
- 任意の参照先テーブルのカラム名と一致。
- 名前が変数であると仮定。

変数は現在の接続に属し、データベースから切断するか、**DROP VARIABLE** 文を使用すると消去されます。変数は、他の接続からは参照できません。**COMMIT** 文または **ROLLBACK** 文は、変数に影響を与えません。

CREATE VARIABLE 文で作成された変数は、その文が (**BEGIN...END**) 文の内部で発行されたとしても、接続が存続する間は存続します。(**BEGIN...END**) 文の内部(ストアードプロシージャ内など)でのみ存続する変数を作成するには、**DECLARE** を使用します。

変数は、Embedded SQL プログラムから **INSERT** 文または **UPDATE** 文の大きなテキストまたはバイナリオブジェクトを作成するときに役立ちます。

プロシージャとトリガのローカル変数は、複合文の中で宣言します。

標準

(先頭に戻る) (289 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Adaptive Server ではサポートされていません。

パーミッション

(先頭に戻る) (289 ページ)

なし

参照：

- BEGIN ... END 文 (108 ページ)
- DECLARE 文 (297 ページ)
- DROP VARIABLE 文 (347 ページ)
- SET 文 [ESQL] (534 ページ)

CREATE VIEW 文

データベースにビューを作成します。ビューを使用すると、格納されている方法とは異なる形でデータを参照できます。

クイックリンク：

「パラメータ」 (292 ページ)

「例」 (293 ページ)

「使用法」 (293 ページ)

「標準」 (293 ページ)

「パーミッション」 (294 ページ)

構文

```
CREATE [ OR REPLACE ] VIEW
... [ owner.]view-name [ ( column-name [ , ... ] ) ]
... AS select-without-order-by
... [ WITH CHECK OPTION ]
```

パラメータ

(先頭に戻る) (292 ページ)

- **OR REPLACE** – 既存のビューを同じ名前でも置き換えます。既存のパーミッションは保持されますが、そのビューの **INSTEAD OF** トリガは削除されます。
- **column-name** – ビューのカラムの名前を指定します。column name リストが指定されていない場合、ビューのカラムには select リスト項目にある名前が付けられます。select リスト項目にある名前を使用するには、各項目が単純なカラム名であるか、エイリアス名が指定されている必要があります (「SELECT 文」を参照してください)。
- **view-name** – ビューのデフォルトの所有者は、現在のユーザ ID です。**SELECT** 文、**DELETE** 文、**UPDATE** 文、および **INSERT** 文では、ビュー名をテーブル名の代わりに使用できます。ただし、ビューは、データベース内にテーブルとして物理的に存在するわけではありません。ビューは使用するたびに抽出されます。ビューは、**CREATE VIEW** 文で指定した **SELECT** 文の結果として派生的に作成されます。ビューで使用するテーブル名は、テーブルの所有者のユーザ ID によって修飾する必要があります。そのようにしないと、別のユーザ ID ではテーブルを見つけることができなかつたり、間違ったテーブルが取得される可能性があります。

- **AS** - ビューに基づいている **SELECT** 文には、**ORDER BY** 句、**SELECT** リスト内のサブクエリ、および **TOP** または **FIRST** 修飾を含めないようにする必要があります。**GROUP BY** 句を含めたり、**UNION** にすることはできません。
- **WITH CHECK OPTION** - **SELECT** 文で定義されたビューの基準と一致しないビューへの更新や挿入がすべて拒否されます。ただし、現在は、**SAP Sybase IQ** でこのオプションが無視されます (互換性を考慮して構文はサポートされています)。

例

(先頭に戻る) (292 ページ)

- **例 1** - 男性従業員のみを表示するビューを作成します。このビューはベーステーブルと同じカラム名を持ちます。

```
CREATE VIEW male_employee
AS SELECT *
FROM Employees
WHERE Sex = 'M'
```

- **例 2** - 従業員とその所属部署を表示するビューを作成します。

```
CREATE VIEW emp_dept
AS SELECT Surname, GivenName, DepartmentName
FROM Employees JOIN Departments
ON Employees.DepartmentID = Departments.DepartmentID
```

使用法

(先頭に戻る) (292 ページ)

ビューでは、**IDENTITY** カラムまたは **AUTOINCREMENT** カラムを追加または削除することはできません。

ビューを定義する **SELECT** 文が **GROUP BY** 句や集合関数を含まない場合、または **UNION** 操作が関係しない場合は、ビューを更新できます。ビューを更新すると、基になっているテーブルも更新されます。

関連する動作：

- オートコミット。

標準

(先頭に戻る) (292 ページ)

- **SQL - ISO/ANSI SQL** 文法のベンダ拡張。
- **SAP Sybase Database 製品 - Adaptive Server** でサポートされています。

パーミッション

(先頭に戻る) (292 ページ)

自分が所有するビューの場合、CREATE VIEW システム権限が必要です。次のいずれかも必要です。

- SELECT ANY TABLE システム権限
- ビューの基礎となるテーブルに対する SELECT オブジェクトパーミッション

任意のユーザが所有するビューの場合、次のいずれかが必要です。

- CREATE ANY VIEW システム権限
- CREATE ANY OBJECT システム権限
- 次のいずれかも必要です。
 - SELECT ANY TABLE システム権限
 - ビューの基礎となるテーブルに対する SELECT オブジェクト権限

自分が所有するマテリアライズドビューの場合、CREATE MATERIALIZED VIEW システム権限が必要です。次のいずれかも必要です。

- CREATE ANY OBJECT システム権限
- マテリアライズドビューが作成される DB 領域に対する CREATE 権限
- 次のいずれかも必要です。
 - SELECT ANY TABLE システム権限
 - マテリアライズドビューの基礎となるテーブルに対する SELECT 権限

任意のユーザが所有するマテリアライズドビューの場合、次のいずれかが必要です。

- CREATE ANY MATERIALIZED VIEW システム権限
- CREATE ANY OBJECT システム権限
- 次のいずれかも必要です。
 - CREATE ANY OBJECT システム権限
 - マテリアライズドビューが作成される DB 領域に対する CREATE 権限さらに、次のいずれかも必要です。
 - SELECT ANY TABLE システム権限
 - マテリアライズドビューの基礎となるテーブルに対する SELECT 権限

参照：

- CREATE TABLE 文 (257 ページ)
- DROP 文 (319 ページ)
- SELECT 文 (523 ページ)

DEALLOCATE DESCRIPTOR 文 [ESQL]

SQL 記述子領域に関連付けられているメモリを解放します。

クイックリンク：

「例」 (295 ページ)

「使用法」 (295 ページ)

「標準」 (295 ページ)

「パーミッション」 (295 ページ)

構文

```
DEALLOCATE DESCRIPTOR
  descriptor-name:
string
```

例

(先頭に戻る) (295 ページ)

• 例 1

「ALLOCATE DESCRIPTOR 文 [ESQL]」を参照してください。

使用法

(先頭に戻る) (295 ページ)

データ項目、インジケータ変数、構造体自体など、記述子領域に関連付けられているすべてのメモリを解放します。

標準

(先頭に戻る) (295 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Open Client/Open Server でサポートされています。

パーミッション

(先頭に戻る) (295 ページ)

なし

参照：

- ALLOCATE DESCRIPTOR 文 [ESQL] (7 ページ)
- SET DESCRIPTOR 文 [ESQL] (540 ページ)

宣言セクション [ESQL]

Embedded SQL プログラムでホスト変数を宣言します。ホスト変数を使用して、データベースとデータを交換します。

クイックリンク：

「例」 (296 ページ)

「使用法」 (296 ページ)

「標準」 (296 ページ)

「パーミッション」 (297 ページ)

構文

```
EXEC SQL BEGIN DECLARE SECTION;  
... C declarations  
EXEC SQL END DECLARE SECTION;
```

例

(先頭に戻る) (296 ページ)

• 例 1 –

```
EXEC SQL BEGIN DECLARE SECTION;  
char *emp_lname, initials[5];  
int dept;  
EXEC SQL END DECLARE SECTION;
```

使用法

(先頭に戻る) (296 ページ)

宣言セクションは、**BEGIN DECLARE SECTION** 文と **END DECLARE SECTION** 文で囲まれた C 変数宣言のセクションです。宣言セクションによって SQL プリプロセッサはホスト変数として使用される C 変数を認識します。すべての C 宣言が宣言セクションの中で有効なわけではありません。

標準

(先頭に戻る) (296 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。

パーミッション

(先頭に戻る) (296 ページ)

なし

参照：

- BEGIN ... END 文 (108 ページ)

DECLARE 文

複合文 (**BEGIN... END**) 内で SQL 変数を宣言します。

クイックリンク：

「パラメータ」 (298 ページ)

「例」 (298 ページ)

「使用法」 (298 ページ)

「標準」 (298 ページ)

「パーミッション」 (299 ページ)

構文

DECLARE

```
variable_name [ , ... ]
data-type [{
=
| DEFAULT}
initial-value]
```

initial-value - (構文に戻る)

```
special-value
| string
| [ - ] number
| ( constant-expression )
| built-in-function ( constant-expression )
| NULL
```

special-value - (back to initial-value)

```
CURRENT
{ DATABASE
| DATE
| PUBLISHER
| TIME
| TIMESTAMP
```

```

| USER
| UTC TIMESTAMP }
| USER

```

パラメータ

(先頭に戻る) (297 ページ)

- **initial-value** – 変数はこの値に設定されます。データ型は、*data-type* で定義された型と一致する必要があります。**initial-value** を指定しないと、変数に NULL 値が設定されます。この値は、**SET** 文によって別の値が割り当てられるまで変わりません。

例

(先頭に戻る) (297 ページ)

- **例 1** – 次のバッチは、**DECLARE** 文の使用法を示し、メッセージをサーバウィンドウに表示します。

```

BEGIN
  DECLARE varname CHAR(61);
  SET varname = 'Test name';
  MESSAGE varname;
END

```

使用法

(先頭に戻る) (297 ページ)

DECLARE 文を使用して、プロシージャの本体で使用する変数を宣言します。複合文で宣言した変数は、その複合文の間持続します。また、変数は、複合文内でユニークである必要があります。

プロシージャの本体は複合文で、変数は **BEGIN** の直後に宣言する必要があります。Transact-SQL のプロシージャまたはトリガには、そのような制約はありません。

標準

(先頭に戻る) (297 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - Adaptive Server でサポートされています。
 - Adaptive Server との互換性を維持するには、変数名の前に @ を付けます。
 - Adaptive Server では、プロシージャまたはトリガで宣言される変数はそのプロシージャまたはトリガの存続期間だけ存在します。SAP Sybase IQ では、変数を複合文内で宣言した場合、(SAP Sybase IQ と Transact-SQL のどちらの

複合文で宣言されているかにかかわらず) 変数は複合文の存続期間だけ存在します。

パーミッション

(先頭に戻る) (297 ページ)

なし

参照:

- BEGIN ... END 文 (108 ページ)

DECLARE CURSOR 文 [ESQL] [SP]

カーソルを宣言します。カーソルはクエリの結果を操作する主要な手段です。

クイックリンク:

「パラメータ」 (299 ページ)

「例」 (301 ページ)

「使用法」 (302 ページ)

「標準」 (305 ページ)

「パーミッション」 (305 ページ)

構文

```
DECLARE cursor-name
  [ SCROLL
    | NO SCROLL
    | DYNAMIC SCROLL
  ]
  CURSOR FOR
  { select-statement FOR for-clause
    | statement-name
    | USING variable-name }

for-clause
  READ ONLY | UPDATE
```

パラメータ

(先頭に戻る) (299 ページ)

- **statement-name** – 識別子またはホスト変数。 **PREPARE** 文を使用して文に名前を付けます。カーソルは、準備された **SELECT** または **CALL** に対してのみ宣言できます。
- **SCROLL-SCROLL** として宣言されたカーソルは、**FETCH** 文の **NEXT**、**PRIOR**、**FIRST**、**LAST**、**ABSOLUTE**、**RELATIVE** の各オプションをサポートします。 **SCROLL** カーソルでは、カーソルが開いている間、結果セットにある任意のローをフェッチすることができます。
- **NO SCROLL-NO SCROLL** として宣言されたカーソルは、**FETCH NEXT** と **FETCH ABSOLUTE (0)** のシーク操作のみを使用して、結果セット内の移動が前方向に制限されます。

- **DYNAMIC SCROLL-DYNAMIC SCROLL** として宣言されたカーソルは、**FETCH** 文の **NEXT**、**PRIOR**、**FIRST**、**LAST**、**ABSOLUTE**、**RELATIVE** の各句をサポートします。 **DYNAMIC SCROLL** カーソルでは、カーソルが開いている間、結果セットにある任意のローをフェッチすることができます。

一度カーソルがローを離れたら、そのローに戻ることはできないため、このカーソルに感知性の制限はありません。したがって、**NO SCROLL** カーソルが要求されると、SAP Sybase IQ は最も効率的なカーソル、すなわち **asensitive** カーソルを提供します。

- **READ ONLY** – (デフォルト) **FOR READ ONLY** として宣言されたカーソルは、位置付け **UPDATE** 操作や位置付け **DELETE** 操作には使用できません。

FOR READ ONLY として宣言されたカーソルからは、最初に **FETCH** を実行したときのテーブルのバージョンではなく、カーソルを開いたときにカーソルの宣言に使用されるテーブルのバージョンを認識します。

たとえば、カーソルがフェッチされる時、テーブルからフェッチできるローは 1 つだけです。

```
CREATE TABLE t1 ( c1 INT );
INSERT t1 VALUES ( 1 );

BEGIN
DECLARE t1_cursor CURSOR FOR SELECT * FROM t1
FOR READ ONLY;
OPEN t1_cursor;
INSERT t1 VALUES ( 2 );
FETCH T1_CURSOR;
END
```

- **UPDATE-FOR UPDATE** として宣言されたカーソルでは、カーソルの結果セットを更新できます。更新可能なカーソルには、**asensitive** 動作だけがサポートされます。その他の感知性はすべて無視されます。

カーソルが開くと、更新対象として開かれたすべてのテーブルの排他テーブルロックを取得します。SAP Sybase IQ では、テーブルを変更する文が一度に 1 つ

しか許可されないため、同じトランザクション内で更新対象として開かれたテーブルに対するスタンドアロンの **LOAD TABLE** 文、**UPDATE** 文、**INSERT** 文、**DELETE** 文、**TRUNCATE** 文は許可されません。特定のテーブルで、一度に開くことができる更新可能なカーソルは1つだけです。

更新可能なカーソルは、Open Client に対する場合を除き、スクロール可能です。

- **USING** – ストアドプロシージャやユーザ定義関数の変数にカーソルを宣言できます。この変数はカーソルの **SELECT** 文を含む文字列です。変数は **DECLARE** の処理時に使用できなければならないため、次の方法のいずれかを使用してください。

プロシージャに対するパラメータ。次に例を示します。

```
create function get_row_count(in qry varchar)
returns int
begin
    declare crsr cursor using qry;
    declare rowcnt int;

    set rowcnt = 0;
    open crsr;
    lp: loop
        fetch crsr;
        if SQLCODE <> 0 then leave lp end if;
        set rowcnt = rowcnt + 1;
    end loop;
    return rowcnt;
end
```

変数に値を割り当ててから、別の **BEGIN...END** 内にネストする。次に例を示します。

```
create procedure get_table_name(
    in id_value int, out tabname char(128))

begin
    declare qry varchar;

    set qry = 'select table_name from SYS.ISYSTAB ' ||
        'where table_id=' || string(id_value);
    begin
        declare crsr cursor using qry;

        open crsr;
        fetch crsr into tabname;
        close crsr;
    end
end
```

例

(先頭に戻る) (299 ページ)

- **例 1** – Embedded SQL 内のスクロールカーソルを宣言する方法

```
EXEC SQL DECLARE cur_employee SCROLL CURSOR
FOR SELECT * FROM Employees;
```

- **例 2** – Embedded SQL 内の準備文のためのカーソルを宣言する方法

```
EXEC SQL PREPARE employee_statement
FROM 'SELECT emp_lname FROM Employees';
EXEC SQL DECLARE cur_employee CURSOR
FOR employee_statement ;
```

- **例 3** – ストアドプロシージャでのカーソルの使用法

```
BEGIN
  DECLARE cur_employee CURSOR FOR
    SELECT emp_lname
    FROM Employees;
  DECLARE name CHAR(40);
  OPEN cur_employee;
  LOOP
    FETCH NEXT cur_employee INTO name;
    ...
  END LOOP;
  CLOSE cur_employee;
END
```

使用法

(先頭に戻る) (299 ページ)

DECLARE CURSOR 文は、**SELECT** 文または **CALL** 文に対して指定された名前を持つカーソルを宣言します。

PREPARE 文を使用して Embedded SQL の文に名前を付けます。カーソルは、準備された **SELECT** または **CALL** に対してのみ宣言できます。

SAP Sybase IQ では、1 つのテーブルに対する更新可能なカーソルをサポートしています。

SAP Sybase IQ でサポートされるカーソルの感知性は 1 種類です。この感知性は、基礎データに対する変更の可視性を基準に決められます。SAP Sybase IQ のカーソルはすべて **asensitive** です。つまり、結果セットのメンバーシップ、順序、または値に反映される変更は、カーソルを通して確認できるか、まったく反映されないかのどちらかです。

asensitive カーソルを使用すると、位置付け **UPDATE** 文と位置付け **DELETE** 文による変更がカーソル結果セット内で確認できます。ただし、クライアント側のキャッシュによってこれらの変更の確認が妨げられた場合を除きます。挿入されたローは確認できません。

ローが更新され、開いているカーソルの **WHERE** 句の要件を満たさなくなった場合でも、そのローは表示されます。

カーソルの使用には、効率と一貫性のトレードオフが常に伴います。asensitive カーソルを使用すると、パフォーマンスは向上しますが一貫性が低下します。

LONG VARCHAR データ型と LONG BINARY データ型は、更新可能なカーソルではサポートされません。

スカラユーザ定義関数とユーザ定義の集合関数は、更新可能なカーソルではサポートされません。

次に、SAP Sybase IQ における更新可能なカーソルに対してサポートされるクエリの様子は次のとおりです。

- select リスト内の式は、更新対象のカラムに機能的に依存しないカラムに対して適用されます。
- 任意のサブクエリの動作は asensitive になります。このため、サブクエリから参照されるデータへの変更は、カーソル結果セット内で認識されません。
- **ORDER BY** 句。**ORDER BY** を指定したカラムも更新可能ですが、結果セットが再度順序付けされることはありません。
- 次の要件を満たすカラム
 - カラムで CAST が行われない
 - **SELECT** 句のベーステーブルのベースカラムである
 - **SELECT** 句のカラムを参照する式や関数が存在しない。また、select リスト内でカラムが重複しない (SELECT c1, c1 など)
 - **FOR UPDATE OF column-name-list** 句が指定された場合、ベーステーブルのベースカラムは、この句にリストされたカラムに制限される

SAP Sybase IQ では、結果セットのローとベーステーブルのローの、1 対 1 の関係を妨げる演算子を含むクエリに対して、更新可能なカーソルを使用することはできません。具体的には、以下が挙げられます。

- **SELECT DISTINCT**
- **UNION** が指定された演算子
- **GROUP BY** が指定された演算子
- **SET** 関数が指定された演算子
- **OLAP** 関数が定義された演算子。ただし、**RANK()** を除く

カーソルの結果セット内のローを更新する **SET** 句に指定可能なカラムと式については、「UPDATE (位置付け) 文 [ESQL] [SP]」の説明を参照してください。

SAP Sybase IQ で更新可能なカーソルへの挿入がサポートされるのは、NULL が扱えないカラムと IDENTITY でないカラムの両方がすべて選択され、かつ更新可能である場合だけです。

SAP Sybase IQ では、**COMMIT** と **ROLLBACK** を、開いている更新可能なカーソルの内部で実行することはできません。これは、カーソルがホールドカーソルとして

開かれている場合も同じです。SAP Sybase IQ は、更新可能なカーソル内部での **ROLLBACK TO SAVEPOINT** をサポートします。

カーソルを開いた後でエラーが発生すると、このカーソルをとおして実行されたすべての処理がロールバックされます。

更新可能なカーソルの制限

次のケースでは、宣言されたカーソルは読み込み専用となり、更新可能になりません。

- パス名に TEMP_EXTRACT_NAME1 オプションが設定され、データ抽出機能が有効になっている
- ANSI_CLOSE_CURSORS_ON_ROLLBACK が OFF に設定されている
- CHAINED が OFF に設定されている
- 文が **INSERT SELECT** または **SELECT INTO** である
- 複数のテーブルが含まれる
- 更新可能なカラムが存在しない

SAP Sybase IQ が更新可能なカーソルを要求されて設定に失敗した場合は、.iqmsg ファイルを参照して関連情報を調べてください。

更新可能カーソルと ODBC に関しては、制限事項があります。次の ODBC 関数を使用した場合、一度に更新、削除、または挿入できるローまたはレコードの数は、最大で 65535 です。

- **SQLSetPos SQL_UPDATE、SQL_DELETE、SQL_ADD**
- **SQLBulkOperations SQL_ADD、SQL_UPDATE_BY_BOOKMARK、SQL_DELETE_BY_BOOKMARK**

影響されるロー数を制御する文の属性の最大値が UNSIGNED SMALL INT の最大値 (65535) に制限されるという、実装固有の制限事項があります。

```
SQLSetStmtAttr(HANDLE, SQL_ATTR_ROW_ARRAY_SIZE, VALUE, 0)
```

SAP Sybase IQ の更新可能カーソルには、ANSI SQL3 標準の動作と比較して次の相違点があります。

- ホールドカーソルの更新がコミット時にクローズする。
- SAP Sybase IQ はカーソルが開いているときにテーブルをロックする。
- 更新、削除、挿入の処理はすべて、カーソルが閉じたときに、削除、更新、挿入の順序で適用される。

注意： サーバ上で現在オープンしているカーソルについての詳細情報を表示するには、**sp_iqcursorinfo** システムプロシージャを使用します。

標準

(先頭に戻る) (299 ページ)

- SQL — ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Open Client/Open Server でサポートされています。

パーミッション

(先頭に戻る) (299 ページ)

なし

参照：

- CALL 文 (115 ページ)
- DELETE (位置付け) 文 [ESQL] [SP] (312 ページ)
- FETCH 文 [ESQL] [SP] (357 ページ)
- OPEN 文 [ESQL] [SP] (461 ページ)
- PREPARE 文 [ESQL] (469 ページ)
- SELECT 文 (523 ページ)
- UPDATE (位置付け) 文 [ESQL] [SP] (571 ページ)

DECLARE CURSOR 文 [T-SQL]

Adaptive Server と互換性があるカーソルを宣言します。

クイックリンク：

「使用法」 (305 ページ)

「標準」 (306 ページ)

「パーミッション」 (306 ページ)

構文

```
DECLARE cursor-name  
... CURSOR FOR select-statement  
... [ FOR { READ ONLY | UPDATE } ]
```

使用法

(先頭に戻る) (305 ページ)

SAP Sybase IQ では、Adaptive Server でサポートされていない **DECLARE CURSOR** 構文がサポートされます。**DECLARE CURSOR** 構文の詳細については、「**DECLARE CURSOR** 文 [ESQL] [SP]」を参照してください。

注意： サーバ上で現在オープンしているカーソルについての詳細情報を表示するには、**sp_iqcursorinfo** システムプロシージャを使用します。

標準

(先頭に戻る) (305 ページ)

- SQL — FOR UPDATE オプションと FOR READ ONLY オプションは、ISO/ANSI SQL 文法の Transact-SQL 拡張機能です。
- SAP Sybase Database 製品 - Adaptive Server の **DECLARE CURSOR** 文の機能の中には、SAP Sybase IQ でサポートされていないものがいくつかあります。
 - SAP Sybase IQ 言語では、**DECLARE CURSOR** をプロシージャまたはバッチ内の BEGIN キーワードの直後に置く必要があります。Transact-SQL 言語では、このような制限はありません。
 - Adaptive Server では、カーソルがプロシージャまたはバッチ内で宣言されている場合、そのプロシージャまたはバッチの存続期間だけ存在します。SAP Sybase IQ では、カーソルを複合文内で宣言した場合、(SAP Sybase IQ と Transact-SQL のどちらの複合文で宣言されているかにかかわらず) カーソルは複合文の存続期間だけ存在します。

パーミッション

(先頭に戻る) (305 ページ)

なし

参照：

- **DECLARE CURSOR** 文 [ESQL] [SP] (299 ページ)

DECLARE LOCAL TEMPORARY TABLE 文

ローカルテンポラリテーブルを宣言します。

クイックリンク：

「例」 (307 ページ)

「使用法」 (307 ページ)

「標準」 (309 ページ)

「パーミッション」 (309 ページ)

構文

```
DECLARE LOCAL TEMPORARY TABLE table-name
... ( column-definition [ column-constraint ] ...
[ , column-definition [ column-constraint ] ... ]
[ , table-constraint ] ... )
...[ ON COMMIT { DELETE | PRESERVE } ROWS
]
```

例

(先頭に戻る) (306 ページ)

- **例 1** – Embedded SQL でのローカルテンポラリテーブルの宣言方法を示します。

```
EXEC SQL DECLARE LOCAL TEMPORARY TABLE MyTable (
    number INT
);
```

- **例 2** – ストアドプロシージャでのローカルテンポラリテーブルの宣言方法を示します。

```
BEGIN
    DECLARE LOCAL TEMPORARY TABLE TempTab (
        number INT
    );
    ...
END
```

使用法

(先頭に戻る) (306 ページ)

DECLARE LOCAL TEMPORARY TABLE はテンポラリテーブルを宣言します。

ローカルテンポラリテーブルとそのローは、テーブルを作成し、ローを挿入した接続からしか見えません。デフォルトで、テンポラリテーブルのローは **COMMIT** のときに削除されます。

複合文内で宣言したローカルテンポラリテーブルは、複合文内に存在します。それ以外の場合、宣言したローカルテンポラリテーブルは接続の終了時まで存在します。

column-definition、*column-constraint*、*table-constraint* の各構文の定義については、「CREATE TABLE 文」を参照してください。テンポラリテーブルへのデータの選択方法の例については、「SELECT 文」を参照してください。

ローカルテンポラリテーブルを作成すると、それが明示的であるかどうかにかかわらず、テンポラリテーブルが存在している間は別のテンポラリテーブルと同じ

名前で作成できません。たとえば、次のように入力すると、ローカルテンポラリテーブルが暗黙的に作成されます。

```
select * into #tmp from table1
```

または、次のように宣言して、ローカルテンポラリテーブルを明示的に作成することもできます。

```
declare local temporary table foo
```

その後、#tmp または foo に対して **SELECT INTO** を実行するか、#tmp または foo をもう一度宣言すると、#tmp または foo がすでに存在することを示すエラーが表示されます。

ローカルテンポラリテーブルを宣言するときは、所有者を指定しないでください。同じセッション内の複数の **DECLARE LOCAL TEMPORARY TABLE** 文で同じ *owner.table* を指定すると、構文エラーがレポートされます。たとえば、次の文を同じセッション内で実行すると、エラーがレポートされます。

```
DECLARE LOCAL TEMPORARY TABLE user1.temp(coll int);
DECLARE LOCAL TEMPORARY TABLE user1.temp(coll int);
```

所有者名を省略した場合は、エラー「アイテム 'temp' はすでに存在しています。」がレポートされます。

```
DECLARE LOCAL TEMPORARY TABLE temp(coll int);
DECLARE LOCAL TEMPORARY TABLE temp(coll int);
```

接続に同じ名前のローカルテンポラリテーブルがある場合、ベーステーブルまたはグローバルテンポラリテーブルを作成しようとしても失敗します。これは、新しいテーブルを *owner.table* がユニークに識別できないからです。

ただし、既存のベーステーブルまたはグローバルテンポラリテーブルとしてなら、ローカルテンポラリテーブルを同じ名前で作成できます。テーブル名への参照は、ローカルテンポラリテーブルにアクセスします。ローカルテンポラリテーブルが最初に解決されるからです。

次のシーケンス例を見てみましょう。

```
CREATE TABLE t1 (c1 int);
INSERT t1 VALUES (9);

DECLARE LOCAL TEMPORARY TABLE t1 (c1 int);
INSERT t1 VALUES (8);

SELECT * FROM t1;
```

返される結果は 8 です。ローカルテンポラリテーブルが接続により削除されるまで、t1 に対する参照はいずれも、ローカルテンポラリテーブル t1 を参照します。

ローカルテンポラリテーブルに対して、**ALTER TABLE** 文や **DROP INDEX** 文を使用することはできません。

ローカルテンポラリテーブルに対して、**sp_iqindex**、**sp_iqtablesizer**、**sp_iqindexsize** のストアードプロシージャを使用することはできません。

標準

(先頭に戻る) (306 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Adaptive Server では **DECLARE TEMPORARY TABLE** はサポートされていません。

パーミッション

(先頭に戻る) (306 ページ)

なし

参照：

- CREATE TABLE 文 (257 ページ)
- SELECT 文 (523 ページ)

DELETE 文

指定したテーブルから、検索条件を満たすすべてのローを削除します。WHERE 句を指定しない場合、指定したテーブルからすべてのローが削除されます。

クイックリンク：

「パラメータ」へ移動 (310 ページ)

「例」 (311 ページ)

「使用法」 (311 ページ)

「標準」 (311 ページ)

「パーミッション」 (312 ページ)

構文

DELETE

```
[ FROM ] [ owner. ] table-name [[AS correlation-name]
...[ FROM table-expression ]
[ WHERE search-condition ]]
```

table-expression

```
table-spec
```

```
| table-expression join-type table-spec [ ON condition ]
| table-expression, ...
```

パラメータ

(先頭に戻る) (309 ページ)

- **FROM 句** – ローが削除されるテーブルを示します。**DELETE** 文にオプションで第 2 FROM 句を指定すると、他のテーブルとのジョインに基づいて指定されたテーブルから削除されるローが決定されます。第 2 FROM 句がある場合、WHERE 句によってこの第 2 FROM 句のローの条件が指定されます。ローは、最初の FROM 句内で指定したテーブル名から削除されます。

注意： ジョイン仮想テーブルに対して **DELETE** 文を使用することはできません。ジョイン仮想テーブルから削除しようとする、エラーがレポートされません。

どちらの FROM 句にも関連名が使用されていない場合、**DELETE** 文のテーブル名には潜在的なあいまいさが存在します。次の例を考えます。

```
DELETE
FROM table_1
FROM table_1 AS alias_1, table_2 AS alias_2
WHERE ...
```

table_1 は、最初の FROM 句の中では関連名を使用せずに識別され、2 番目の FROM 句内では関連名を使用して識別されます。2 番目の FROM 句の table_1 に関連名を使用することで、文内に存在する table_1 のインスタンスが 1 つだけになります。これは、同じ文の中で関連名を使用する方法と使用しない方法の両方を使用して同じテーブルを識別する場合には、テーブルの 2 つのインスタンスが考えられるという、一般規則の例外です。

次の例を考えます。

```
DELETE
FROM table_1
FROM table_1 AS alias_1, table_1 AS alias_2
WHERE ...
```

2 番目の FROM 句に table_1 のインスタンスが 2 つあります。どちらのインスタンスで最初の FROM 句を識別するかがわからないため、関連名の一般的な規則では、最初の FROM 句内の table_1 は、2 番目の句内の table_1 のどちらのインスタンスでも識別されません。この文の中には table_1 のインスタンスが 3 つあります。

- **WHERE 句** – 指定した場合、検索条件を満たすローのみが削除されます。WHERE 句を指定しない場合、すべてのローが更新されます。

例

(先頭に戻る) (309 ページ)

- **例 1** – データベースから従業員番号 105 の従業員を削除します。

```
DELETE
FROM Employees
WHERE EmployeeID = 105
```

- **例 2** – FinancialData テーブルから、1993 年より前のデータをすべて削除します。

```
DELETE
FROM FinancialData
WHERE Year < 1993
```

- **例 3** – Customers テーブルにすでに名前がある場合、その名前をすべて Contacts テーブルから削除します。

```
DELETE
FROM Contacts
FROM Contacts, Customers
WHERE Contacts.Surname = Customers.Surname
AND Contacts.GivenName = Customers.GivenName
```

使用法

(先頭に戻る) (309 ページ)

ビューを定義する **SELECT** 文の **FROM** 句内にテーブルが 1 つだけ存在し、**GROUP BY** 句と集合関数が含まれていない、または **UNION** 操作が関係しない場合は、ビュー上で **DELETE** 文を使用できます。

ローを削除しようとしているテーブルと同じテーブル名が両方の **FROM** 句で使用されている場合、以下のいずれかに該当すると、同じテーブルを参照しているとみなされます。

- どちらのテーブル参照も、ユーザ ID を指定して修飾されていない
- 両方のテーブル参照が、ユーザ ID を指定して修飾されている
- 両方のテーブル参照が、相関名で指定されている

それらのテーブル参照が同じかどうかをサーバで判別できない場合、エラーが表示されます。これにより、意図しないローの削除による意図しないセマンティックを防ぎます。

標準

(先頭に戻る) (309 ページ)

SQL 文

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - ベンダ拡張も含めて Adaptive Server でサポートされています。

パーミッション

(先頭に戻る) (309 ページ)

そのテーブルに対する DELETE 権限が必要です。

参照：

- SELECT クエリブロック内のロー制限句 (5 ページ)
- FROM 句 (372 ページ)
- INSERT 文 (412 ページ)
- TRUNCATE 文 (561 ページ)

DELETE (位置付け) 文 [ESQL] [SP]

カーソルの現在の位置にあるデータを削除します。

クイックリンク：

「パラメータ」 (312 ページ)

「例」 (313 ページ)

「使用法」 (313 ページ)

「標準」 (313 ページ)

「パーミッション」 (314 ページ)

構文

```
DELETE [ FROM table-spec ]  
      WHERE CURRENT OF cursor-name
```

table-spec - (構文に戻る)
[owner.] correlation-name

cursor-name - (構文に戻る)
identifier | hostvar

パラメータ

(先頭に戻る) (312 ページ)

- **FROM** – ローが削除されるテーブルは次のように決定されます。
 - **FROM** 句が含まれていない場合、カーソルは 1 つのテーブルのみに配置できます。
 - カーソルがジョインクエリ用の場合は (ジョインがあるビューの使用を含む)、**FROM** 句を使用する必要があります。指定したテーブルの現在のローだけが削除されます。ジョインに含まれた他のテーブルは影響を受けません。
 - **FROM** 句を含め、テーブル所有者を指定しない場合は、最初に **table-spec** がすべての相関名に対して照合されます。
 - 相関名がある場合、**table-spec** は相関名で識別されます。
 - 相関名がない場合、**table-spec** はカーソルのテーブル名として明確に識別できるようにする必要があります。
 - **FROM** 句が含まれ、テーブル所有者が指定されている場合、**table-spec** はカーソルのテーブル名として明確に識別できるようにする必要があります。

例

(先頭に戻る) (312 ページ)

- **例 1** – データベースから現在のローを削除します。

```
DELETE WHERE CURRENT OF cur_employee
```

使用法

(先頭に戻る) (312 ページ)

この形式の **DELETE** 文は、指定されたカーソルの現在のローを削除します。現在のローとは、直前にカーソルからフェッチされたローのことです。

位置付け **DELETE** 文はビューでカーソルを開くときに使用できます。ただし、ビューが更新可能である場合にかぎられます。

位置付け **DELETE** 文による変更は、カーソル結果セット内で確認できます。ただし、クライアントサイドのキャッシュによって、これらの変更の確認が妨げられた場合を除きます。

標準

(先頭に戻る) (312 ページ)

- **SQL — ANSI_UPDATE_CONSTRAINTS** オプションが **OFF** に設定されている場合、更新可能なカーソルの範囲に ISO/ANSI SQL 文法のベンダ拡張が含まれる可能性があります。

- SAP Sybase Database 製品 - Embedded SQL の使用は、Open Client/Open Server でサポートされています。プロシージャとトリガの使用は、SQL Anywhere でサポートされています。

パーミッション

(先頭に戻る) (312 ページ)

カーソルで使用するテーブルに対する DELETE 権限が必要です。

参照：

- DECLARE CURSOR 文 [ESQL] [SP] (299 ページ)
- INSERT 文 (412 ページ)
- UPDATE 文 (566 ページ)
- UPDATE (位置付け) 文 [ESQL] [SP] (571 ページ)

DESCRIBE 文 [ESQL]

データベースから取り出したデータを格納するために必要なホスト変数、またはデータベースにデータを渡すために使用されるホスト変数に関する情報を取得します。

クイックリンク：

「パラメータ」 (315 ページ)

「例」 (317 ページ)

「使用法」 (317 ページ)

「標準」 (317 ページ)

「パーミッション」 (317 ページ)

構文

DESCRIBE

```
...[ USER TYPES ]
...[ { ALL | BIND VARIABLES FOR | INPUT
  | OUTPUT | SELECT LIST FOR } ]
...[ { LONG NAMES [ long-name-spec ] | WITH VARIABLE RESULT } ]
...[ FOR ] { statement-name | CURSOR cursor-name }
...INTO sqllda-name
```

long-name-spec

```
{ OWNER.TABLE.COLUMN
| TABLE.COLUMN
| COLUMN }
```

パラメータ

(先頭に戻る) (314 ページ)

- **statement-name** – 識別子またはホスト変数。文名を指定する場合、同じ文名で **PREPARE** 文を使用して、文を事前に作成しておく必要があります。また、事前に **SQLDA** を割り当てておく必要もあります (「**ALLOCATE DESCRIPTOR** 文 [ESQL]」を参照してください)。
- **cursor-name** – 宣言されたカーソル。事前にカーソルを宣言して開いておく必要があります。デフォルトの動作は、**OUTPUT** です。**OUTPUT** が存在するのは、**SELECT** 文と **CALL** 文だけです。他の文または動的カーソルではないカーソルの **DESCRIBE OUTPUT** は、**SQLDA** の **sqld** フィールドを 0 に設定して出力なしを示します。
- **sqlda-name** – 識別子
- **USER TYPES** – カラムのユーザ定義データ型に関する情報を返します。通常、このような **DESCRIBE** は、以前の **DESCRIBE** が **DT_HAS_USERTYPE_INFO** のインジケータを返したときに行われます。

返される情報は、**USER TYPES** 句を付けない **DESCRIBE** の場合と同じですが、**sqlname** フィールドがカラム名の代わりにユーザ定義データ型の名前を保持する点が異なります。

DESCRIBE が **LONG NAMES** 句を使用する場合、**sqldata** フィールドがこの情報を保持します。

- **ALL** – データベースサーバに対して 1 回要求するだけで **INPUT** と **OUTPUT** を記述します。マルチユーザ環境では、これがパフォーマンスによい影響を与えます。**INPUT** 情報が最初に **SQLDA** に格納され、次に **OUTPUT** 情報が格納されます。**sqld** フィールドには、**INPUT** と **OUTPUT** 変数の総数が入ります。インジケータ変数内の **DT_DESCRIBE_INPUT** ビットは **INPUT** 変数で設定され、**OUTPUT** 変数でクリアされます。
- **BIND VARIABLES FOR** – **INPUT** 句と同じです。**INPUT** 句と一緒に使用した場合、**DESCRIBE BIND VARIABLES** は **SQLDA** でデータ型を設定しません。アプリケーションがデータ型を設定する必要があります。
- **SELECT LIST FOR** – **OUTPUT** 句と同じです。
- **INPUT** – **SQLDA** の名前フィールドにバインド変数名を入力します。バインド変数は、データベースが文を実行するときにアプリケーションによって提供される値です。バインド変数は、文に対するパラメータとみなすことができます。また、**INPUT** 句は、バインド変数の数を **SQLDA** の **sqld** フィールドに格納します。

DESCRIBE は、**SQLDA** 内のインジケータ変数を使用して情報を追加します。

DT_PROCEDURE_IN と **DT_PROCEDURE_OUT** は、**CALL** 文を記述するときに、

インジケータ変数の中に設定されるビットです。DT_PROCEDURE_IN は IN または INOUT パラメータを示し、DT_PROCEDURE_OUT は INOUT または OUT パラメータを示します。プロシージャの RESULT カラムは、両方のビットをクリアします。DESCRIBE OUTPUT の後、これらのビットは結果セットを持っている文 (OPEN、FETCH、RESUME、CLOSE を使用する必要がある) と持っていない文 (EXECUTE を使用する必要がある) を区別するのに使用できます。DESCRIBE INPUT は、バインド変数が CALL 文に対する引数であるときのみ、状況に応じた DT_PROCEDURE_IN と DT_PROCEDURE_OUT を設定します。CALL 文の引数である式内のバインド変数は、ビットを設定します。

- **OUTPUT** – それぞれの select リスト項目に対する SQLDA でのデータ型と長さを入力します。名前フィールドにも、select リスト項目の名前が入ります。select リスト項目に対してエイリアスが指定されている場合、名前はそのエイリアスになります。エイリアスが指定されていない場合、select リスト項目から派生する名前が付けられます。項目が単純なカラム名である場合は、その名前を使用します。そうでない場合は、式の部分文字列を使用します。また、**DESCRIBE** は、select リスト項目の数を SQLDA の sqld フィールドに格納します。
 - 記述されている文が 2 つ以上の **SELECT** 文の UNION である場合、DESCRIBE OUTPUT に対して返されるカラム名は、最初の **SELECT** 文に対して返されるカラム名と同じです。
 - CALL 文を記述すると、DESCRIBE OUTPUT によって、プロシージャ中の INOUT または OUT パラメータごとに、データ型、長さ、名前が SQLDA に格納されます。DESCRIBE OUTPUT 文は、SQLDA の sqld フィールドの中に INOUT または OUT パラメータの数も格納します。
 - 結果セットを返す CALL 文を記述すると、OUTPUT によってプロシージャ定義の RESULT カラムごとに、データ型、長さ、名前が SQLDA に格納されます。また、DESCRIBE OUTPUT は、結果カラムの数を SQLDA の sqld フィールドに格納します。
- **LONG NAMES** – 文またはカーソルに対応するカラム名を取得します。この句がなければ、カラム名の長さは 29 文字に制限されます。この句を指定すると、任意の長さの名前がサポートされます。LONG NAMES を使用した場合、カーソルからフェッチを行う場合と同様に、長い名前は SQLDA の SQLDATA フィールドに格納されます。それ以外のどのフィールド (SQLLEN、SQLTYPE など) も入力されません。SQLDA は **FETCH SQLDA** と同じように設定してください。つまり、各カラムには 1 つのエントリが含まれ、エントリは文字列データ型である必要があります。長い名前のデフォルトの仕様は、TABLE.COLUMN です。
- **WITH VARIABLE RESULT** – カラムの数とデータ型が異なる複数の結果セットを持つ可能性があるプロシージャを記述します。WITH VARIABLE RESULT を

使用する場合、記述後にデータベースサーバによって SQLCOUNT 値に次のいずれかの値が設定されます。

- **0** – 結果セットは変更される場合があります。各 **OPEN** 文の後でプロシージャコールを再度記述してください。
- **1** – 結果セットは固定です。再度記述する必要はありません。

例

(先頭に戻る) (314 ページ)

- **例 1 – DESCRIBE 文の使用法を示します。**

```
sqllda = alloc_sqllda( 3 );
EXEC SQL DESCRIBE OUTPUT
  FOR employee_statement
  INTO sqllda;
if( sqllda->sqlld > sqllda->sqln ) {
  actual_size = sqllda->sqlld;
  free_sqllda( sqllda );
  sqllda = alloc_sqllda( actual_size );
  EXEC SQL DESCRIBE OUTPUT
    FOR employee_statement
    INTO sqllda;
}
```

使用法

(先頭に戻る) (314 ページ)

DESCRIBE は、指定した **SQLDA** を設定し、指定した文に対して **OUTPUT (SELECT LIST と同じ)** または **INPUT (BIND VARIABLES)** を記述します。

標準

(先頭に戻る) (314 ページ)

- SQL 一部の句は、ISO/ANSI SQL 文法のベンダ拡張です。
- SAP Sybase Database 製品 - 一部の句は、Open Client/Open Server でサポートされています。

パーミッション

(先頭に戻る) (314 ページ)

なし

参照：

- **ALLOCATE DESCRIPTOR 文 [ESQL] (7 ページ)**

SQL 文

- DECLARE CURSOR 文 [ESQL] [SP] (299 ページ)
- OPEN 文 [ESQL] [SP] (461 ページ)
- PREPARE 文 [ESQL] (469 ページ)

DISCONNECT 文 [Interactive SQL]

データベースとの接続を削除します。

クイックリンク：

「パラメータ」 (318 ページ)

「例」 (318 ページ)

「使用法」 (318 ページ)

「標準」 (319 ページ)

「パーミッション」 (319 ページ)

構文

```
DISCONNECT [ { connection-name | CURRENT | ALL } ]
```

パラメータ

(先頭に戻る) (318 ページ)

- **connection-name** – 識別子、文字列、またはホスト変数
- **ALL** – すべてのデータベース環境へのアプリケーションの接続をすべて切断します。
- **CURRENT** – (デフォルト) 現在の接続を切断します。

例

(先頭に戻る) (318 ページ)

- **例 1** – Embedded SQL での **DISCONNECT** の使用法：

```
EXEC SQL DISCONNECT :conn_name
```

- **例 2** – **dbisql** から **DISCONNECT** を使用してすべての接続を切断する方法：

```
DISCONNECT ALL
```

使用法

(先頭に戻る) (318 ページ)

DISCONNECT 文は、データベースサーバとの接続を切断し、データベースサーバが使用しているすべてのリソースを解放します。切断対象の接続が **CONNECT** 文で指定されている場合、その名前を指定できます。

接続が切断されると、暗黙の **ROLLBACK** が実行されます。

標準

(先頭に戻る) (318 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - Open Client/Open Server でサポートされています。

パーミッション

(先頭に戻る) (318 ページ)

なし

参照：

- **CONNECT** 文 [ESQL] [Interactive SQL] (133 ページ)
- **SET CONNECTION** 文 [ESQL] [Interactive SQL] (539 ページ)

DROP 文

データベースからオブジェクトを削除します。

クイックリンク：

「パラメータ」 (320 ページ)

「例」 (321 ページ)

「使用法」 (321 ページ)

「標準」 (322 ページ)

「パーミッション」 (322 ページ)

構文

DROP

```
{ DBSPACE dbspace-name
| { DATATYPE [ IF EXISTS ]
| DOMAIN } datatype-name
| EVENT [ IF EXISTS ] event-name
| INDEX [ IF EXISTS ] [ [ owner ].table-name. ] index-name
| MESSAGE message-number
```

```

| TABLE [ IF EXISTS ] [ owner. ] table-name
| VIEW [ IF EXISTS ] [ owner. ] view-name
| MATERIALIZED VIEW [ IF EXISTS ] [ owner. ] view-name
| PROCEDURE [ IF EXISTS ] [ owner. ] procedure-name
| FUNCTION [ IF EXISTS ] [ owner. ] function-name }

```

パラメータ

(先頭に戻る) (319 ページ)

- **IF EXISTS** – この **DROP** 文が存在しないデータベースオブジェクトを削除しようとしたときに、エラーを返さないようにする場合に使用します。
- **INDEX** – 明示的に作成されたインデックスがあれば削除します。一意性制約または外部キー制約、または関連付けられたプライマリキーがない場合は、暗黙的に作成されたインデックスだけが削除されます。

DROP INDEX ユニークでない HG インデックスに、強制力のない外部キーが関連付けられているとエラーになります。

警告！ DBO ユーザが所有するビューを削除しないでください。このようなビューを削除したり、テーブルに変更したりすると、問題が起きる可能性があります。

DROP INDEX は、文が他の接続で現在使用中のテーブルに影響するときには処理されません。

- **TABLE – DROP TABLE** は、プライマリテーブルに外部キー制約 (強制力のない外部キー制約を含む) が関連付けられている場合は処理されません。

テーブルに **IDENTITY** カラムがあって **IDENTITY_INSERT** がこのテーブルに設定されている場合も、**DROP TABLE** は処理されません。そのテーブルを削除するには、**IDENTITY_INSERT** をクリア、すなわち空の文字列 ('') に設定するか、別のテーブル名に設定します。

外部キーには、ユニークでないシングルまたはマルチカラムの HG インデックスを設定できます。プライマリキーは、ユニークなシングルまたはマルチカラムの HG インデックスを設定できます。既存の外部キー制約、プライマリキー制約、一意性制約に対して暗黙的に作成された HG インデックスを削除することはできません。

初期の DB 領域は **SYSTEM**、**IQ_SYSTEM_MAIN**、**IQ_SYSTEM_TEMP**、**IQ_SYSTEM_MSG** の 4 つです。これらの初期の DB 領域を削除することはできませんが、複数の DB 領域を含んだ IQ メインストアまたはカタログストアからは DB 領域を削除できます。ただし、少なくとも 1 つの DB 領域が読み書きモードのままである必要があります。

DB 領域内のテーブルを削除しなければ、DB 領域を削除することはできません。DB 領域にユーザデータが残っていればエラーが返されます。これ以外の

構造は、DB 領域が削除される時自動的に移動されます。DB 領域を削除できるのは、DB 領域を読み込み専用指定した後だけです。

注意： コマンドによって使用された後の DB 領域には、コマンド使用後のどの時点のデータも含まれている可能性があるため、その DB 領域に対する **DROP DBSPACE** は処理されません。

DROP TABLE は、文が他の接続で現在使用中のテーブルに影響するときには処理されません。

- **PROCEDURE – DROP PROCEDURE** はプロシージャが別の接続で使用中の場合は処理されません。
- **DATATYPE – DROP DATATYPE** はデータ型がテーブルで使用されている場合、処理されません。データ型を削除するには、ユーザ定義データ型を持つすべてのカラムのデータ型を変更する必要があります。**DROP DATATYPE** ではなく **DROP DOMAIN** の使用をおすすめします。これは、**DROP DOMAIN** が ANSI/ISO SQL3 のドラフトで使用されている構文であるためです。

例

(先頭に戻る) (319 ページ)

- **例 1** – データベースから Departments テーブルを削除します。

```
DROP TABLE Departments
```

- **例 2** – データベースから emp_dept ビューを削除します。

```
DROP VIEW emp_dept
```

- **例 3** – 接続されているシンプレックスノードまたはマルチプレックスノードから myDAS メインキャッシュを削除します。

```
DROP DBSPACE myDAS
```

使用法

(先頭に戻る) (319 ページ)

DROP は指定したデータベース構造の定義を削除します。構造が DB 領域である場合、DB 領域を削除する前に、DB 領域内のデータを含むすべてのテーブルを削除または移動する必要があります。これ以外の構造の場合は自動的に移動されます。構造がテーブルである場合、テーブル内のデータは削除プロセスの一部として自動的に削除されます。同様に、テーブルのすべてのインデックスとキーも、**DROP TABLE** によって削除されます。

文が別の接続で現在使用されているテーブルに影響する場合、**DROP DBSPACE** は処理されません。別の接続で使用中のプロシージャも、**DROP PROCEDURE** では処理されません。

関連する動作

- オートコミット。**dbisql** の [データ] ウィンドウは閉じられます。**DROP TABLE** および **DROP INDEX** では、現在の接続のすべてのカーソルが閉じられます。
- ローカルテンポラリテーブルは例外です。削除されると、コミットは実行されません。

標準

(先頭に戻る) (319 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - Adaptive Server でサポートされています。

パーミッション

(先頭に戻る) (319 ページ)

DBSPACE 句 - **DROP ANY OBJECT** システム権限が必要です。ユーザが当該データベースに対して排他的な接続を保持している必要があります。

DOMAIN 句 - 次のいずれかが必要です。

- **DROP DATATYPE** システム権限
- **DROP ANY OBJECT** システム権限
- オブジェクトを所有している

FUNCTION 句 - 次のいずれかが必要です。

- **DROP ANY PROCEDURE** システム権限
- **DROP ANY OBJECT** システム権限
- 関数を所有している

INDEX 句 - 次のいずれかが必要です。

- **DROP ANY INDEX** システム権限
- **DROP ANY OBJECT** システム権限
- インデックス付きのテーブルに対する **REFERENCES** 権限
- インデックスの基本となるテーブルを所有している

DBA または適切な権限を持つユーザは、完全に修飾された名前を使用しなくても、別のユーザが所有するテーブルのインデックスを削除できます。DBA が所有

するベーステーブルのインデックスを削除する場合、他のユーザはすべて完全に修飾されたインデックス名を指定する必要があります。

MATERIALIZED VIEW 句 - 次のいずれかが必要です。

- DROP ANY MATERIALIZED VIEW システム権限
- DROP ANY OBJECT システム権限
- マテリアライズドビューを所有している

PROCEDURE 句 - 次のいずれかが必要です。

- DROP ANY PROCEDURE システム権限
- DROP ANY OBJECT システム権限
- プロシージャを所有している

TABLES 句 - 次のいずれかが必要です。

- DROP ANY TABLE システム権限
- DROP ANY OBJECT システム権限
- テーブルを所有している

このテンポラリテーブルを参照したすべてのユーザが切断されるまで、グローバルテンポラリテーブルは削除されません。

VIEW 句 - 次のいずれかが必要です。

- DROP ANY VIEW システム権限
- DROP ANY OBJECT システム権限
- ビューを所有している

それ以外のすべての句 - 次のいずれかが必要です。

- DROP ANY OBJECT システム権限
- オブジェクトを所有している

DROP AGENT 文

SAP Control Center の SAP Sybase IQ エージェントを削除します。

クイックリンク：

「使用法」 (324 ページ)

「パーミッション」 (324 ページ)

構文

DROP AGENT FOR MULTIPLEX SERVER *server-name*

適用対象

マルチプレックスのみ。

使用法

(先頭に戻る) (323 ページ)

DROP AGENT は、SAP Sybase IQ エージェントとサーバの関連付けを削除します。

SYS.ISYSIQMPXSERVERAGENT システムテーブルには、サーバのエージェント接続定義が格納されています。

関連する動作

- オートコミット

パーミッション

(先頭に戻る) (323 ページ)

MANAGE MULTIPLEX システム権限が必要です。

DROP CONNECTION 文

すべてのユーザのデータベースへの接続を削除します。

クイックリンク：

「パラメータ」 (324 ページ)

「例」 (325 ページ)

「使用法」 (325 ページ)

「標準」 (325 ページ)

「パーミッション」 (325 ページ)

構文

DROP CONNECTION *connection-id*

パラメータ

(先頭に戻る) (324 ページ)

- **connection-id – connection_property** 関数を使用し、接続番号を要求して取得します。次の文は、現在の接続の接続 ID を返します。

```
SELECT connection_property( 'number' )
```

例

(先頭に戻る) (324 ページ)

- **例 1** – ID 番号 4 の接続を削除します。

```
DROP CONNECTION 4
```

使用法

(先頭に戻る) (324 ページ)

現在の接続は削除できません。最初の接続を削除するには事前に別の接続を作成する必要があります。

標準

(先頭に戻る) (324 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Adaptive Server ではサポートされていません。

パーミッション

(先頭に戻る) (324 ページ)

DROP CONNECTION システム権限が必要です。

参照：

- CONNECT 文 [ESQL] [Interactive SQL] (133 ページ)

DROP DATABASE 文

データベースとそれに関連する DB 領域セグメントファイルを削除します。

クイックリンク：

「パラメータ」 (326 ページ)

「例」 (326 ページ)

「使用法」 (326 ページ)

SQL 文

「標準」 (327 ページ)

「パーミッション」 (327 ページ)

構文

```
DROP DATABASE db-filename [ KEY key-spec ]
```

パラメータ

(先頭に戻る) (325 ページ)

- **key-spec** – 大文字と小文字が混在する英数字、および特殊文字で構成される文字列。キーをコマンドシェルによる解読または変更から保護するために必要です。
- **db-filename** – **CREATE DATABASE** を使用してデータベースに定義したデータベースファイル名に対応しています。**CREATE DATABASE** コマンドで、この値にディレクトリパスを指定した場合は、**DROP DATABASE** でもディレクトリパスを指定する必要があります。指定しないと、SAP Sybase IQ は、サーバファイルが格納されているデフォルトディレクトリ内のデータベースファイルを探します。

例

(先頭に戻る) (325 ページ)

- **例 1** – データベース `mydb` を削除します。

```
DROP DATABASE 'mydb.db'
```

- **例 2** – 次の文では、暗号化されたデータベース `marvin.db` を削除します。このデータベースは、キー `is!seCret` を使用して作成されています。

```
DROP DATABASE 'marvin.db' KEY 'is!seCret'
```

- **例 3** – UNIX システムで、データベース `temp.db` を `/s1/temp` ディレクトリから削除します。

```
DROP DATABASE '/s1/temp/temp.db'
```

使用法

(先頭に戻る) (325 ページ)

DROP DATABASE は、カタログストアファイルを削除する前に、IQ ストアとテンポラリストアに関連付けられたすべてのデータベースセグメントファイルを削除します。

データベースは、停止してから削除してください。接続パラメータ `AUTOSTOP=no` が使用されている場合、**STOP DATABASE** 文の実行が必要になる場合があります。

DatabaseStart イベントが定義された IQ データベースを、**DROP DATABASE** 文を実行して削除することはできません。

標準

(先頭に戻る) (325 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Adaptive Server ではサポートされていません。

パーミッション

(先頭に戻る) (325 ページ)

この文を実行するのに必要なパーミッションは、以下のように、**-gu** サーバコマンドラインオプションを使用して設定します。

- **NONE** - この文を発行できるユーザはいません。
- **DBA - SERVER OPERATOR** システム権限が必要です。
- **UTILITY_DB - utility_db** データベースに接続できるユーザのみがこの文を発行できます。

参照：

- CREATE DATABASE 文 (137 ページ)
- STOP DATABASE 文 [Interactive SQL] (556 ページ)

DROP EXTERNLOGIN 文

SAP Sybase IQ システムテーブルから外部ログインを削除します。

クイックリンク：

「パラメータ」 (328 ページ)

「例」 (328 ページ)

「使用法」 (328 ページ)

「標準」 (328 ページ)

「パーミッション」 (328 ページ)

構文

```
DROP EXTERNLOGIN login-name  
TO remote-server
```

パラメータ

(先頭に戻る) (327 ページ)

- **login-name** – ローカルユーザログイン名を指定します。
- **TO** – リモートサーバ名を指定します。このサーバのローカルユーザの代替ログイン名とパスワードが、削除される外部ログインです。

例

(先頭に戻る) (327 ページ)

- **例 1** – リモートデータベース `sybase1` からログイン `dba` を削除します。

```
DROP EXTERNLOGIN dba TO sybase1
```

使用法

(先頭に戻る) (327 ページ)

DROP EXTERNLOGIN による変更は、次にリモートサーバに接続するまで有効になりません。

関連する動作：

- オートコミット。

標準

(先頭に戻る) (327 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - Open Client/Open Server でサポートされています。

パーミッション

(先頭に戻る) (327 ページ)

MANAGE ANY USER システム権限が必要です。

参照：

- CREATE EXTERNLOGIN 文 (166 ページ)

DROP LDAP SERVER 文

指定された LDAP サーバ設定オブジェクトが **READY** または **ACTIVE** 状態ではないことを検証した後に、**SYSLDAPSERVER** システムビューから削除します。

クイックリンク：

「パラメータ」 (329 ページ)

「例」 (329 ページ)

「使用法」 (330 ページ)

「標準」 (330 ページ)

「パーミッション」 (330 ページ)

構文

```
DROP LDAP SERVER ldapua-server-name  
[ WITH DROP ALL REFERENCES ] [ WITH SUSPEND ]
```

パラメータ

(先頭に戻る) (329 ページ)

- **WITH DROP ALL REFERENCES** – ログインポリシーに参照がある LDAP サーバ設定オブジェクトをサービスから削除します。
- **WITH SUSPEND** – **READY** 状態または **ACTIVE** 状態の LDAP サーバ設定オブジェクトでも削除を可能にします。

例

(先頭に戻る) (329 ページ)

- **例 1** – LDAP サーバ設定オブジェクトへの参照がすべてのログインポリシーから削除されている場合、次の 2 つのコマンドセットは同じになります。**WITH DROP ALL REFERENCES** パラメータと **WITH SUSPEND** パラメータを使用すると、**ALTER LDAP SERVER** 文の後に **DROP LDAP SERVER** 文を実行する必要がなくなります。

```
DROP LDAP SERVER ldapserver1 WITH DROP ALL REFERENCES WITH SUSPEND
```

この文は次の文と同義です。

```
ALTER LDAP SERVER ldapserver1 WITH SUSPEND DROP LDAP SERVER  
ldapserver1 WITH DROP ALL REFERENCES
```

使用法

(先頭に戻る) (329 ページ)

READY 状態または ACTIVE 状態の LDAP サーバ設定オブジェクトに対して **DROP LDAP SERVER** 文を発行すると、失敗します。これにより、アクティブな LDAP サーバ設定オブジェクトが偶発的に削除されないようにできます。また、LDAP サーバ設定オブジェクトを参照するログインポリシーが存在する場合も、**DROP LDAP SERVER** 文は失敗します。

標準

(先頭に戻る) (329 ページ)

ANSI SQL – 準拠レベル：Transact-SQL 拡張。

パーミッション

(先頭に戻る) (329 ページ)

MANAGE ANY LDAP SERVER システム権限が必要です。

DROP LOGIN POLICY 文

ログインポリシーをデータベースから削除します。

クイックリンク：

「例」 (330 ページ)

「使用法」 (331 ページ)

「パーミッション」 (331 ページ)

構文

```
DROP LOGIN POLICY policy-name
```

例

(先頭に戻る) (330 ページ)

- **例 1** – Test11 ログインポリシーを作成してから削除します。

```
CREATE LOGIN POLICY Test11;  
DROP LOGIN POLICY Test11 ;
```

使用法

(先頭に戻る) (330 ページ)

DROP LOGIN POLICY 文は、ユーザに割り当てられたポリシーを削除しようとする
と失敗します。**ALTER USER** 文を使用してユーザのポリシー割り当てを変更する
か、**DROP USER** を使用してユーザを削除できます。

パーミッション

(先頭に戻る) (330 ページ)

MANAGE ANY LOGIN POLICY システム権限が必要です。

参照：

- ALTER USER 文 (92 ページ)
- CREATE LOGIN POLICY 文 (194 ページ)
- DROP USER 文 (346 ページ)
- ALTER LOGIN POLICY 文 (32 ページ)

DROP LOGICAL SERVER 文

ユーザ定義の論理サーバを削除します。次の文では、論理サーバで共有される物
理ノード間で、一貫した共有システムテナポラリストア設定が強制的に適用され
ます。

クイックリンク：

「パラメータ」 (331 ページ)

「例」 (332 ページ)

「使用法」 (332 ページ)

「パーミッション」 (332 ページ)

構文

```
DROP LOGICAL SERVER logical-server-name  
[ WITH STOP SERVER ]
```

パラメータ

(先頭に戻る) (331 ページ)

- **WITH STOP SERVER** -TEMP_DATA_IN_SHARED_TEMP オプションが直接的または間接的に変更された場合、論理サーバ内のすべてのサーバを自動的に停止します。

適用対象

マルチプレックスのみ。

例

(先頭に戻る) (331 ページ)

- **例 1** - ユーザ定義の論理サーバ `ls1` を削除します。

```
DROP LOGICAL SERVER ls1
```

使用法

(先頭に戻る) (331 ページ)

SAP Sybase IQ は、論理サーバの削除時に、次のカタログ変更を内部で実行します。

- 論理サーバのすべてのメンバーシップ定義を削除します。
- 削除対象の論理サーバへの明示的な割り当てが存在するログインポリシーのそれぞれから論理サーバの割り当てを削除します。削除対象の論理サーバがログインポリシーに割り当てられている唯一の論理サーバである場合、SAP Sybase IQ によってそのログインポリシーの論理サーバ割り当てが `NONE` に設定されます。
- 論理サーバエントリを `ISYSIQ.LOGICALSERVER` から削除します。

パーミッション

(先頭に戻る) (331 ページ)

`MANAGE MULTIPLEX` システム権限が必要です。

DROP LS POLICY 文

マルチプレックスから論理サーバポリシーを削除します。

クイックリンク：

「パラメータ」 (333 ページ)

「例」 (333 ページ)

「標準」 (333 ページ)

「パーミッション」 (333 ページ)

構文

```
DROP LS POLICY ls-policy-name
```

パラメータ

(先頭に戻る) (332 ページ)

- **ls-policy-name** – ROOT を除く任意のポリシー名で、いずれの論理サーバでも現在使用されていないポリシーを参照する必要があります。

適用対象

マルチプレックス。

例

(先頭に戻る) (332 ページ)

- **例 1** – Test20 論理サーバポリシーを削除します。

```
DROP LS POLICY Test20
```

標準

(先頭に戻る) (332 ページ)

ANSI SQL – 準拠レベル：Transact-SQL 拡張。

パーミッション

(先頭に戻る) (332 ページ)

MANAGE MULTIPLEX システム権限が必要です。

DROP MULTIPLEX SERVER 文

マルチプレックスからサーバを削除します。

クイックリンク：

「パラメータ」 (334 ページ)

「例」 (334 ページ)

「使用法」 (334 ページ)

「パーミッション」 (335 ページ)

構文

```
DROP MULTIPLEX SERVER {server-name} [drop_mpx_server_clause]
```

```
drop_mpx_server_clause
```

```
{ WITH DROP MEMBERSHIP | WITH DROP LOGICAL SERVER }
```

パラメータ

(先頭に戻る) (333 ページ)

- **WITH DROP MEMBERSHIP** – 削除するマルチプレックスサーバに、1つ以上の論理サーバメンバーシップが存在する場合は、エラーが発生して失敗します。マルチプレックスサーバをそのすべてのメンバーシップとともに削除するには、WITH DROP MEMBERSHIP 句を使用します。
- **WITH DROP LOGICAL SERVER** – 最後のセカンダリサーバがすべてのユーザ定義論理サーバとともに削除されます。最後のセカンダリサーバを削除する際に、1つ以上のユーザ定義の論理サーバが存在していると、**DROP MULTIPLEX SERVER** コマンドは失敗します。

注意： WITH DROP LOGICAL SERVER 句は、最後のセカンダリサーバを削除する場合のみ有効です。他の場合は、エラーが報告されます。

適用対象

マルチプレックスのみ。

例

(先頭に戻る) (333 ページ)

- **例 1** – *writer1* という名前のマルチプレックスサーバを削除します。

```
DROP MULTIPLEX SERVER writer1
```

使用法

(先頭に戻る) (333 ページ)

削除前に各マルチプレックスサーバを停止します。この文は自動的にコミットされます。

削除されるサーバが、推奨されているように停止されていない場合は、この文の実行後に自動的に停止します。

最後のセカンダリサーバを削除すると、マルチプレックスはシンプレックスに変換されます。マルチプレックス内の最後のセカンダリサーバの削除後に、コオーディネータは自動的に停止します。必要に応じて再起動する必要があります。

パーミッション

(先頭に戻る) (333 ページ)

MANAGE MULTIPLEX システム権限が必要です。

DROP ROLE 文

データベースからユーザ定義ロールを削除したり、ユーザ拡張ロールを通常のユーザに変換したりします。

クイックリンク：

「パラメータ」 (335 ページ)

「例」 (336 ページ)

「使用法」 (336 ページ)

「標準」 (336 ページ)

「パーミッション」 (336 ページ)

構文

```
DROP ROLE [ FROM USER ] role_name  
[ WITH REVOKE ]
```

パラメータ

(先頭に戻る) (335 ページ)

- **role_name** – データベースに既存のロールの名前である必要があります。
- **FROM USER** – データベースから削除するのではなく、ユーザ拡張ロールを通常のユーザに戻す際に必要です。role_name は、データベースに存在している必要があります。

このユーザは、ユーザ拡張ロールに付与されていたログイン権限、システム権限、およびロールをそのまま保持し、ユーザ拡張ロールが所有していたオブジェクトの所有者になります。ユーザ拡張に付与されていたユーザはただちに取り消されます。

- **WITH REVOKE** – ユーザがロールの基礎となるシステム権限を付与されたスタンドアロンロールまたはユーザ拡張ロールを削除するときに必要です。WITH ADMIN OPTION 句と WITH NO ADMIN OPTION 句のどちらかを使用して付与できます。

例

(先頭に戻る) (335 ページ)

- **例 1** – 他のユーザまたはロールには付与されていない Joe という名前のユーザ拡張ロールを通常のユーザに戻します。

```
DROP ROLE FROM USER Joe
```

- **例 2** – 他のユーザまたはロールには付与されていない Jack という名前のユーザ拡張ロールをデータベースから削除します。

```
DROP ROLE Jack
```

- **例 3** – 他のユーザまたはロールに付与されている Sam という名前のユーザ拡張ロールを通常のロールに戻します。

```
DROP ROLE FROM USER Sam  
WITH REVOKE
```

- **例 4** – 他のユーザまたはロールに付与されている Sales2 という名前のスタンダードロールをデータベースから削除します。

```
DROP ROLE Sales2  
WITH REVOKE
```

使用法

(先頭に戻る) (335 ページ)

ユーザ定義ロールは、残される依存ロールのすべてが、有効なパスワードを持つ管理ユーザの最少数を満たしていれば、データベースから削除することも、通常のユーザに戻すこともできます。

標準

(先頭に戻る) (335 ページ)

ANSI SQL – 準拠レベル： Transact-SQL 拡張。

パーミッション

(先頭に戻る) (335 ページ)

- 削除するロールに対する管理権限が必要です。
- 削除するロールがオブジェクトを所有している場合、DROP 文の実行時に、オブジェクトがセッションおよびユーザで使用されていない場合があります。

DROP SEQUENCE 文

シーケンスを削除します。この文は、SAP Sybase IQ のカタログストアテーブルのみに適用されます。

構文

```
DROP SEQUENCE [ owner. ] sequence-name
```

備考

指定したシーケンスが見つからない場合は、エラーメッセージが返されます。シーケンスを削除すると、そのシーケンス名のすべての同義語がデータベースサーバによって自動的に削除されます。

権限

シーケンスの所有者であるか、DROP ANY SEQUENCE または DROP ANY OBJECT システム権限を持っている必要があります。

関連する動作

なし

標準と互換性

- **SQL/2008** – シーケンスは、SQL/2008 のオプションの言語機能 T176 を構成します。

例

次の例では、Test という名前のシーケンスを作成した後に削除します。

```
CREATE SEQUENCE Test
START WITH 4
INCREMENT BY 2
NO MAXVALUE
NO CYCLE
CACHE 15;
DROP SEQUENCE Test;
```

DROP SERVER 文

SAP Sybase IQ システムテーブルからリモートサーバを削除します。

クイックリンク：

「例」 (338 ページ)

「使用法」 (338 ページ)

SQL 文

「標準」 (338 ページ)

「パーミッション」 (338 ページ)

構文

```
DROP SERVER server-name
```

例

(先頭に戻る) (337 ページ)

- **例 1** – 次の例では、サーバ IQ_prod を削除します。

```
DROP SERVER iq_prod
```

使用法

(先頭に戻る) (337 ページ)

DROP SERVER を正常に実行するには、リモートサーバに定義されたプロキシテーブルをすべて削除しておく必要があります。

関連する動作

- オートコミット。

標準

(先頭に戻る) (337 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - Open Client/Open Server でサポートされています。

パーミッション

(先頭に戻る) (337 ページ)

SERVER OPERATOR システム権限が必要です。

参照：

- CREATE SERVER 文 (241 ページ)

DROP SERVICE 文

Web サービスを削除します。

クイックリンク：

「例」 (339 ページ)

「使用法」 (339 ページ)

「標準」 (339 ページ)

「パーミッション」 (339 ページ)

構文

```
DROP SERVICE service-name
```

例

(先頭に戻る) (338 ページ)

- **例 1** - tables という名前の Web サービスを削除するには、次の文を実行します。

```
DROP SERVICE tables
```

使用法

(先頭に戻る) (338 ページ)

DROP SERVICE は Web サービスを削除します。

標準

(先頭に戻る) (338 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - Adaptive Server ではサポートされていません。

パーミッション

(先頭に戻る) (338 ページ)

MANAGE ANY WEB SERVICE システム権限が必要です。

参照：

- ALTER SERVICE 文 (57 ページ)
- CREATE SERVICE 文 (244 ページ)

DROP SPATIAL REFERENCE SYSTEM 文

空間参照系を削除します。

クイックリンク：

「パラメータ」 (340 ページ)

「標準」 (340 ページ)

「パーミッション」 (340 ページ)

構文

```
DROP SPATIAL REFERENCE SYSTEM [ IF EXISTS ] name
```

パラメータ

(先頭に戻る) (340 ページ)

- **IF EXISTS** – 存在しない空間参照系を **DROP SPATIAL REFERENCE SYSTEM** 文が削除しようとしたときにエラーを返さないようにします。

標準

(先頭に戻る) (340 ページ)

ANSI SQL – 準拠レベル：Transact-SQL 拡張。

パーミッション

(先頭に戻る) (340 ページ)

次のいずれかが必要です。

- **MANAGE ANY SPATIAL OBJECT** システム権限
- **DROP ANY OBJECT** システム権限
- 空間参照系を所有していること

DROP SPATIAL UNIT OF MEASURE 文

空間測定単位を削除します。

クイックリンク：

「パラメータ」 (341 ページ)

「例」 (341 ページ)

「標準」 (341 ページ)

「パーミッション」 (341 ページ)

構文

DROP SPATIAL UNIT OF MEASURE [IF EXISTS] *identifier*

パラメータ

(先頭に戻る) (340 ページ)

- **IF EXISTS** – 存在しない空間測定単位を **DROP SPATIAL UNIT OF MEASURE** 文が削除しようとしたときにエラーを返さないようにします。

例

(先頭に戻る) (340 ページ)

- **例** – 次の例は、Test という名前の架空の空間測定単位を削除します。

```
DROP SPATIAL UNIT OF MEASURE Test;
```

標準

(先頭に戻る) (340 ページ)

ANSI SQL – 準拠レベル：Transact-SQL 拡張。

パーミッション

(先頭に戻る) (340 ページ)

次のいずれかが必要です。

- **MANAGE ANY SPATIAL OBJECT** システム権限
- **DROP ANY OBJECT** システム権限
- 空間測定単位を所有していること

DROP STATEMENT 文 [ESQL]

指定した準備文によって使われているリソースを解放します。これらのリソースは、**PREPARE** 文を実行して割り付けられます。通常、データベース接続が解放されるまで、リソースは解放されません。

クイックリンク：

SQL 文

「パラメータ」 (342 ページ)

「例」 (342 ページ)

「標準」 (342 ページ)

「パーミッション」 (342 ページ)

構文

```
DROP STATEMENT [ owner. ] statement-name
```

パラメータ

(先頭に戻る) (341 ページ)

- **statement-name** – 識別子またはホスト変数

例

(先頭に戻る) (341 ページ)

- **例 1** – *s1* 文と *stmt* 文を削除します。

```
EXEC SQL DROP STATEMENT s1;  
EXEC SQL DROP STATEMENT :stmt;
```

標準

(先頭に戻る) (341 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Open Client/Open Server ではサポートされていません。

パーミッション

(先頭に戻る) (341 ページ)

文が準備されている必要があります。

参照：

- PREPARE 文 [ESQL] (469 ページ)

DROP TEXT CONFIGURATION 文

テキスト設定オブジェクトを削除します。

注意： この文は、非構造化データ分析 (IQ_UDA) ライセンスを必要とします。

クイックリンク：

「例」 (343 ページ)

「使用法」 (343 ページ)

「パーミッション」 (343 ページ)

構文

```
DROP TEXT CONFIGURATION [ owner. ] text-config-name
```

例

(先頭に戻る) (343 ページ)

- **例 1** – mytextconfig テキスト設定オブジェクトを作成して、削除します。

```
CREATE TEXT CONFIGURATION mytextconfig FROM default_char;  
DROP TEXT CONFIGURATION mytextconfig;
```

使用法

(先頭に戻る) (343 ページ)

テキスト設定オブジェクトを削除するには、**DROP TEXT CONFIGURATION** を使用します。

依存する TEXT インデックスがあるテキスト設定オブジェクトを削除しようとすると、エラーが発生します。テキスト設定オブジェクトを削除する前に、依存する TEXT インデックスを削除する必要があります。

テキスト設定オブジェクトは、ISYSTEMTEXTCONFIG システムテーブルに格納されます。

関連する動作：

- オートコミット。

パーミッション

(先頭に戻る) (343 ページ)

SQL 文

自分が所有するテキスト設定オブジェクトの場合、何も必要ありません。
任意のユーザが所有するテキスト設定オブジェクトの場合、次のいずれかが必要です。

- DROP ANY TEXT CONFIGURATION システム権限
- DROP ANY OBJECT システム権限

DROP TEXT INDEX 文

データベースから TEXT インデックスを削除します。

注意： この文は、非構造化データ分析 (IQ_UDA) ライセンスを必要とします。。

クイックリンク：

「パラメータ」 (344 ページ)

「例」 (344 ページ)

「使用法」 (344 ページ)

「パーミッション」 (345 ページ)

構文

```
DROP TEXT INDEX text-index-name  
ON [ owner ] table-name
```

パラメータ

(先頭に戻る) (344 ページ)

- ON - TEXT インデックスが構築されているテーブルを指定します。

例

(先頭に戻る) (344 ページ)

- **例 1** - TextIdx TEXT インデックスを作成して、削除します。

```
CREATE TEXT INDEX TextIdx ON Customers ( Street );  
DROP TEXT INDEX TextIdx ON Customers;
```

使用法

(先頭に戻る) (344 ページ)

テキスト設定オブジェクトを削除する前に、依存する TEXT インデックスを削除する必要があります。

関連する動作：

- オートコミット。

パーミッション

(先頭に戻る) (344 ページ)

次のいずれかが必要です。

- DROP ANY INDEX システム権限
- DROP ANY OBJECT システム権限
- インデックス付きのテーブルに対する REFERENCES 権限
- 基本となるテーブルを所有している

DROP TRIGGER 文

データベースからトリガを削除します。

構文

```
DROP TRIGGER [ IF EXISTS ] [ owner. ] [ table-name. ] trigger-name
```

備考

IF EXISTS 句は、存在しないデータベースオブジェクトをこの DROP 文が削除しようとしたときにエラーを返さないようにする場合に使用します。

権限

テーブルのトリガを削除するには、次のいずれかの条件に該当する必要があります。

- そのテーブルの所有者である。
- そのテーブルに対する ALTER 権限を持っている。
- ALTER ANY TABLE システム権限を持っている。
- ALTER ANY OBJECT システム権限を持っている。

他のユーザが所有するビュートリガを削除するには、ALTER ANY VIEW または ALTER ANY OBJECT システム権限が必要です。

関連する動作

オートコミット。Interactive SQL で [結果] ウィンドウ枠の [結果] タブをクリアします。

標準と互換性

- **SQL/2008 – DROP TRIGGER** は、SQL/2008 標準のオプションの SQL 言語機能 T211、"Basic trigger capability" の一部です。IF EXISTS 句はベンダー拡張です。

例

この例は、Employees テーブルの更新前に郵便番号を大文字にする emp_upper_postal_code というトリガを作成した後、削除します。このトリガが存在しない場合は、エラーが返されます。

```
CREATE TRIGGER emp_upper_postal_code
BEFORE UPDATE OF PostalCode
ON GROUPO.Employees
REFERENCING NEW AS new_emp
FOR EACH ROW
WHEN ( ISNUMERIC( new_emp.PostalCode ) = 0 )
BEGIN
    -- Ensure postal code is uppercase (employee might be
    -- in Canada where postal codes contain letters)
    SET new_emp.PostalCode = UPPER(new_emp.PostalCode)
END;
DROP TRIGGER MyTrigger;
```

DROP USER 文

ユーザを削除します。

クイックリンク：

「パラメータ」 (346 ページ)

「例」 (347 ページ)

「標準」 (347 ページ)

「パーミッション」 (347 ページ)

構文

```
DROP USER user-name
```

パラメータ

(先頭に戻る) (346 ページ)

- **user-name** – 削除するユーザの名前。

例

(先頭に戻る) (346 ページ)

- **例 1** - データベースからユーザ SQLTester を削除します。

```
DROP USER SQLTester
```

標準

(先頭に戻る) (346 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - Adaptive Server ではサポートされていません。

パーミッション

(先頭に戻る) (346 ページ)

MANAGE ANY USER システム権限が必要です。

注意: ユーザを削除すると、このユーザが所有しているオブジェクトとこのユーザが付与した権限がすべて削除されます。

参照:

- CREATE LOGIN POLICY 文 (194 ページ)
- CREATE USER 文 (287 ページ)
- DROP LOGIN POLICY 文 (330 ページ)
- ALTER LOGIN POLICY 文 (32 ページ)
- GRANT ROLE 文 (394 ページ)
- GRANT システム権限文 (402 ページ)
- REVOKE システム権限文 (512 ページ)
- REVOKE ROLE 文 (508 ページ)

DROP VARIABLE 文

CREATE VARIABLE 文を使用して作成した SQL 変数を削除します。

クイックリンク:

「パラメータ」 (348 ページ)

「使用法」 (348 ページ)

「標準」 (348 ページ)

SQL 文

[「パーミッション」](#) (348 ページ)

構文

```
DROP VARIABLE [ IF EXISTS ] identifier
```

パラメータ

[\(先頭に戻る\)](#) (347 ページ)

- **IF EXISTS** - 存在しないデータベースオブジェクトをこの **DROP VARIABLE** 文が削除しようとしたときにエラーを返さないようにします。

使用法

[\(先頭に戻る\)](#) (347 ページ)

データベース接続を解放すると、変数は自動的に削除されます。大きいオブジェクト用に変数がよく使用されます。したがって、使い終わった変数を削除したり、NULL に設定したりすると、リソース (主にディスク領域) が大幅に解放される場合があります。

標準

[\(先頭に戻る\)](#) (347 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Adaptive Server ではサポートされていません。

パーミッション

[\(先頭に戻る\)](#) (347 ページ)

なし

参照:

- CREATE VARIABLE 文 (289 ページ)
- SET 文 [ESQL] (534 ページ)

EXECUTE 文 [ESQL]

SQL 文を実行します。

クイックリンク:

[「パラメータ」](#) (349 ページ)

「例」 (350 ページ)

「使用法」 (350 ページ)

「標準」 (350 ページ)

「パーミッション」 (351 ページ)

構文

構文 1 - 事前に準備された指定の動的文を実行します。

```
EXECUTE statement-name
... [ { USING DESCRIPTOR sqlda-name | USING host-variable-list } ]
... [ { INTO DESCRIPTOR into-sqlda-name | INTO into-host-variable-list ]
... [ ARRAY :nnn } ]
```

構文 2 - バインド変数や出力がない文の **PREPARE** と **EXECUTE** の省略形です。

```
EXECUTE IMMEDIATE statement
```

パラメータ

(先頭に戻る) (348 ページ)

- **statement-name** – 識別子またはホスト変数
- **sqlda-name** – 識別子
- **into-sqlda-name** – 識別子
- **statement** – 文字列またはホスト変数
- **USING** – **SELECT** 文または **CALL** 文の **OUTPUT** は、変数リストの変数、または指定した **SQLDA** が記述するプログラムデータ領域のどちらかに格納されます。**OUTPUT** (**select** リストまたはパラメータ) と、ホスト変数リストまたは **SQLDA** 記述子配列のいずれかとの対応は、1 対 1 です。
- **INTO** – **INSERT** 文とともに使用した場合、挿入されたローが 2 番目の記述子に返されます。たとえば、プライマリーキー値を生成するオートインクリメントプライマリーキーを使用した場合、**EXECUTE** 文によって、即座にローを再フェッチしてそのローに割り当てられているプライマリーキー値を決定するためのメカニズムが提供されます。
- **ARRAY** – 準備された **INSERT** 文とともに使用することによって、ワイド挿入が可能になります。ワイド挿入では、複数のローを一度に挿入できるため、パフォーマンスが向上することがあります。値 *nnn* は、挿入するローの数です。**SQLDA** には *nnn* * (ローあたりのカラム数) 変数を指定する必要があります。最

初のローは `SQLDA` の変数 0 ~ (ローあたりのカラム数) -1 に入り、以後のローも同様です。

例

(先頭に戻る) (348 ページ)

- **例 1 – DELETE** を実行します。

```
EXEC SQL EXECUTE IMMEDIATE  
'DELETE FROM Employees WHERE EmployeeID = 105';
```

- **例 2 – 準備された DELETE 文** を実行します。

```
EXEC SQL PREPARE del_stmt FROM  
'DELETE FROM Employees WHERE EmployeeID = :a';  
EXEC SQL EXECUTE del_stmt USING :employee_number;
```

- **例 3 – 準備されたクエリ** を実行します。

```
EXEC SQL PREPARE sell FROM  
'SELECT Surname FROM Employees WHERE EmployeeID = :a';  
EXEC SQL EXECUTE sell USING :employee_number INTO :emp_lname;
```

使用法

(先頭に戻る) (348 ページ)

構文 1 - 要求に関する情報を提供するホスト変数プレースホルダ (バインド変数) が動的な文に含まれている場合は、文に出現するすべてのバインド変数の記述子を格納する `SQLDA` を指す C 変数を `sqlda-name` で指定するか、または `host-variable-list` でバインド変数を指定する必要があります。

構文 2 - 文字列またはホスト変数の中にある SQL 文を即座に実行し、完了時に削除します。

EXECUTE 文は、準備可能な任意の SQL 文に使用できます。データベースから多数のローを返す **SELECT** 文または **CALL** 文にはカーソルを使用します。

注意： **EXECUTE** 文では、テーブル UDF は参照できません。

INSERT、**UPDATE**、または **DELETE** 文が正常に実行されると、操作によって変更されたロー数が `SQLCA` (`SQLCOUNT` の `sqlerrd[2]` フィールド) に挿入されます。

標準

(先頭に戻る) (348 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Open Client/Open Server でサポートされています。

パーミッション

(先頭に戻る) (348 ページ)

実行される文に対するパーミッションを確認します。

参照：

- DECLARE CURSOR 文 [ESQL] [SP] (299 ページ)
- PREPARE 文 [ESQL] (469 ページ)

EXECUTE 文 [T-SQL]

Adaptive Server と互換性のある、**CALL** 文の代替としてプロシージャを呼び出します。

クイックリンク：

「例」 (351 ページ)

「使用法」 (352 ページ)

「パーミッション」 (352 ページ)

構文

```
EXECUTE [ @return_status = ] [owner.]procedure_name
... { [ @parameter-name = ] expression
| [ @parameter-name = ] @variable [ output ] } ,...
```

例

(先頭に戻る) (351 ページ)

- **例 1** – プロシージャ p1 を作成します。

```
CREATE PROCEDURE p1( @var INTEGER = 54 )
AS
PRINT 'on input @var = %1! ', @var
DECLARE @intvar integer
SELECT @intvar=123
SELECT @var=@intvar
PRINT 'on exit @var = %1!', @var;
```

次の文では、パラメータに 23 を入力してプロシージャを実行します。Open Client アプリケーションから接続している場合は、クライアントウィンドウに **PRINT** メッセージが表示されます。ODBC または Embedded SQL アプリケーションから接続している場合は、データベースサーバウィンドウにメッセージが表示されます。

SQL 文

```
EXECUTE p1 23
```

次の文を使用してプロシージャを実行することもできます。複数のパラメータがあるときに便利です。

```
EXECUTE p1 @var = 23
```

次の文では、パラメータにデフォルト値を使用してプロシージャを実行します。

```
EXECUTE p1
```

次の文では、プロシージャを実行し、リターンステータスを確認できるように戻り値を変数に格納します。

```
EXECUTE @status = p1 23
```

使用法

(先頭に戻る) (351 ページ)

EXECUTE では、ストアードプロシージャが実行されます。オプションとして、プロシージャパラメータを指定し、出力値とリターンステータス情報を取得することもできます。

EXECUTE は、Transact-SQL との互換性のために実装されていますが、Transact-SQL と SAP Sybase IQ のどちらのバッチおよびプロシージャでも使用できます。

注意： **EXECUTE** 文では、テーブル UDF は参照できません。

パーミッション

(先頭に戻る) (351 ページ)

プロシージャの所有者であるか、プロシージャの **EXECUTE** パーミッションを保持しているか、または **EXECUTE ANY PROCEDURE** システム権限を保持していることが必要です。

参照：

- **CALL** 文 (115 ページ)

EXECUTE IMMEDIATE 文 [**ESQL**] [**SP**]

プロシージャから実行可能な文の範囲を拡張します。これにより、プロシージャに渡されるパラメータを使用して構成された文など、動的に準備される文の実行が可能になります。

クイックリンク：

「パラメータ」 (353 ページ)

「例」 (354 ページ)

「使用法」 (354 ページ)

「標準」 (355 ページ)

「パーミッション」 (355 ページ)

構文

構文 1

```
EXECUTE IMMEDIATE [ execute-option ] string-expression
```

execute-option :

```
  WITH QUOTES [ ON | OFF ]
  | WITH ESCAPES { ON | OFF }
  | WITH RESULT SET { ON | OFF }
```

構文 2

```
EXECUTE ( string-expression )
```

パラメータ

(先頭に戻る) (352 ページ)

- **WITH QUOTES [ON]** – 文字列式にあるすべての二重引用符が、識別子のデリミタとみなされます。指定されていない場合、文字列式内の二重引用符は、**QUOTED_IDENTIFIER** データベースオプションの現在の設定に応じて処理されます。

WITH QUOTES は、実行される文がストアードプロシージャに渡されるオブジェクト名を使って構成されるが、オブジェクト名に二重引用符が必要であり、プロシージャが呼び出されるとき **QUOTED_IDENTIFIER** が **OFF** に設定されている、といった場合に便利です。

「**QUOTED_IDENTIFIER** オプション [TSQL]

」を参照してください。

- **WITH ESCAPES** – 文字列式のエスケープシーケンス (**¥n**、**¥x**、**¥¥** など) がすべて無視されます。たとえば、連続する 2 つの円記号は、1 つの円記号に変換されるのではなく、2 つの円記号として保持されます。デフォルト設定は **ON** です。
- **WITH ESCAPES OFF** を使用すると、円記号を含んだファイル名を参照したり、動的に構成された文の実行を簡単にしたりできます。
- **string-expression** – 場合によっては、*string-expression* 内のエスケープシーケンスが、**EXECUTE IMMEDIATE** を実行する前に変換されることもあります。たとえ

ば、複合文は実行される前に解析されますが、この解析中に WITH ESCAPES の設定とは関係なく、エスケープシーケンスの変換が行われます。このような場合には、**WITH ESCAPES OFF** を指定することで、それ以上変換が行われるのを防ぎます。次に例を示します。

```
BEGIN
DECLARE String1 LONG VARCHAR;
DECLARE String2 LONG VARCHAR;
EXECUTE IMMEDIATE
  'SET String1 = 'One backslash: \\\\'';
EXECUTE IMMEDIATE WITH ESCAPES OFF
  'SET String2 = 'Two backslashes: \\\\'';
SELECT String1, String2
END
```

- **WITH RESULT SET – ON** と指定されると、**EXECUTE IMMEDIATE** 文が結果セットを返します。この句を使用すると、この句を含むプロシージャが結果セットを返すように指定されます。この句を指定しない場合、プロシージャを呼び出したとき文が結果セットを生成しなければエラーが返されます。

注意： デフォルトオプションは OFF で、文の実行時に結果セットが生成されないことを意味します。

例

(先頭に戻る) (352 ページ)

- **例 1** – プロシージャに対するパラメータとして指定されたテーブル名でテーブルを作成します。**EXECUTE IMMEDIATE** 文は全体を 1 行で指定してください。

```
CREATE PROCEDURE CreateTableProc(
    IN tablename char(30)
)
BEGIN
  EXECUTE IMMEDIATE 'CREATE TABLE ' || tablename ||
    ' ( column1 INT PRIMARY KEY) '
END;
```

プロシージャを呼び出し、mytable というテーブルを作成します。:

```
CALL CreateTableProc( 'mytable' )
```

使用法

(先頭に戻る) (352 ページ)

文内のリテラル文字列は、一重引用符で囲む必要があります。また、**PREPARE** 文や **EXECUTE IMMEDIATE** 文にある既存のどの文の名前とも同じでないことが必要です。文は 1 行に記述してください。

EXECUTE IMMEDIATE で実行される文から参照できるのはグローバル変数だけです。

Transact-SQL のストアードプロシージャの内部では、構文 2 だけが使用できます。

関連する動作：

なし。ただし、関連する動作としてのオートコミットを伴うデータ定義文である場合は、そのコミットが起こります。

標準

(先頭に戻る) (352 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - Open Client/Open Server でサポートされています。

パーミッション

(先頭に戻る) (352 ページ)

なし。文は、プロシージャの所有者のパーミッションを使用して実行されます。プロシージャを呼び出すユーザのパーミッションは使いません。

参照：

- BEGIN ... END 文 (108 ページ)
- CREATE PROCEDURE 文 (210 ページ)
- QUOTED_IDENTIFIER オプション [TSQL] (759 ページ)

EXIT 文 [Interactive SQL]

Interactive SQL を終了します。

クイックリンク：

「例」 (356 ページ)

「使用法」 (356 ページ)

「標準」 (357 ページ)

「パーミッション」 (357 ページ)

構文

```
{ EXIT | QUIT | BYE } [ return-code ]
```

return-code*number | connection-variable***例**

(先頭に戻る) (355 ページ)

- **例 1** – テーブル T にローが存在する場合は Interactive SQL の戻り値を 1 に設定し、テーブル T にローがない場合は 0 に設定します。

```
CREATE VARIABLE rowCount INT;
CREATE VARIABLE retcode INT;
SELECT COUNT(*) INTO rowCount FROM T;
IF( rowCount > 0 ) THEN
    SET retcode = 1;
ELSE
    SET retcode = 0;
END IF;
EXIT retcode;
```

注意： 次の文は指定できません。これは、**EXIT** が (SQL 文ではなく) Interactive SQL 文であり、Interactive SQL 文を他の SQL ブロック文に含めることはできないためです。

```
CREATE VARIABLE rowCount INT;
SELECT COUNT(*) INTO rowCount FROM T;
IF( rowCount > 0 ) THEN
    EXIT 1;    // <-- not allowed
ELSE
    EXIT 0;    // <-- not allowed
END IF;
```

使用法

(先頭に戻る) (355 ページ)

Interactive SQL をウィンドウプログラムとして実行している場合には Interactive SQL ウィンドウを閉じ、コマンドプロンプト (バッチ) モードで実行している場合には Interactive SQL を完全に終了します。どちらの場合でも、データベース接続も閉じられます。COMMIT_ON_EXIT オプションが ON に設定されている場合は、データベース接続を閉じる前に Interactive SQL が自動的に **COMMIT** 文を実行します。このオプションが OFF に設定されている場合は、Interactive SQL は暗黙の **ROLLBACK** を実行します。デフォルトでは、COMMIT_ON_EXIT オプションは ON に設定されています。

オプションのリターンコードをバッチファイルで使用すると、Interactive SQL コマンドファイルにコマンドの成功または失敗が示されます。デフォルトのリターンコードは 0 です。

関連する動作：

- この文は、オプション `COMMIT_ON_EXIT` が `ON` (デフォルト) に設定されている場合は自動的にコミットを実行し、そうでない場合は暗黙のロールバックを実行します。
- Windows オペレーティングシステムでは、オプションの戻り値を `ERRORLEVEL` として使用できます。

標準

(先頭に戻る) (355 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Adaptive Server では使用できません。

パーミッション

(先頭に戻る) (355 ページ)

なし

参照：

- `SET OPTION` 文 (542 ページ)

FETCH 文 [ESQL] [SP]

指定したカーソルからローを 1 つ取り出します。カーソルを事前に開いておく必要があります。

クイックリンク：

「パラメータ」 (358 ページ)

「例」 (359 ページ)

「使用法」 (360 ページ)

「標準」 (361 ページ)

「パーミッション」 (361 ページ)

構文

```

FETCH
  { NEXT | PRIOR | FIRST | LAST
  | ABSOLUTE row-count | RELATIVE row-count }
  ... cursor-name
  ... { [ INTO host-variable-list ]
  | USING DESCRIPTOR sqlda-name
  | INTO variable-list }
```

```
... [ PURGE ] [ BLOCK n ] [ ARRAY fetch-count ]
... INTO variable-list
... IQ CACHE row-count
```

次の句は、Embedded SQL の中だけで使用します。

- **USING DESCRIPTOR** *sqlda-name*
- **INTO** *host-variable-list*
- **PURGE**
- **BLOCK** *n*
- **ARRAY** *fetch-count*
- *cursor-name* および *row-count* での *host-variable* の使用

パラメータ

(先頭に戻る) (357 ページ)

- **cursor-name** – 識別子またはホスト変数
- **qlda-name** – 識別子
- **host-variable-list** – インジケータ変数を含めることが可能
- **row-count** – 数値またはホスト変数
- **fetch-count** – 整数またはホスト変数
- **INTO** – これを指定しない場合、**FETCH** はカーソルだけを配置します。OPEN は、まず、先頭のローの前にカーソルを配置します。オプションの配置パラメータを指定すると、カーソルを移動してからローをフェッチできます。
- **NEXT** – (デフォルト) ローをフェッチする前に、カーソルを 1 つ前方のローに進めます。
- **PRIOR** – ローをフェッチする前に、カーソルを 1 つ前のローに戻します。
- **ABSOLUTE** – 使用すると、特定のローに移動できます。ゼロは最初のローの前の位置を示します。

1 は先頭のロー、2 は 2 番目のロー、のようになります。負の数を使用して、カーソルの最後からの絶対位置を指定します。-1 はカーソルの最後のローを示します。FIRST は ABSOLUTE 1 の省略形です。LAST は ABSOLUTE -1 の省略形です。

注意： SAP Sybase IQ では、FIRST、LAST、ABSOLUTE、および負数の RELATIVE 句を他の DBMS 製品ほど効率よく処理できないため、これらの句を使用するとパフォーマンスが低下します。

- **RELATIVE** – フェッチする前にいずれかの方向へ指定した数のローだけカーソルを移動します。

正の数は前進を示し、負の数は後退を示します。したがって、NEXT は RELATIVE 1 と等しく、PRIOR は RELATIVE -1 と等しくなります。RELATIVE 0 は、このカーソル上の最後の FETCH 文と同じローを取り出します。

- **BLOCK** – アプリケーションでフェッチ可能なローの個数に関する情報をクライアントとサーバに提供します。0 という特別な値は、要求がサーバに送信されてシングルローが返却されることを示します (ローブロッキングはありません)。
- **PURGE** – クライアントはすべてのローのバッファをフラッシュし、次にフェッチ要求をサーバに送信します。このフェッチ要求はローのブロックを返すことに注意してください。
- **ARRAY** – ワイドフェッチを可能にします。複数のローを同時に取り出せるため、パフォーマンスが向上します。Embedded SQL でワイドフェッチを使用するには、コードに次のような **FETCH** 文を含めます。

```
EXEC SQL FETCH . . . ARRAY nnn
```

ARRAY nnn は **FETCH** 文の最後の項目です。フェッチ回数を示す nnn にはホスト変数も使用できます。SQLDA には nnn * (ローあたりのカラム数) 変数を指定する必要があります。最初のローは SQLDA の変数 0 ~ 変数 (ローあたりのカラム数) -1 に入り、以後のローも同様です。

- **IQ CACHE** – FIFO キューにバッファされるローの最大数を指定します。IQ CACHE に値を指定しなければ、**CURSOR_WINDOW_ROWS** データベースオプションの値が使用されます。**CURSOR_WINDOW_ROWS** のデフォルト設定は 200 です。

例

(先頭に戻る) (357 ページ)

- **例 1** – Embedded SQL の例です。

```
EXEC SQL DECLARE cur_employee CURSOR FOR
SELECT EmployeeID, Surname FROM Employees;
EXEC SQL OPEN cur_employee;
EXEC SQL FETCH cur_employee
INTO :emp_number, :emp_name:indicator;
```

- **例 2** – プロシージャの例です。

```
BEGIN
  DECLARE cur_employee CURSOR FOR
    SELECT Surname
    FROM Employees;
```

```
DECLARE name CHAR(40) ;
OPEN cur_employee;
LOOP
    FETCH NEXT cur_employee into name ;
    .
    .
    .
END LOOP
CLOSE cur_employee;
END
```

使用法

(先頭に戻る) (357 ページ)

SELECT 文の結果から 1 つのローを、変数リストの変数に格納します。select リスト対ホスト変数リストは、1 対 1 に対応します。

SELECT の結果から 1 つまたは複数のローを、変数リストの変数、または指定した **SQLDA** が記述するプログラムデータ領域のどちらかに格納します。select リスト対ホスト変数リスト、または select リスト対 **SQLDA** 記述子配列は、どちらも 1 対 1 に対応します。

FOR READ ONLY として宣言されたカーソルからは、最初に **FETCH** を実行したときのテーブルのバージョンではなく、カーソルをオープンしたときにカーソルの宣言に使用されるテーブルのバージョンが表示されます。

FETCH 文に指定された配置パラメータがカーソル位置の許容範囲外である場合は、**SQLC_NOTFOUND** が警告されます。

C 言語のソースコード内では、**DECLARE CURSOR** 文が **FETCH** 文の前になるようにしてください。また、**FETCH** 文よりも先に **OPEN** 文を実行します。カーソル名の代わりにホスト変数を使用している場合は、**DECLARE** 文が実際にコードを生成するため、**FETCH** の前に実行する必要があります。

マルチユーザ環境では、ローが複数のクライアントから一度にフェッチされる可能性があります。これはブロックフェッチまたはマルチローフェッチと呼ばれます。最初のフェッチによって、複数のローがサーバから送り返されます。クライアントは、これらのローをバッファに格納します。後に続くフェッチは、サーバへ新しい要求を行わないで、これらのバッファからローを取り出します。

フェッチ時に **SQLSTATE_NOTFOUND** 警告が返される場合は、**SQLCA** (**SQLCOUNT**) の **sqlerrd[2]** フィールドに、カーソル位置の許容範囲を超えてフェッチを試みたときのローの数が入ります (カーソルは、ローの上にも、最初のローより前または最後のローより後にも置くことができます)。ローが見つからなくても位置が有効な場合の値は 0 です。たとえば、カーソル位置が最後のローのときに、**RELATIVE 1** 句を使用して **FETCH** を実行した場合です。実行しようとするフェッ

チがカーソルの最後よりも後の場合、値は正になります。実行しようとするフェッチがカーソルの先頭よりも前の場合、値は負になります。

FETCH 文が正常に実行されると、SQLCA (SQLIOCOUNT) の sqlerrd[1] フィールドは、フェッチの実行に必要な入出力操作の回数だけ増加します。このフィールドは、実際にはデータベース文が発行されるたびに増加します。

サーバは、フェッチしたレコードの数を SQLCOUNT に返します。エラーがない場合は、0 より大きい SQLCOUNT を常に返します。古いバージョンのサーバはローを1つだけ返し、SQLCOUNT は0に設定されます。エラーではなくて SQLCOUNT が0の場合、有効なローが1つフェッチされたことを示します。

標準

(先頭に戻る) (357 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - Adaptive Server でサポートされています。

パーミッション

(先頭に戻る) (357 ページ)

カーソルをオープンしている必要があります。また、ユーザには、カーソル宣言内で参照するテーブルの SELECT パーミッションが必要です。

参照：

- DECLARE CURSOR 文 [ESQL] [SP] (299 ページ)
- OPEN 文 [ESQL] [SP] (461 ページ)
- PREPARE 文 [ESQL] (469 ページ)
- CURSOR_WINDOW_ROWS オプション (637 ページ)

FOR 文

カーソル内の各ローに対して一度ずつ、文リストの実行を繰り返します。

クイックリンク：

「パラメータ」 (362 ページ)

「例」 (363 ページ)

「使用法」 (363 ページ)

「標準」 (364 ページ)

「パーミッション」 (364 ページ)

構文

```
[ statement-label: ]
  FOR for-loop-name AS cursor-name [ cursor-type ] CURSOR
  { FOR statement
  ... [ { FOR { UPDATE cursor-concurrency | FOR READ ONLY } ]
      | USING variable-name }
  DO statement-list
  END FOR [ statement-label ]
```

cursor-type - (構文に戻る)

```
NO SCROLL
| DYNAMIC SCROLL
| SCROLL
| INSENSITIVE
| SENSITIVE
```

cursor-concurrency - (構文に戻る)

```
BY { VALUES
| TIMESTAMP
| LOCK }
```

variable-name - (構文に戻る)

```
identifier
```

パラメータ

(先頭に戻る) (361 ページ)

- **NO SCROLL** - **NOSCROLL** として宣言されたカーソルは、**FETCH NEXT** と **FETCH RELATIVE 0** の各シーク操作を使用して、結果セット内の移動が前方向に制限されます。カーソルがローを出た後は、そのローに戻ることができないため、カーソルに sensitivity の制限はありません。NO SCROLL カーソルを要求されると、データベースサーバは最も効率的な種類のカーソルである asensitive カーソルを提供します。
- **DYNAMIC SCROLL** - **DYNAMIC SCROLL** はデフォルトカーソルタイプです。DYNAMIC SCROLL カーソルでは、**FETCH** 文のすべてのフォーマットを使用できます。DYNAMIC SCROLL カーソルを要求されると、データベースサーバは asensitive カーソルを提供します。カーソルを使用する場合、効率と一貫性の間には常にトレードオフ関係があります。asensitive カーソルの場合、パフォーマンスは効率化されますが、その代償として一貫性が損なわれます。
- **SCROLL** - **SCROLL** として宣言されたカーソルは、**FETCH** 文のすべてのフォーマットを使用できます。SCROLL カーソルを要求されると、データベースサーバは value-sensitive カーソルを提供します。データベースサーバは、結果セットのメンバーシップが保証されるような方法で value-sensitive カーソルを実行する必要があります。DYNAMIC SCROLL カーソルの方が効率的です。

SCROLL カーソルの一貫した動作が必要でない場合は、DYNAMIC SCROLL カーソルを使用してください。

- **INSENSITIVE – INSENSITIVE** として宣言されたカーソルは、存続期間中、値とメンバーシップが固定されます。**SELECT** 文の結果セットはカーソルが開かれると、実体化されます。INSENSITIVE カーソルからの FETCHING は、そのカーソルを開いた接続を含むどの接続からの **INSERT**、**UPDATE**、**MERGE**、**PUT**、**DELETE** 文の影響も受けません。
- **SENSITIVE – SENSITIVE** として宣言されたカーソルは、結果セットのメンバーシップまたは値の変更に依存します。

例

(先頭に戻る) (361 ページ)

- **例 1** – 次のコードフラグメントは、**FOR** ループの使用法を示します。

```
FOR names AS curs CURSOR FOR
SELECT Surname
FROM Employees
DO
    CALL search_for_name( Surname );
END FOR;
```

使用法

(先頭に戻る) (361 ページ)

FOR は、カーソル内の各ローに対して一度ずつ SQL 文のリストを実行できる制御文です。

FOR 文は、カーソルの **DECLARE** と、カーソルの結果セットのそれぞれのカラムに対する変数の **DECLARE** があり、その後には、カーソルから 1 つのローをフェッチしてローカル変数に格納し、カーソル内の各ローに対して一度だけ *statement-list* を実行するループが続く複合文と同じです。

ローカル変数の名前とデータ型は、カーソル内で使用する *statement* にもとづいて宣言されます。**SELECT** 文の場合、データ型は select リスト中の式のデータ型です。名前は、それらが存在する select リスト項目のエイリアスです。そうでない場合、カラムの名前になります。select リスト項目が単一のカラム参照でない場合は、エイリアスが必要です。**CALL** 文の場合、名前とデータ型はプロシージャ定義内の **RESULT** 句から取得されます。

LEAVE 文を使用して、**END FOR** に続く最初の文から実行を再開できます。終了の *statement-label* を指定する場合は、開始の *statement-label* と一致させる必要があります。

SQL 文

標準

(先頭に戻る) (361 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - Adaptive Server ではサポートされていません。

パーミッション

(先頭に戻る) (361 ページ)

なし

参照：

- DECLARE CURSOR 文 [ESQL] [SP] (299 ページ)
- FETCH 文 [ESQL] [SP] (357 ページ)
- LEAVE 文 (428 ページ)
- LOOP 文 (455 ページ)

FOR JSON 文

SELECT 文内で FOR JSON 句を使用することによって、データベースに対して SQL クエリを実行し、その結果を JSON ドキュメントとして返すことができます。

JavaScript Object Notation (JSON) は、言語に依存しないテキストベースのデータ交換フォーマットで、JavaScript データの直列化のために開発されました。また、JSON では、オブジェクトと配列という 2 つの構造型も表現されます。JSON フォーマットの詳細については、<http://www.json.org/>

クイックリンク：

「パラメータ」 (365 ページ)

「例」 (365 ページ)

「使用法」 (369 ページ)

「パーミッション」 (370 ページ)

を参照してください。

構文

```
...For JSON mode
```

```
mode
```

```
RAW | AUTO | EXPLICIT
```


パラメータ

(先頭に戻る) (364 ページ)

- **mode** – 出力フォーマットを制御します。
- **RAW** – クエリ結果をフラットにされた JSON 表現として返します。このモードはより冗長性が高いですが、解析が容易です。RAW 句は、解析と理解が最も容易な方法であるため、クエリ結果を JSON オブジェクトとして取得する際に推奨される推奨方法です。
- **AUTO** – クエリのジョインに基づいて、クエリの結果をネストされた JSON オブジェクトとして返します。結果セットで JSON オブジェクト間の階層関係を表示する場合は、クエリ内で AUTO 句を使用します。
- **EXPLICIT** – カラムデータの表現方法を指定できます。カラムを単一の値、オブジェクト、またはネストされたオブジェクトとして指定して、統一された配列または異なる配列を生成できます。

EXPLICIT 句は、カラムエイリアスを使用して、詳細なフォーマット仕様を提供します。エイリアスが存在しない場合は、指定されたカラムが値として出力されます。ネストされた構造内で値 (またはオブジェクト) を表すには、エイリアスが必要です。select-list タグと親内の最初の 2 つのカラムに名前を付ける必要があります。複数のクエリを結合して使用すると、各クエリ内のタグと親の関係を指定して、ネストされた JSON 出力を返すことができます。

例

(先頭に戻る) (364 ページ)

- **例 1** – JSON RAW を使用して、Employees テーブルから従業員情報を返します。

```
SELECT
    emp.EmployeeID,
    so.CustomerID,
    so.Region
FROM Employees AS emp KEY JOIN SalesOrders AS so WHERE
emp.EmployeeID <= 195
ORDER BY 1
FOR JSON RAW;
```

- **例 2** – 結果を階層としてネストする FOR JSON AUTO を使用した場合に返される結果とは異なり、FOR JSON RAW ではフラット化された結果セットが返されます。

```
[
  { "EmployeeID" : 129, "CustomerID" : 107, "Region" :
    "Eastern" },
  { "EmployeeID" : 129, "CustomerID" : 119, "Region" :
    "Western" },
```

```

...
  { "EmployeeID" : 129, "CustomerID" : 131, "Region" :
"Eastern" },
  { "EmployeeID" : 195, "CustomerID" : 176, "Region" : "Eastern" }
]

```

- **例 3** – FOR JSON RAW とは異なり、FOR JSON AUTO を使用すると、データがネストされた階層で返され、emp つまり Employee オブジェクトが CustomerID データの配列を含む so つまり SalesOrders オブジェクトで構成されます。

```

[
  { "emp":
    { "EmployeeID" : 129,
      "so" : [
        { "CustomerID" : 107 , "Region" : "Eastern" },
        ...
        { "CustomerID" : 131 , "Region" : "Eastern" }
      ]
    }
  },
  { "emp" :
    { "EmployeeID" : 195,
      "so" : [
        { "CustomerID" : 109 , "Region" : "Eastern" },
        ...
        { "CustomerID" : 176 , "Region" : "Eastern" }
      ]
    }
  }
]

```

- **例 4** – FOR JSON EXPLICIT を使用して、Employees テーブルから従業員情報を返します。

```

SELECT
  1 AS tag,
  NULL AS parent,
  emp.EmployeeID AS [!EmployeeID],
  so.CustomerID AS [!CustomerID],
  so.Region AS [!Region]
FROM Employees AS emp KEY JOIN SalesOrders AS so WHERE
emp.EmployeeID <= 195
ORDER BY 3
FOR JSON EXPLICIT;

```

FOR JSON RAW の例と同じ結果を返します。

```

[
  { "EmployeeID" : 129, "CustomerID" : 107, "Region" :
"Eastern" },
  { "EmployeeID" : 129, "CustomerID" : 119, "Region" :
"Western" },
  ...
  { "EmployeeID" : 129, "CustomerID" : 131, "Region" :
"Eastern" },

```

```
{ "EmployeeID" : 195, "CustomerID" : 176, "Region" : "Eastern" }
]
```

- **例 5 – FOR JSON AUTO の例と同様の結果を返します。**

```
SELECT
    1          AS tag,
    NULL      AS parent,
    emp.EmployeeID AS [emp!1!EmployeeID],
    null      AS [so!2!CustomerID],
    null      AS [!2!Region]
FROM Employees as emp where emp.EmployeeID <= 195
UNION ALL
SELECT
    2,
    1,
    emp.EmployeeID,
    so.CustomerID,
    so.Region
FROM Employees as emp KEY JOIN SalesOrders as so where
emp.EmployeeID <= 195
ORDER BY 3, 1
FOR JSON EXPLICIT;
```

上記のクエリは、次の結果を返します。

```
[
  { "emp": [{"EmployeeID":102}]},
  { "emp": [{"EmployeeID":105}]},
  { "emp":
    [ {"EmployeeID":129,
      "so": [
        {"CustomerID":101,"Region":"Eastern"},
        ...
        {"CustomerID":205,"Region":"Eastern"}
      ]
    }
  ],
  { "emp": [{"EmployeeID":148}]},
  { "emp": [{"EmployeeID":160}]},
  { "emp": [{"EmployeeID":184}]},
  { "emp": [{"EmployeeID":191}]},
  { "emp":
    [ {"EmployeeID":195,
      "so": [
        {"CustomerID":101,"Region":"Eastern"},
        ...
        {"CustomerID":209,"Region":"Western"}
      ]
    }
  ]
}
```

配列の順序と受注のない従業員を含めること以外で、上記のフォーマットが FOR JSON AUTO の結果と異なるのは、emp が構造体の配列であることだけです。FOR JSON AUTO では、emp に存在するオブジェクトが1つだけであると

理解されます。FOR JSON EXPLICIT は、集合をサポートする配列のカプセル化を使用します。

次の例では、emp のカプセル化を削除して、値として Region を返します。この例は、FOR JSON EXPLICIT モードが詳細なフォーマット制御を行って、RAW モードと AUTO モードの中間的な結果を生成する方法を示します。

```

SELECT
    1          AS tag,
    NULL      AS parent,
    emp.EmployeeID AS [!1!EmployeeID],           // remove "emp"
encapsulation
    null      AS [so!2!id],                       // change
"CustomerID" to just "id"
    null      AS [!2!]                             // stipulate that
region should be emitted as a value
FROM Employees AS emp WHERE emp.EmployeeID <= 195
UNION ALL
SELECT
    2,
    1,
    emp.EmployeeID,
    so.CustomerID,
    so.Region
FROM Employees as emp KEY JOIN SalesOrders AS so WHERE
emp.EmployeeID <= 195
ORDER BY 3, 1
FOR JSON EXPLICIT;

```

クエリの結果で、現在は so がオブジェクトの配列ではなく、2次元配列になっています。

```

[
  { "EmployeeID":102}, {"EmployeeID":105}, {"EmployeeID":129,
    "so":[
      [{"id":101}, "Eastern"],
      ...
      [{"id":205}, "Eastern"]
    ]
  },
  {"EmployeeID":148},
  {"EmployeeID":160},
  {"EmployeeID":184},
  {"EmployeeID":191},
  {"EmployeeID":195,
    "so":[
      [{"id":101}, "Eastern"],
      ...
      [{"id":209}, "Western"]
    ]
  }
]

```

次の例は、FOR JSON RAW を使用した場合に似ていますが、employeeID、CustomerID、Region が名前／値のペアとしてではなく、値として出力されます。

```
SELECT
  1 AS tag,
  NULL AS parent,
  emp.EmployeeID, // no alias directives
  so.CustomerID,
  so.Region
FROM Employees AS emp KEY JOIN SalesOrders AS so WHERE
emp.EmployeeID <= 195
ORDER BY 3
FOR JSON EXPLICIT;
```

クエリから、EmployeeID、CustomerID、Region で構成される 2 次元配列を生成する結果が返されます。

```
[
  [129,107,"Eastern"],
  ...
  [195,176,"Eastern"]
]
```

使用法

(先頭に戻る) (364 ページ)

FOR JSON 句は、サブクエリ、GROUP BY 句または集合関数を含むクエリ、ビュー定義など、あらゆる SELECT 文内で使用できます。FOR JSON 句の使用は、配列、オブジェクト、スカラ要素で構成される JSON 配列としてのリレーショナルデータに相当します。

エイリアスディレクティブのフォーマット：

```
[encapsulating_object!tag!name!qualifier]
```

各項目の意味は次のとおりです。

- **!** – ディレクティブの基準を区切ります。
- **encapsulating_object** – select-list 項目のカプセル化 (配列) オブジェクトを生成します。
- **tag** – 後続のクエリで使用されるカラムの識別子を定義します。ネスティングの基準 (親に対して) も作成します。
- **name** – (名前／値のペア) オブジェクトの名前を割り当てます。
- **qualifier** – element (デフォルト) にすることも、hide にして結果セットの要素を難読化することもできます。

ANSI SQL – 準拠レベル： Transact-SQL 拡張。

パーミッション

(先頭に戻る) (364 ページ)

なし

FORWARD TO 文

ネイティブな構文をリモートサーバに送信します。

クイックリンク：

「パラメータ」 (370 ページ)

「例」 (371 ページ)

「使用法」 (371 ページ)

「標準」 (371 ページ)

「パーミッション」 (372 ページ)

構文

構文 1

```
FORWARD TO server-name { sql-statement }
```

構文 2

```
FORWARD TO [ server-name ]
```

パラメータ

(先頭に戻る) (370 ページ)

- **server-name** – リモートサーバの名前。
- **sql-statement** – リモートサーバのネイティブ構文によるコマンド。コマンドまたはコマンドグループを中カッコ ({}) または一重引用符で囲みます。
- **server-name** – *server-name* を指定する場合、**FORWARD TO** クエリに文を指定しないと、セッションはパススルーモードに入って、後続のクエリはすべて直接リモートサーバに渡されます。パススルーモードを OFF にするには、*server_name* を指定しないで **FORWARD TO** 文を発行してください。

注意： **FORWARD TO** 文はサーバディレクティブであり、ストアプロシージャ、トリガ、イベント、またはバッチ内では使用できません。

例

(先頭に戻る) (370 ページ)

- **例 1** - リモートサーバ `ase_prod` とのパススルーセッションの例です。

```
FORWARD TO aseprod
SELECT * from titles
SELECT * from authors
FORWARD TO
```

使用法

(先頭に戻る) (370 ページ)

FORWARD TO を使用すると、パススルー接続が必要なサーバを指定できます。この文は、次の目的で使用できます。

- 文をリモートサーバに送信する (構文 1)
- 一連の文をリモートサーバに送信するために、SAP Sybase IQ をパススルーモードに設定する (構文 2)

ユーザの代わりに *server-name* への接続を確立する場合、サーバは次のものを使用します。

- **CREATE EXTERNLOGIN** を使用するリモートログインエイリアスセット
- リモートログインエイリアスを設定しない場合は、SAP Sybase IQ との通信に使用する名前とパスワード

指定のサーバに接続できない場合は、その理由を示すメッセージがユーザに返されます。

文が要求されたサーバに渡されると、結果はすべて、クライアントプログラムで認識できるフォームに変換されます。

関連する動作：

- リモート接続は、**FORWARD TO** セッションの間、**AUTOCOMMIT** (非連鎖) モードに設定されます。**FORWARD TO** 文の前に保留中であった作業はすべて自動的にコミットされます。

標準

(先頭に戻る) (370 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - Open Client/Open Server でサポートされています。

パーミッション

(先頭に戻る) (370 ページ)

なし

参照：

- CREATE EXTERNLOGIN 文 (166 ページ)
- CREATE SERVER 文 (241 ページ)

FROM 句

SELECT 文に必要なデータベーステーブルまたはビューを指定します。

クイックリンク：

「パラメータ」 (374 ページ)

「例」 (378 ページ)

「使用法」 (378 ページ)

「標準」 (380 ページ)

「パーミッション」 (380 ページ)

構文

```
...FROM table-expression [, ...]
```

table-expression - (構文に戻る)

```
table-name  
| view-name  
| procedure-name  
| common-table-expression  
| (subquery) [[ AS ] derived-table-name [ column_name, ... ] ]  
| derived-table  
| join-expression  
| ( table-expression , ... )  
| opening-expression  
| apply-expression  
| contains-expression  
| dml-derived-table
```

table-name - (back to table-expression)

```
[ userid. ] table-name  
[ [ AS ] correlation-name ]  
[ FORCE INDEX ( index-name ) ]
```

view-name - (back to table-expression)

```
[ userid. ] view-name [ [ AS ] correlation-name ]
```



```

procedure-name - (back to table-expression)
  [ owner, ] procedure-name ( [ parameter, ... ] )
  [ WITH ( column-name datatype, ) ]
  [ [ AS ] correlation-name ]

parameter - (back to procedure-name)
  scalar-expression | table-parameter

table-parameter - (back to parameter)
  TABLE ( select-statement ) [ OVER ( table-parameter-over ) ]

table-parameter-over - (back to table-parameter)
  [ PARTITION BY { ANY
  | NONE | table-expression } ]
  [ ORDER BY { expression | integer } ]
  [ ASC | DESC ] [ , ... ]

derived-table - (back to table-expression)
  ( select-statement )
  [ AS ] correlation-name [ ( column-name, ... ) ]

join-expression - (back to table-expression)
  table-expression join-operator table-expression
  [ ON join-condition ]

join-operator - (back to join-expression)
  [ KEY | NATURAL ] [ join-type ] JOIN | CROSS JOIN

join-type - (back to join-operator)
  INNER
  | LEFT [ OUTER ]
  | RIGHT [ OUTER ]
  | FULL [ OUTER ]

openstring-expression - (back to table-expression)
  OPENSTRING ( { FILE | VALUE } string-expression )
  WITH ( rowset-schema )
  [ OPTION ( scan-option ... ) ]
  [ AS ] correlation-name

apply-expression - (back to table-expression)
  table-expression { CROSS | OUTER } APPLY table-expression

contains-expression - (back to table-expression)
  { table-name | view-name } CONTAINS
  ( column-name [ , ... ], contains-query )
  [ [ AS ] score-correlation-name ]

rowset-schema - (back to openstring-expression)
  column-schema-list
  | TABLE [owner.]table-name [ ( column-list ) ]

column-schema-list - (back to rowset-schema)
  { column-name user-or-base-type | filler( ) } [ , ... ]

```

```

column-list - (back to rowset-schema)
  { column-name | filler( ) } [ , ... ]

scan-option - (back to openstring-expression)
  BYTE ORDER MARK { ON | OFF }
  | COMMENTS INTRODUCED BY comment-prefix
  | DELIMITED BY string
  | ENCODING encoding
  | ESCAPE CHARACTER character
  | ESCAPES { ON | OFF }
  | FORMAT { TEXT | BCP }
  | HEXADECIMAL { ON | OFF }
  | QUOTE string
  | QUOTES { ON | OFF }
  | ROW DELIMITED BY string
  | SKIP integer
  | STRIP { ON | OFF | LTRIM | RTRIM | BOTH }

contains-query - (back to contains-expression)
  string

dml-derived-table - (back to table-expression)
  ( dml-statement ) REFERENCING ( [ table-version-names | NONE ] )

dml-statement - (back to dml-derived-table)
  insert-statement
  update-statement
  delete-statement

table-version-names - (back to dml-derived-table)
  OLD [ AS ] correlation-name [ FINAL [ AS ] correlation-name ]
  | FINAL [ AS ] correlation-name

```

パラメータ

(先頭に戻る) (372 ページ)

- **table-name** – ベーステーブルまたはテンポラリテーブル。ユーザ ID を指定すると、他のユーザが所有するテーブルを修飾できます。ユーザ ID を指定しないと、現在のユーザの所属グループが所有するテーブルがデフォルトで検索されます。
- **view-name** – クエリに含めるビューを指定します。テーブルの場合と同様に、ユーザ ID を指定して、他のユーザが所有するビューを修飾できます。ユーザ ID を指定しないと、現在のユーザの所属グループが所有するビューがデフォルトで検索されます。構文ではビューに対してテーブルヒントを指定できますが、このようなヒントの効果はありません。
- **procedure-name** – 結果セットを返すストアードプロシージャ。この句は SELECT 文の FROM 句にのみ適用されます。プロシージャにパラメータを指定しない場

合でも、プロシージャ名の後のカッコは必要です。オプションのパラメータの代わりに DEFAULT を指定できます。

- **parameter** – scalar-parameter または table-parameter 句を指定します。scalar-parameter は有効な SQL データ型の任意のオブジェクトです。table-parameter は、テーブル名、ビュー名、または共通テーブル式名を使用して指定できます。このオブジェクトが table-parameter の外部でも使用されている場合、オブジェクトの新しいインスタンスとして処理されます。

次のクエリは有効な **FROM** 句を示しています。この場合、同一テーブル T に対する 2 つの参照が、同一テーブル T の異なる 2 つのインスタンスとして処理されます。

```
SELECT * FROM T, my_proc(TABLE(SELECT T.Z, T.X FROM T)
OVER(PARTITION BY T.Z));
```

テーブルパラメータ化関数 (TPF) の例 - 次のクエリは有効な **FROM** 句を示しています。

```
SELECT * FROM R, SELECT * FROM my_udf(1);
SELECT * FROM my_tpf(1, TABLE(SELECT c1, c2 FROM t))
(my_proc(R.X, TABLE T OVER PARTITION BY T.X)) AS XX;
```

サブクエリを使用して TABLE パラメータを定義する場合は、次の制限が適用されます。

- table-parameter 句は IN 型でなければなりません。
- PARTITION BY または ORDER BY 句で派生テーブルおよび外部参照のカラムを参照する必要があります。expression-list の式は、TABLE 入力パラメータの K 番目のカラムを参照する整数 K にすることができます。

注意： テーブル UDF は SQL 文の **FROM** 句でのみ参照できます。

- **PARTITION BY** – 実行エンジンによる関数の呼び出しの実行方法を論理的に指定します。実行エンジンは各パーティションに対して関数を呼び出す必要があり、関数は各呼び出しで 1 つのパーティション全体を処理する必要があります。

また、PARTITION BY 句によって、関数の各呼び出しで厳密に 1 つのデータパーティションが処理されるようにするための入力データの分割方法も決まります。パーティション数と同じ数だけ、関数を呼び出す必要があります。TPF の場合、サーバと UDF 間の動的なネゴシエーションを通じて実行時に並列処理特性が設定されます。TPF を N 個の入力パーティションに対して並列実行することが可能な場合、関数を M 回インスタンス化できます ($M \leq N$)。関数の各インスタンス化を複数回呼び出し、各呼び出しで厳密に 1 つのパーティションを使用できます。

PARTITION BY *expression-list* または PARTITION BY ANY 句に対して、TABLE 入力パラメータを 1 つのみ指定できます。他のすべての TABLE 入力パラメー

タに対しては、PARTITION BY NONE 句を明示的または暗黙的に指定する必要があります。

注意： 実行エンジンは任意のパーティション順序で関数を呼び出すことができます。パーティションの順序にかかわらず、同じ結果セットが関数から返されると想定されます。1つのパーティションを関数の2つの呼び出し間で分割することはできません。

- ORDER BY** – 実行エンジンが各パーティションの入力データを *expression-list* を基準にソートすることを指定します。UDF は各パーティションがこの物理プロパティを持っているものと想定します。パーティションが1つしか存在しない場合は、入力データ全体が ORDER BY の指定に基づいてソートされます。ORDER BY 句は、PARTITION BY NONE 句を持つか、または PARTITION BY 句を持たない任意の TABLE 入力パラメータに対して指定できます。
- derived-table** – FROM 句の中で、テーブル名またはビュー名の代わりに SELECT 文を指定できます。このように使用する SELECT 文を派生テーブルと呼び、エイリアスを指定する必要があります。たとえば、次の文には、派生テーブル MyDerivedTable が含まれています。この派生テーブルは、Products テーブル内の製品を UnitPrice でランキングしています。

```
SELECT TOP 3 *
FROM ( SELECT Description,
           Quantity,
           UnitPrice,
           RANK() OVER ( ORDER BY UnitPrice ASC )
           AS Rank
FROM Products ) AS MyDerivedTable
ORDER BY Rank;
```

- join-expression、join-operator、join-type** – join-type キーワードには次のものがあります。

キーワード	説明
CROSS JOIN	2つのソーステーブルの直積(クロス積)を返す
NATURAL JOIN	2つのテーブルの間で、同じ名前を持つすべての対応するカラムの等価性を比較する(等価ジョインの特殊なケースであり、カラムの長さやデータ型は同じ)
KEY JOIN	最初のテーブルの外部キーの値と、2つ目のテーブルのプライマリキーの値が同じであるものに制限する
INNER JOIN	結果テーブルから、両方のテーブルに一致するローを持つロー以外をすべて破棄する
LEFT OUTER JOIN	左側のテーブルからの一致しないローは残すが、右側のテーブルからの一致しないローは破棄する

キーワード	説明
RIGHT OUTER JOIN	右側のテーブルからの一致しないローは残すが、左側のテーブルからの一致しないローは破棄する
FULL OUTER JOIN	左と右の両方のテーブルからの一致しないローを維持する

カンマ形式のジョインとキーワード形式のジョインを **FROM** 句に混在させないでください。同じクエリを、それぞれいずれかの形式を使用した 2 とおりの方法で記述できます。ANSI 構文のキーワード形式の方が望ましいとされています。

次のクエリには、カンマ形式のジョインが使用されています。

```
SELECT *
  FROM Products pr, SalesOrders so, SalesOrderItems si
 WHERE pr.ProductID = so.ProductID
       AND pr.ProductID = si.ProductID;
```

同じクエリを、望ましいとされるキーワード形式のジョインで記述すると次のようになります。

```
SELECT *
  FROM Products pr INNER JOIN SalesOrders so
     ON (pr.ProductID = so.ProductID)
     INNER JOIN SalesOrderItems si
     ON (pr.ProductID = si.ProductID);
```

ON 句は内部ジョイン、左ジョイン、右ジョイン、フルジョインのデータをフィルタします。クロスジョインには ON 句を付けません。内部ジョインでは ON 句は WHERE 句と同じですが、外部ジョインでは ON 句と WHERE 句は違います。外部ジョインの ON 句は、クロス積のローをフィルタし、その結果に NULL で拡張された一致しないローを含めます。その後、WHERE 句によって、外部ジョインで生成された一致するローと一致しないローからローを取り除きます。一致しないローのうち、必要なものまで WHERE 句の述部で取り除いてしまわないように注意が必要です。

外部ジョインの ON 句の中に、サブクエリを使用することはできません。

- **openstring-expression** – OPENSTRING 句は、ファイルまたは BLOB 内を問い合わせ、これらのソースの内容をローセットとして処理するときに指定します。テーブルやビューなどの定義された構造を問い合わせているわけではないため、この句を指定するときは、ファイルまたは BLOB のスキーマ情報も指定して結果セットを生成します。この句は SELECT 文の FROM 句に適用されます。UPDATE 文または DELETE 文ではサポートされません。
- **apply-expression** – この句は、左の table-expression のローごとに、右の table-expression を評価するジョイン条件を指定するときに使用します。たとえば、

適用式を使用して、テーブル式のローごとに関数、プロシージャ、または派生テーブルを評価できます。

- **contains-expression** – テーブル名の後に付ける CONTAINS 句は、テーブルをフィルタして、contains-query で指定した全文クエリに一致するローのみを返すときに使用します。テーブルの一致するすべてのローが、score-correlation-name を使用して参照できるスコアカラムとともに返されます (score-correlation-name を指定している場合)。score-correlation-name を指定しない場合は、デフォルトの相関名 contains によってスコアカラムを参照できます。
- **dml-derived-table** – DML 文 (INSERT、UPDATE、または DELETE) をクエリの FROM 句内のテーブル式として使用することをサポートします。

例

(先頭に戻る) (372 ページ)

- **例 1** – 次は、有効な FROM 句です。

```
...
FROM Employees
...
...
FROM Employees NATURAL JOIN Departments
...
...
FROM Customers
KEY JOIN SalesOrders
KEY JOIN SalesOrderItems
KEY JOIN Products
...
```

- **例 2** – 次のクエリは、クエリ内での派生テーブルの使用法を示しています。

```
SELECT Surname, GivenName, number_of_orders
FROM Customers JOIN
  ( SELECT CustomerID, count(*)
    FROM SalesOrders
    GROUP BY CustomerID )
  AS sales_order_counts ( CustomerID,
                          number_of_orders )
ON ( Customers.ID = sales_order_counts.cust_id )
WHERE number_of_orders > 3
```

使用法

(先頭に戻る) (372 ページ)

SELECT 文には、文で使用するテーブルを指定するためのテーブルリストが必要です。

注意：ここでの説明はテーブルに関するものですが、特に指定がない限りビューにも当てはまります。

FROM テーブルリストでは、指定した全テーブルのすべてのカラムから成る結果セットが作成されます。最初は、コンポーネントテーブルのすべてのローの組み合わせが結果セットに入ります。通常、その後、ジョイン条件、**WHERE** 条件、またはその両方によって組み合わせの数が削減されます。

userid を指定すると、別のユーザが所有するテーブルを修飾できます。ユーザ ID を指定しないと、現在のユーザの所属ロールが所有するテーブルがデフォルトで検索されます。

相関名を使用して、この SQL 文専用の一時的な名前がテーブルに付けられます。カラムを参照するときにテーブル名で修飾されたカラム名を指定する必要がある場合に、テーブル名が長くて入力が煩わしいときに、この相関名を使うと便利です。相関名は、同じクエリで複数回、同じテーブルを参照するときに、テーブルインスタンス間で区別するためにも必要です。相関名を指定しない場合は、テーブル名を現在の文の相関名として使用します。

テーブル式の中で同じテーブルに対して同じ相関名を 2 回使用する場合、そのテーブルは、1 回だけリストされたものとして扱われます。次に例を示します。

```
SELECT *
FROM SalesOrders
KEY JOIN SalesOrderItems,
SalesOrders
KEY JOIN Employees
```

SalesOrders テーブルの 2 つのインスタンスは、1 つのインスタンスとして扱われます。これは、次のものと等しくなります。

```
SELECT *
FROM SalesOrderItems
KEY JOIN SalesOrders
KEY JOIN Employees
```

一方、次のように、異なる相関名 HUSBAND と WIFE を使用している場合は、Person テーブルの 2 つのインスタンスとして扱われます。

```
SELECT *
FROM Person HUSBAND, Person WIFE
```

ジョインカラムのパフォーマンスを高めるために、データ型は似ているものを使用してください。

- **パフォーマンスに関する考慮事項** – クエリにもよりますが、SAP Sybase IQ では、オプティマイザをオンにした状態で 16 ~ 64 個のテーブルを **FROM** 句に指定できます。ただし、非常に複雑なクエリで 16 ~ 18 個よりも多いテーブルを **FROM** 句に指定すると、パフォーマンスが損なわれる可能性があります。

注意： **FROM** 句を省略した場合や、クエリ内のすべてのテーブルが **SYSTEMDB** 領域にある場合、クエリは **SAP Sybase IQ** ではなく **SQL Anywhere** によって処理されます。そのため、特に構文およびセマンティックの制限やオプションの設定方法の違いによって、動作が変わる可能性があります。

FROM 句を必要としないクエリがある場合は、**FROM iq_dummy** 句を追加することによって、**SAP Sybase IQ** でのクエリの処理を強制することができます。この **iq_dummy** は、ユーザがデータベースに作成する 1 ロー、1 カラムのテーブルです。

標準

(先頭に戻る) (372 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - **JOIN** 句は、Adaptive Server の一部のバージョンではサポートされていません。代わりに、**WHERE** 句を使用してジョインを作成してください。

パーミッション

(先頭に戻る) (372 ページ)

データベースに接続されている必要があります。

参照：

- DELETE 文 (309 ページ)
- SELECT 文 (523 ページ)

GET DESCRIPTOR 文 [ESQL]

記述子領域内の変数に関する情報を取り出します。または、記述子領域内の変数から実際のデータを取り出します。

クイックリンク：

「例」 (381 ページ)

「使用法」 (381 ページ)

「標準」 (381 ページ)

「パーミッション」 (381 ページ)

構文

```
GET DESCRIPTOR descriptor-name
  { ...hostvar = COUNT } | VALUE n assignment [, ... ] }
```

assignment

```
hostvar = { TYPE
  | LENGTH
  | PRECISION
  | SCALE
  | DATA
  | INDICATOR
  | NAME
  | NULLABLE
  | RETURNED_LENGTH }
```

例

(先頭に戻る) (380 ページ)

- **例 1** – 例については、「ALLOCATE DESCRIPTOR 文 [ESQL]」を参照してください。

使用法

(先頭に戻る) (380 ページ)

値 *n* には、情報を取り出す記述子領域内の変数を指定します。

GET DESCRIPTOR ... DATA の実行中に型チェックが実行され、ホスト変数と記述子変数のデータ型が同じかどうか確認されます。**GET DESCRIPTOR ... DATA** では、LONG VARCHAR と LONG BINARY がサポートされていません。

エラーが発生した場合は、SQLCA に返されます。

標準

(先頭に戻る) (380 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - Open Client/Open Server でサポートされています。

パーミッション

(先頭に戻る) (380 ページ)

なし

参照：

- ALLOCATE DESCRIPTOR 文 [ESQL] (7 ページ)

- DEALLOCATE DESCRIPTOR 文 [ESQL] (295 ページ)
- SET DESCRIPTOR 文 [ESQL] (540 ページ)

GOTO 文 [T-SQL]

ラベルの付いた文に制御を分岐します。

クイックリンク：

「例」 (382 ページ)

「使用法」 (382 ページ)

「標準」 (382 ページ)

「パーミッション」 (383 ページ)

構文

```
label : GOTO label
```

例

(先頭に戻る) (382 ページ)

- **例 1** – 次の Transact-SQL バッチは、サーバウィンドウに "yes" のメッセージを 4 回表示します。

```
declare @count smallint
select @count = 1
restart:
    print 'yes'
    select @count = @count + 1
    while @count <=4
        goto restart
```

使用法

(先頭に戻る) (382 ページ)

Transact-SQL のプロシージャまたはバッチに含まれる任意の文に、ラベルを付けることができます。ラベル名の末尾にコロン (:) を付けて有効な識別子とします。

GOTO 文では、コロンは使用できません。

標準

(先頭に戻る) (382 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - Adaptive Server では、GOTO 文がサポートされていません。

パーミッション

(先頭に戻る) (382 ページ)

なし

GRANT CHANGE PASSWORD 文

ユーザが、他のユーザのパスワードを管理し、CHANGE PASSWORD システム権限を管理できるようにします。

クイックリンク：

「パラメータ」 (383 ページ)

「例」 (384 ページ)

「使用法」 (384 ページ)

「標準」 (385 ページ)

「パーミッション」 (385 ページ)

構文

```
GRANT CHANGE PASSWORD ( target_user_list | ANY | ANY WITH ROLES
target_role_list )
  TO userID [,...]
  [ WITH ADMIN [ONLY] OPTION | WITH NO ADMIN OPTION]
```

パラメータ

(先頭に戻る) (383 ページ)

- **target_user_list** – 被付与者が同一化の対象とする可能性のあるユーザ。このリストは、ログインパスワードを持つ既存のユーザまたはユーザ拡張ロールで構成される必要があります。リスト内の *userID* はカンマで区切ります。
- **ANY** – ログインパスワードを持つすべてのデータベースユーザが、各被付与者のパスワードを管理する潜在的ターゲットユーザになります。
- **ANY WITH ROLES target_role_list** – 各被付与者のターゲットロールのリスト。いずれかのターゲットロールを付与されたユーザは、各被付与者の潜在的ターゲットユーザになります。 *target_role_list* は既存のロールで構成される必要があります、前述のロールを付与されたユーザはログインパスワードを持つデータ

ベースユーザで構成される必要があります。複数の userID を指定する場合は、カンマで区切ります。

- **userID** – ログインパスワードを持つ既存のユーザまたはロールの名前になります。複数の userID はカンマで区切ります。
- **WITH ADMIN OPTION** – (ANY 句でのみ有効) ユーザは、パスワードを管理でき、別のユーザに CHANGE PASSWORD システム権限を付与することもできます。
- **WITH ADMIN ONLY OPTION** – (ANY 句でのみ有効) ユーザは、別のユーザに CHANGE PASSWORD システム権限を付与できますが、他のユーザのパスワードを管理することはできません。
- **WITH NO ADMIN OPTION** – ユーザはパスワードを管理できますが、別のユーザに CHANGE PASSWORD システム権限を付与することはできません。

例

(先頭に戻る) (383 ページ)

- **例 1** – Sally と Laurel に、Bob、Sam、および Peter のパスワードを管理する権限を付与します。

```
GRANT CHANGE PASSWORD (Bob, Sam, Peter) TO (Sally, Laurel)
```

- **例 2** – データベース内の任意のユーザに CHANGE PASSWORD システム権限を付与する権限を Mary に付与します。ただし、システム権限は WITH ADMIN ONLY OPTION 句付きで付与されるので、Mary は、別のユーザのパスワードを管理できません。

```
GRANT CHANGE PASSWORD (ANY) TO Mary WITH ADMIN ONLY OPTION
```

- **例 3** – Steve と Joe に、Role1 または Role2 のメンバーのパスワードを管理する権限を付与します。

```
GRANT CHANGE PASSWORD (ANY WITH ROLES Role1, Role2) TO Steve, Joe
```

使用法

(先頭に戻る) (383 ページ)

データベース内の任意のユーザ (ANY) または特定のユーザのみ (*target_users_list*) または特定のロールのメンバー (ANY WITH ROLES *target_roles_list*) のパスワードを管理する機能をユーザに付与できます。CHANGE PASSWORD システム権限に対する管理権限は、ANY 句を使用する場合のみ付与できます。

句を指定しない場合、ANY がデフォルトで使用されます。GRANT 文で管理句の指定がない場合、WITH NO ADMIN OPTION 句が使用されます。

デフォルトでは、CHANGE PASSWORD システム権限は、SYS_AUTH_SA_ROLE 互換ロールには WITH NO ADMIN OPTION 句付きで、SYS_AUTH_SSO_ROLE 互換ロールには ADMIN ONLY OPTION 句付きで付与されます (互換ロールが存在する場合)。

標準

(先頭に戻る) (383 ページ)

ANSI SQL – 準拠レベル：Transact-SQL 拡張。

パーミッション

(先頭に戻る) (383 ページ)

- CHANGE PASSWORD システム権限が管理権限付きで付与されている必要があります。
- 指定される各ターゲットユーザ (target_users_list) は、ログインパスワードが設定されている既存のユーザまたはユーザ拡張ロールです。
- 指定された各ターゲットロール (target_roles_list) は、既存のユーザ拡張ロールまたはユーザ定義ロールである必要があります。

参照：

- ALTER USER 文 (92 ページ)
- REVOKE CHANGE PASSWORD 文 (499 ページ)

GRANT CONNECT 文

ユーザに CONNECT 権限を付与します。

クイックリンク：

「パラメータ」 (386 ページ)

「例」 (386 ページ)

「使用法」 (386 ページ)

「標準」 (387 ページ)

「パーミッション」 (387 ページ)

構文

```
GRANT CONNECT
  TO userID [,...]
  IDENTIFIED BY password [,...]
```

パラメータ

(先頭に戻る) (385 ページ)

- **userID** – ログインパスワードを持つ既存のユーザまたはロールの名前になります。複数の userID はカンマで区切ります。

例

(先頭に戻る) (385 ページ)

- **例 1** – データベースに Laurel と Hardy という名前の 2 人の新規ユーザを作成します。

```
GRANT CONNECT TO Laurel, Hardy
IDENTIFIED BY Stan, Ollie
```

- **例 2** – ユーザ Jane をパスワードなしで作成します。

```
GRANT CONNECT TO Jane
```

- **例 3** – Bob のパスワードを newpassword に変更します。

```
GRANT CONNECT TO Bob IDENTIFIED BY newpassword
```

使用法

(先頭に戻る) (385 ページ)

GRANT CONNECT は新規ユーザの作成に使用できます。また、すべてのユーザが自分のパスワードの変更にも使用できます。

ヒント： ユーザを作成するには、**GRANT CONNECT** 文ではなく、**CREATE USER** 文を使用してください。

新しいユーザを追加するときに既存のユーザのユーザ ID を誤って入力すると、その既存ユーザのパスワードを変更することになります。これは正常な動作とみなされるため、警告は発生しません。

sp_addlogin と **sp_adduser** のストアードプロシージャもユーザの追加に使用できます。これらのプロシージャでは、既存のユーザ ID を追加しようとするエラーが表示されます。

注意： ユーザ ID の追加と削除を行うには、**GRANT** 文や **REVOKE** 文ではなくシステムプロシージャを使用してください。

パスワードのないユーザは、データベースに接続できません。これは、ロールユーザ ID への接続をすべて拒否する場合のグループ作成に便利です。パスワードなしでユーザを作成するには、IDENTIFIED BY 句を含めません。

パスワードを指定する際には、有効な識別子である必要があります。パスワードの最大文字長は 255 バイトです。データベースオプション

VERIFY_PASSWORD_FUNCTION に空の文字列以外が設定されている場合、**GRANT CONNECT TO** 文はそのオプションの値で識別される関数を呼び出します。その関数が NULL を返す場合は、パスワードがルールに従っていることを示します。

VERIFY_PASSWORD_FUNCTION オプションが設定されている場合、**GRANT CONNECT** 文には *userid* と *password* をそれぞれ 1 つだけ指定できます。

データベースユーザ ID 名およびパスワードとして無効なものは次のとおりです。

- 空白文字または一重引用符や二重引用符で始まる
- 最後の文字をスペースにする
- セミコロンを含める

標準

(先頭に戻る) (385 ページ)

- SQL – その他の構文は、ISO/ANSI SQL 文法のベンダ拡張です。
- SAP Sybase Database 製品 - セキュリティモデルは Adaptive Server と SAP Sybase IQ では異なるため、他の構文も異なります。

パーミッション

(先頭に戻る) (385 ページ)

- 新しいユーザを作成する場合は、MANAGE ANY USER システム権限が必要です。
- すべてのユーザが自分のパスワードを変更できます。
- 別のユーザのパスワードを変更する場合は、CHANGE PASSWORD システム権限が必要です。

注意：別のユーザのパスワードを変更する場合は、そのユーザがデータベースに接続していないことが必要です。

参照：

- CREATE USER 文 (287 ページ)
- REVOKE CONNECT 文 (501 ページ)

GRANT CREATE 文

指定の DB 領域に対する CREATE 権限を指定のユーザおよびロールに付与します。

クイックリンク：

「パラメータ」 (388 ページ)

「例」 (388 ページ)

「標準」 (388 ページ)

「パーミッション」 (389 ページ)

構文

```
GRANT CREATE
ON dbspace_name
TO userID [,...]
```

パラメータ

(先頭に戻る) (388 ページ)

- **userID** – ログインパスワードを持つ既存のユーザまたはロールの名前になります。複数の userID はカンマで区切ります。

例

(先頭に戻る) (388 ページ)

- **例 1** – ユーザ Lawrence と Swift に DB 領域 *DspHist* に対する CREATE 権限を付与します。

```
GRANT CREATE ON DspHist
TO LAWRENCE, SWIFT
```

- **例 2** – ユーザ Fiona と Ciaran に DB 領域 *DspHist* に対する CREATE 権限を付与します。

```
GRANT CREATE ON DspHist TO Fiona, Ciaran
```

標準

(先頭に戻る) (388 ページ)

- SQL – その他の構文は、ISO/ANSI SQL 文法のベンダ拡張です。

- SAP Sybase Database 製品 - セキュリティモデルは Adaptive Server と SAP Sybase IQ では異なるため、他の構文も異なります。

パーミッション

(先頭に戻る) (388 ページ)

MANAGE ANY DBSPACE システム権限が必要です。

参照：

- REVOKE CREATE 文 (502 ページ)

GRANT オブジェクトレベル権限文

個々のテーブルまたはビューに対するデータベースオブジェクトレベル権限をユーザまたはロールに付与します。

クイックリンク：

「パラメータ」 (389 ページ)

「使用法」 (390 ページ)

「標準」 (390 ページ)

「パーミッション」 (391 ページ)

構文

```
GRANT object-level-privilege [, ...]
ON [ owner.]object-name
TO userID [,...]
[ WITH GRANT OPTION ]

object-level-privilege
ALL [ PRIVILEGES ]
| ALTER
| DELETE
| INSERT
| LOAD
| REFERENCE [ ( column-name [, ...] ) ]
| SELECT [ ( column-name [, ...] ) ]
| TRUNCATE
| UPDATE [ ( column-name, ... ) ] }
```

パラメータ

(先頭に戻る) (389 ページ)

SQL 文

- **userID** – 既存ユーザまたは不変ロールの名前になります。リストは、ログインパスワードを持つ既存ユーザで構成する必要があります。リスト内の userID はカンマで区切ります。
- **ALL** – すべての権限をユーザに付与します。
- **ALTER** – ユーザが **ALTER TABLE** 文を使用してテーブルを変更できます。この権限は、ビューには使用できません。
- **DELETE** – ユーザがテーブルまたはビューからローを削除できます。
- **INSERT** – ユーザが指定のテーブルまたはビューにローを挿入できます。
- **LOAD** – ユーザが指定のテーブルまたはビューにデータをロードできます。
- **REFERENCES** – ユーザが、指定したテーブルのインデックス、および、指定したテーブルを参照する外部キーを作成できます。カラム名を指定した場合は、ユーザは指定したカラムだけを参照できます。カラムの **REFERENCES** 権限は、ビューに対しては付与できません。テーブルに対してのみ付与できます。
- **SELECT** – ユーザがビューまたはテーブルの情報を参照できます。カラム名を指定すると、ユーザは指定したカラムだけを参照できます。カラムの **SELECT** パーミッションは、ビューに対しては付与できません。テーブルに対してのみ付与できます。
- **TRUNCATE** – ユーザが指定のテーブルまたはビューをトランケートできます。
- **UPDATE** – ユーザがビューまたはテーブルのローを更新できます。カラム名を指定すると、ユーザは指定したカラムだけを更新できます。カラムの **UPDATE** 権限は、ビューに対しては付与できません。テーブルに対してのみ付与できます。テーブルを更新するには、ユーザはそのテーブルに対する **SELECT** 権限と **UPDATE** 権限の両方を保有している必要があります。
- **WITH GRANT OPTION** – 指定したユーザ ID に、同じ権限を他のユーザ ID に付与する権限も与えます。

使用法

(先頭に戻る) (389 ページ)

テーブル権限をリストすることも、**ALL** を指定して一度にすべての権限を付与することもできます。

標準

(先頭に戻る) (389 ページ)

- SQL - 構文は初級レベル機能です。
- SAP Sybase Database 製品 - 構文は Adaptive Server でサポートされています。

パーミッション

(先頭に戻る) (389 ページ)

次のいずれかが必要です。

- `MANAGE ANY OBJECT PRIVILEGE` システム権限。
- そのテーブルに対して `WITH GRANT OPTION` 句を指定してオブジェクト権限が付与されている。
- テーブルを所有している。

参照：

- `REVOKE` オブジェクトレベル権限文 (506 ページ)

GRANT EXECUTE 文

プロシージャまたはユーザ定義関数に対する `EXECUTE` 権限を付与します。

クイックリンク：

「パラメータ」 (391 ページ)

「標準」 (391 ページ)

「パーミッション」 (392 ページ)

構文

GRANT EXECUTE

```
ON [ owner. ] { procedure-name | user-defined-function-name }  
TO userID [, ...]
```

パラメータ

(先頭に戻る) (391 ページ)

- **userID** - ログインパスワードを持つ既存のユーザまたはロールの名前になります。複数の `userID` はカンマで区切ります。

標準

(先頭に戻る) (391 ページ)

SQL 文

- SQL – 構文は永続的ストアドモジュール機能です。
- SAP Sybase Database 製品 - セキュリティモデルは Adaptive Server と SAP Sybase IQ では異なるため、他の構文も異なります。

パーミッション

(先頭に戻る) (391 ページ)

次のいずれかが必要です。

- `MANAGE ANY OBJECT PRIVILEGE` システム権限。
- そのプロシージャを所有している。

参照：

- `REVOKE EXECUTE` 文 (503 ページ)

GRANT INTEGRATED LOGIN 文

1つまたは複数の Windows ユーザプロファイルと既存のデータベースユーザ ID の間に、明示的な統合ログインマッピングを作成します。これによって、ローカルマシンにログインできたユーザは、ユーザ ID またはパスワードを入力しなくてもデータベースに接続できます。

クイックリンク：

「パラメータ」 (392 ページ)

「標準」 (392 ページ)

「パーミッション」 (393 ページ)

構文

GRANT INTEGRATED LOGIN

```
TO user_profile_name [, ...]  
AS USER userID [, ...]
```

パラメータ

(先頭に戻る) (392 ページ)

- **userID** – ログインパスワードを持つ既存のユーザまたはロールの名前になります。複数の userID はカンマで区切ります。

標準

(先頭に戻る) (392 ページ)

- SQL – その他の構文は、ISO/ANSI SQL 文法のベンダ拡張です。
- SAP Sybase Database 製品 - セキュリティモデルは Adaptive Server と SAP Sybase IQ では異なるため、他の構文も異なります。

パーミッション

(先頭に戻る) (392 ページ)

MANAGE ANY USER システム権限が必要です。

参照：

- REVOKE INTEGRATED LOGIN 文 (504 ページ)

GRANT KERBEROS LOGIN 文

1 つまたは複数の Kerberos プリンシパルから既存のデータベースユーザ ID への Kerberos 認証ログインマッピングを作成します。これによって、Kerberos に正常にログインしたユーザ (有効な Kerberos チケット付与権限付きチケットを持っているユーザ) は、ユーザ ID またはパスワードを入力しなくてもデータベースに接続できます。

クイックリンク：

「パラメータ」 (393 ページ)

「標準」 (393 ページ)

「パーミッション」 (394 ページ)

構文

GRANT KERBEROS LOGIN

```
TO client-Kerberos-principal [, ...]  
AS USER userID [, ...]
```

パラメータ

(先頭に戻る) (393 ページ)

- **userID** – ログインパスワードを持つ既存のユーザまたはロールの名前になります。複数の userID はカンマで区切ります。

標準

(先頭に戻る) (393 ページ)

SQL 文

- SQL – その他の構文は、ISO/ANSI SQL 文法のベンダ拡張です。
- SAP Sybase Database 製品 - セキュリティモデルは Adaptive Server と SAP Sybase IQ では異なるため、他の構文も異なります。

パーミッション

(先頭に戻る) (393 ページ)

MANAGE ANY USER システム権限が必要です。

参照：

- REVOKE KERBEROS LOGIN 文 (505 ページ)

GRANT ROLE 文

ユーザまたは他のロールに、管理権限付きたまたはなしでロールを付与します。

クイックリンク：

「パラメータ」 (395 ページ)

「例」 (396 ページ)

「使用法」 (396 ページ)

「標準」 (398 ページ)

「パーミッション」 (398 ページ)

構文

```
GRANT ROLE role_name [, ...]
  TO grantee [, ...]
  [ { WITH NO ADMIN | WITH ADMIN [ ONLY ] } OPTION ]
  [ WITH NO SYSTEM PRIVILEGE INHERITANCE ]
```

role_name

```
dbo+++
| diagnostics+++
| PUBLIC+++
| rs_systabgroup+++
| SA_DEBUG+++
| SYS+++
| SYS_AUTH_SA_ROLE
| SYS_AUTH_SSO_ROLE
| SYS_AUTH_DBA_ROLE++
| SYS_AUTH_RESOURCE_ROLE+
| SYS_AUTH_BACKUP_ROLE+
| SYS_AUTH_VALIDATE_ROLE+
| SYS_AUTH_WRITEFILE_ROLE
```

```

| SYS_AUTH_WRITEFILECLIENT_ROLE
| SYS_AUTH_READFILE_ROLE
| SYS_AUTH_READFILECLIENT_ROLE
| SYS_AUTH_PROFILE_ROLE
| SYS_AUTH_USER_ADMIN_ROLE
| SYS_AUTH_SPACE_ADMIN_ROLE
| SYS_AUTH_MULTIPLEX_ADMIN_ROLE
| SYS_AUTH_OPERATOR_ROLE
| SYS_AUTH_PERMS_ADMIN_ROLE
| SYS_REPLICATE_ADMIN_ROLE†††
| SYS_RUN_REPLICATE_ROLE†††
| SYS_SPATIAL_ADMIN_ROLE†††
| user-defined role name

```

- 選択互換ロールを他のロールに付与する際は WITH NO SYSTEM PRIVILEGE INHERITANCE 句を使用できます。この句は、ロールのメンバーによる互換ロールの基礎となるシステム権限の自動継承を禁止します。ユーザ拡張ロールに付与した場合は、WITH NO SYSTEM PRIVILEGE INHERITANCE 句はそのロールのメンバーにのみ適用されます。ロールとして機能するユーザは、この句に関係なく、基礎となるシステム権限を自動的に継承します。
- WITH NO ADMIN OPTION WITH NO SYSTEM PRIVILEGE INHERITANCE 句と WITH NO SYSTEM PRIVILEGE INHERITANCE 句はセマンティック上同じです。
- [†]SYS_AUTH_BACKUP_ROLE ロール、SYS_AUTH_RESOURCE_ROLE ロール、または SYS_AUTH_VALIDATE_ROLE ロールを付与する際に、WITH NO SYSTEM PRIVILEGE INHERITANCE 句と組み合わせて WITH ADMIN OPTION 句または WITH ADMIN ONLY 句を指定することはできません。
- ^{††}SYS_AUTH_DBA_ROLE ロールまたは SYS_RUN_REPLICATION_ROLE ロールを付与する際に、WITH ADMIN OPTION 句は、WITH NO SYSTEM PRIVILEGE INHERITANCE 句と組み合わせてのみ指定できます。
- ^{†††}WITH ADMIN OPTION 句と WITH ADMIN ONLY OPTION 句はシステムロールではサポートされません。

パラメータ

(先頭に戻る) (394 ページ)

- **role_name** – データベースにすでに存在する必要があります。複数のロール名を指定するときはカンマで区切ります。
- **grantee** – ログインパスワードを持つ既存のユーザまたはロールの名前になります。複数の userID はカンマで区切ります。
- **WITH NO ADMIN OPTION** – 各 *grantee* は、各 *role_name* の基礎となるシステム権限を付与されますが、*role_name* を別のユーザに付与することはできません。

- **WITH ADMIN ONLY OPTION** – 各 *userID* に各 *role_name* に対する管理権限は付与されますが、*role_name* の基礎となるシステム権限は付与されません。
- **WITH ADMIN OPTION** – 各 *userID* は、各 *role_name* の基礎となるシステム権限を付与され、加えて、*role_name* を別のユーザに付与する権限も付与されます。
- **WITH NO SYSTEM PRIVILEGE INHERITANCE** – 付与元ロールの基礎となるシステム権限は、付与先ロールのメンバーに継承されません。ただし、付与先ロールがユーザ拡張ロールである場合、その拡張ユーザには基礎となるシステム権限が付与されます。

例

(先頭に戻る) (394 ページ)

- **例 1** – Sales_Role を Sally に管理権限付きで付与します。つまり、このユーザはこのロールにより付与される承認済みタスクを実行できるほか、他のユーザに Sales_Role を付与したり、このロールを取り消したりできます。

```
GRANT ROLE Sales_Role TO Sally WITH ADMIN OPTION
```

- **例 2** – 互換ロール SYS_AUTH_PROFILE_ROLE をロール Sales_Admin に管理権限なしで付与します。Sales_Admin はスタンドアロンロールであり、Mary と Peter にはすでに Sales_Admin が付与されています。SYS_AUTH_PROFILE_ROLE は継承可能な互換ロールなので、Mary と Peter には Sales_Role の基礎となるシステム権限が付与されています。このロールには管理権限が付いていないので、ロールの付与と取り消しはできません。

```
GRANT ROLE SYS_AUTH_PROFILE_ROLE TO Sales_Role WITH NO ADMIN OPTION
```

- **例 3** – 互換ロール SYS_AUTH_BACKUP_ROLE を管理権限なしで Tom に付与します。Tom は、Betty と Laurel に付与されているユーザ拡張ロールです。SYS_AUTH_BACKUP_ROLE は継承不可の互換ロールなので、このロールの基礎となるシステム権限は Betty と Laurel に付与されません。ただし、Tom は拡張ユーザなので、Tom には基礎となるシステム権限が直接付与されます。

```
GRANT ROLE SYS_AUTH_BACKUP_ROLE TO Tom WITH NO SYSTEM PRIVILEGE INHERITANCE
```

使用法

(先頭に戻る) (394 ページ)

WITH ADMIN OPTION 句または WITH ADMIN ONLY OPTION 句を使用することによって、被付与者にロールの付与または取り消しは許可されますが、ロールの削除は許可されません。

GRANT 文で管理句の指定がない場合、デフォルトでは、各互換ロールは次のデフォルト管理権限付きで付与されます。

WITH ADMIN OPTION	WITH ADMIN ONLY OPTION	WITH NO ADMIN OPTION
SYS_AUTH_SA_ROLE SYS_AUTH_SSO_ROLE	SYS_AUTH_DBA_ROLE	SYS_AUTH_RESOURCE_ROLE SYS_AUTH_BACKUP_ROLE SYS_AUTH_VALIDATE_ROLE SYS_AUTH_WRITEFILE_ROLE SYS_AUTH_WRITEFILECLIENT_ROLE SYS_AUTH_READFILE_ROLE SYS_AUTH_READFILECLIENT_ROLE SYS_AUTH_PROFILE_ROLE SYS_AUTH_USER_ADMIN_ROLE SYS_AUTH_SPACE_ADMIN_ROLE SYS_AUTH_MULTIPLEX_ADMIN_ROLE SYS_AUTH_OPERATOR_ROLE SA_DEBUG SYS_RUN_REPLICATION_ROLE

SYS_AUTH_PERMS_ADMIN_ROLE ロールは、次のデフォルト管理権限付きで次の基礎となるロールを付与します。

WITH ADMIN OPTION	WITH NO ADMIN OPTION
SYS_AUTH_BACKUP_ROLE	MANAGE ROLES
SYS_AUTH_OPERATOR_ROLE	MANAGE ANY OBJECT PRIVILEGE
SYS_AUTH_USER_ADMIN_ROLE	CHANGE PASSWORD
SYS_AUTH_SPACE_ADMIN_ROLE	
SYS_AUTH_MULTIPLEX_ADMIN_ROLE	
SYS_AUTH_RESOURCE_ROLE	
SYS_AUTH_VALIDATE_ROLE	
SYS_AUTH_PROFILE_ROLE	
SYS_AUTH_WRITEFILE_ROLE	
SYS_AUTH_WRITEFILECLIENT_ROLE	
SYS_AUTH_READFILE_ROLE	
SYS_AUTH_READFILECLIENT_ROLE	

標準

(先頭に戻る) (394 ページ)

- SQL – その他の構文は、ISO/ANSI SQL 文法のベンダ拡張です。
- SAP Sybase Database 製品 - 構文は Adaptive Server でサポートされています。

パーミッション

(先頭に戻る) (394 ページ)

- 以下のシステムロールを付与するには、MANAGE ROLES システム権限が必要です。
 - dbo
 - diagnostics
 - PUBLIC
 - rs_systabgroup
 - SA_DEBUG SYS
 - SYS
 - SYS_REPLICATION_ADMIN_ROLE
 - SYS_RUN_REPLICATION_ROLE
 - SYS_SPATIAL_ADMIN_ROLE
- 以下のロールを付与するには、そのロールに対する管理権限が必要です。
 - SYS_AUTH_SA_ROLE

- SYS_AUTH_SSO_ROLE
- SYS_AUTH_DBA_ROLE
- SYS_AUTH_RESOURCE_ROLE
- SYS_AUTH_BACKUP_ROLE
- SYS_AUTH_VALIDATE_ROLE
- SYS_AUTH_WRITEFILE_ROLE
- SYS_AUTH_WRITEFILECLIENT_ROLE
- SYS_AUTH_READFILE_ROLE
- SYS_AUTH_READFILECLIENT_ROLE
- SYS_AUTH_PROFILE_ROLE
- SYS_AUTH_USER_ADMIN_ROLE
- SYS_AUTH_SPACE_ADMIN_ROLE
- SYS_AUTH_MULTIPLEX_ADMIN_ROLE
- SYS_AUTH_OPERATOR_ROLE
- SYS_AUTH_PERMS_ADMIN_ROLE
- <ユーザ定義ロール名>

参照：

- CREATE USER 文 (287 ページ)
- REVOKE システム権限文 (512 ページ)
- REVOKE ROLE 文 (508 ページ)
- VERIFY_PASSWORD_FUNCTION オプション (807 ページ)

GRANT SET USER 文

別のユーザに同一化できる権限と、SET USER システム権限を管理する権限を付与します。

クイックリンク：

「パラメータ」 (400 ページ)

「例」 (400 ページ)

「使用法」 (401 ページ)

「標準」 (401 ページ)

「パーミッション」 (401 ページ)

構文

```
GRANT SET USER ( target_users_list
| ANY
```

```

| ANY WITH ROLES target_roles_list )
TO userID [... ]
[ WITH ADMIN [ ONLY ] OPTION | WITH NO ADMIN OPTION ]

```

パラメータ

(先頭に戻る) (399 ページ)

- **target_users_list** – ログインパスワードを持つ既存のユーザで構成される必要があり、被付与者ユーザによる同一化の対象ではなくなった可能性のあるターゲットユーザの潜在的リストです。リスト内のユーザ ID はカンマで区切ります。
- **ANY** – 各被付与者のターゲットユーザの潜在的リストは、ログインパスワードを持つすべてのデータベースユーザで構成されます。
- **ANY WITH ROLES target_roles_list** – *target_role_list* は既存のロールで構成される必要があります。各被付与者のターゲットユーザの潜在的リストは、*target_role_list* 内のロールのサブセットが付与されているログインパスワードを持つデータベースユーザで構成される必要があります。リスト内のロールはカンマで区切ります。
- **userID** – 各 *userID* は、既存のユーザまたは不変ロールの名前になります。リストは、ログインパスワードを持つ既存ユーザで構成される必要があります。リスト内の *userID* はカンマで区切ります。
- **WITH ADMIN OPTION** – (ANY 句を指定する場合のみ有効) ユーザは SETUSER コマンドを発行して別のユーザに同一化することも、SET USER システム権限を別のユーザに付与することもできます。
- **WITH ADMIN ONLY OPTION** – (ANY 句を指定する場合のみ有効) ユーザは SETUSER システム権限を別のユーザに付与することはできませんが、SETUSER コマンドを発行して別のユーザに同一化することはできません。
- **WITH NO ADMIN OPTION** – ユーザは SETUSER コマンドを発行して別のユーザに同一化することはできませんが、SET USER システム権限を別のユーザに付与することはできません。

例

(先頭に戻る) (399 ページ)

- **例 1** – Bob、Sam、および Peter に同一化できる権限を Sally と Laurel に付与します。

```
GRANT SET USER (Bob, Sam, Peter) TO (Sally, Laurel)
```

- **例 2** – データベース内の任意のユーザに SET USER システム権限を付与する権限を Mary に付与します。ただし、このシステム権限は WITH ADMIN ONLY

OPTION 句付きで付与されるので、Mary は、他のユーザに同一化することはできません。

```
GRANT SET USER (ANY) TO Mary WITH ADMIN ONLY OPTION
```

- **例 3** – Role1 または Role2 に同一化できる権限を Steve と Joe に付与します。

```
GRANT SET USER (ANY WITH ROLES Role1, Role2) TO Steve, Joe
```

使用法

(先頭に戻る) (399 ページ)

データベース内の任意のユーザ (ANY) または特定のユーザのみ (*target_users_list*) または特定のロールのメンバー (ANY WITH ROLES *target_roles_list*) に同一化する権限を、ユーザに付与できます。SET USER システム権限に対する管理権限は、ANY 句を使用する場合のみ付与できます。

句を指定しない場合、ANY がデフォルトで使用されます。GRANT 文で管理句の指定がない場合、WITH NO ADMIN OPTION 句が使用されます。

SET USER システム権限をユーザに再付与する場合、付与の効果は累積されます。

デフォルトでは、SET USER システム権限は SYS_AUTH_SSO_ROLE 互換ロールに (存在していれば) WITH NO ADMIN OPTION 句付きで付与されます。

SET USER システム権限の付与のみが、別のユーザへの同一化権限を付与します。別のユーザへの同一化に必要とされる最低条件の検証は、SETUSER 文が発行されるまで発生しません。

標準

(先頭に戻る) (399 ページ)

ANSI SQL – 準拠レベル：Transact-SQL 拡張。

パーミッション

(先頭に戻る) (399 ページ)

- SET USER システム権限が管理権限付きで付与されている必要があります。
- 指定される各ターゲットユーザ (*target_users_list*) は、ログインパスワードが設定されている既存のユーザまたはユーザ拡張ロールです。
- 指定された各ターゲットロール (*target_roles_list*) は、既存のユーザ拡張ロールまたはユーザ定義ロールである必要があります。

参照：

- REVOKE SET USER 文 (510 ページ)

GRANT システム権限文

特定のシステム権限を管理権限付きまたはなしでユーザまたはロールに付与します。

クイックリンク：

「パラメータ」 (402 ページ)

「例」 (403 ページ)

「使用法」 (403 ページ)

「標準」 (403 ページ)

「パーミッション」 (403 ページ)

構文

```
GRANT system_privilege_name [, ...]  
  TO userID [, ...]  
  [ { WITH NO ADMIN | WITH ADMIN [ ONLY ] } OPTION ]
```

パラメータ

(先頭に戻る) (402 ページ)

- **system_privilege_name** – 既存のシステム権限の名前になります。
- **userID** – 既存ユーザまたは不変ロールの名前になります。リストは、ログインパスワードを持つ既存ユーザで構成する必要があります。複数の userID はカンマで区切ります。
- **WITH NO ADMIN OPTION** – ユーザはシステム権限を管理できますが、別のユーザにシステム権限を付与することはできません。
- **WITH ADMIN ONLY OPTION** – WITH ADMIN ONLY OPTION 句が使用される場合、各 userID には、*system_privilege* そのものではなく各 *system_privilege* に対する管理権限が付与されます。
- **WITH ADMIN OPTION** – *userID* のそれぞれに、*system_privilege* の基礎となるシステム権限に加え、それぞれの *system_privilege* に対する管理権限も付与されます。

例

(先頭に戻る) (402 ページ)

- **例 1** – Joe に DROP CONNECTION システム権限を管理権限付きで付与します。

```
GRANT DROP CONNECTION TO Joe WITH ADMIN OPTION
```

- **例 2** – Sally に CHECKPOINT システム権限を管理権限なしで付与します。

```
GRANT CHECKPOINT TO Sally WITH NO ADMIN OPTION
```

- **例 3** – Jane に MONITOR システム権限を管理権限のみで付与します。

```
GRANT MONITOR TO Jane WITH ADMIN ONLY OPTION
```

使用法

(先頭に戻る) (402 ページ)

GRANT 文で管理句の指定がない場合、デフォルトでは WITH NO ADMIN OPTION 句が使用されます。

標準

(先頭に戻る) (402 ページ)

- SQL – その他の構文は、ISO/ANSI SQL 文法のベンダ拡張です。
- SAP Sybase Database 製品 - 構文は Adaptive Server でサポートされています。

パーミッション

(先頭に戻る) (402 ページ)

付与するシステム権限に対する管理権限が必要です。

参照：

- CREATE USER 文 (287 ページ)
- REVOKE システム権限文 (512 ページ)
- REVOKE ROLE 文 (508 ページ)
- VERIFY_PASSWORD_FUNCTION オプション (807 ページ)

すべてのシステム権限のリスト

すべてのシステム権限のリスト。

システム権限は、ユーザが承認済みのデータベースタスクを実行する権限を制御します。

次に、使用可能なシステム権限のリストを示します。

SQL 文

- ACCESS SERVER LS
- ALTER ANY INDEX
- ALTER ANY MATERIALIZED VIEW
- ALTER ANY OBJECT
- ALTER ANY OBJECT OWNER
- ALTER ANY PROCEDURE
- ALTER ANY SEQUENCE
- ALTER ANY TABLE
- ALTER ANY TEXT CONFIGURATION
- ALTER ANY TRIGGER
- ALTER ANY VIEW
- ALTER DATABASE
- ALTER DATATYPE
- BACKUP DATABASE
- CHANGE PASSWORD
- CHECKPOINT
- COMMENT ANY OBJECT
- CREATE ANY INDEX
- CREATE ANY MATERIALIZED VIEW
- CREATE ANY OBJECT
- CREATE ANY PROCEDURE
- CREATE ANY SEQUENCE
- CREATE ANY TABLE
- CREATE ANY TEXT CONFIGURATION
- CREATE ANY TRIGGER
- CREATE ANY VIEW
- CREATE DATATYPE
- CREATE EXTERNAL REFERENCE
- CREATE MATERIALIZED VIEW
- CREATE MESSAGE
- CREATE PROCEDURE
- CREATE PROXY TABLE
- CREATE TABLE
- CREATE TEXT CONFIGURATION
- CREATE VIEW
- DEBUG ANY PROCEDURE
- DELETE ANY TABLE
- DROP ANY INDEX
- DROP ANY MATERIALIZED VIEW

- DROP ANY OBJECT
- DROP ANY PROCEDURE
- DROP ANY SEQUENCE
- DROP ANY TABLE
- DROP ANY TEXT CONFIGURATION
- DROP ANY VIEW
- DROP CONNECTION
- DROP DATATYPE
- DROP MESSAGE
- EXECUTE ANY PROCEDURE
- LOAD ANY TABLE
- INSERT ANY TABLE
- MANAGE ANY DBSPACE
- MANAGE ANY EVENT
- MANAGE ANY EXTERNAL ENVIRONMENT
- MANAGE ANY EXTERNAL OBJECT
- MANAGE ANY LDAP SERVER
- MANAGE ANY LOGIN POLICY
- MANAGE ANY MIRROR SERVER
- MANAGE ANY OBJECT PRIVILEGES
- MANAGE ANY SPATIAL OBJECT
- MANAGE ANY STATISTICS
- MANAGE ANY USER
- MANAGE ANY WEB SERVICE
- MANAGE AUDITING
- MANAGE MULTIPLEX
- MANAGE PROFILING
- MANAGE REPLICATION
- MANAGE ROLES
- MONITOR
- READ CLIENT FILE
- READ FILE
- REORGANIZE ANY OBJECT
- SELECT ANY TABLE
- SERVER OPERATOR
- SET ANY PUBLIC OPTION
- SET ANY SECURITY OPTION
- SET ANY SYSTEM OPTION
- SET ANY USER DEFINED OPTION

SQL 文

- SET USER (管理権限のみで付与)
- TRUNCATE ANY TABLE
- UPDATE ANY TABLE
- UPGRADE ROLE
- USE ANY SEQUENCE
- VALIDATE ANY OBJECT
- WRITE CLIENT FILE
- WRITE FILE

GRANT USAGE ON SEQUENCE 文

指定されたシーケンスに対する USAGE システム権限をユーザまたはロールに付与します。

クイックリンク：

「パラメータ」 (406 ページ)

「標準」 (406 ページ)

「パーミッション」 (406 ページ)

構文

```
GRANT USAGE ON SEQUENCE sequence-name  
TO userID [,...]
```

パラメータ

(先頭に戻る) (406 ページ)

- **userID** - ログインパスワードを持つ既存のユーザまたはロールの名前になります。複数の userID はカンマで区切ります。

標準

(先頭に戻る) (406 ページ)

- SQL - 構文は永続的ストアドモジュール機能です。
- SAP Sybase Database 製品 - セキュリティモデルは Adaptive Server と SAP Sybase IQ では異なるため、他の構文も異なります。

パーミッション

(先頭に戻る) (406 ページ)

次のいずれかが必要です。

- `MANAGE ANY OBJECT PRIVILEGE` システム権限。
- そのシーケンスを所有している。

IF 文

search-condition が TRUE と評価された SQL 文の最初のリストを、条件付きで実行できます。

クイックリンク：

「例」 (407 ページ)

「使用法」 (408 ページ)

「標準」 (408 ページ)

「パーミッション」 (409 ページ)

構文

```
IF search-condition THEN statement-list
... [ ELSEIF search-condition THEN statement-list ]...
... [ ELSE statement-list ]
... END IF
```

例

(先頭に戻る) (407 ページ)

- **例 1** – 次のプロシージャは、IF 文の使用法を示します。

```
CREATE PROCEDURE TopCustomer (OUT TopCompany CHAR(35), OUT
TopValue INT)
BEGIN
    DECLARE err_notfound EXCEPTION
    FOR SQLSTATE '02000' ;
    DECLARE curThisCust CURSOR FOR
    SELECT CompanyName, CAST(
        sum(SalesOrderItems.Quantity *
        Products.UnitPrice) AS INTEGER) VALUE
    FROM Customers
    LEFT OUTER JOIN SalesOrders
    LEFT OUTER JOIN SalesOrsderItems
    LEFT OUTER JOIN Product
    GROUP BY CompanyName ;

    DECLARE ThisValue INT ;
    DECLARE ThisCompany CHAR(35) ;
    SET TopValue = 0 ;
    OPEN curThisCust ;
    CustomerLoop:
    LOOP
```

```

    FETCH NEXT curThisCust
    INTO ThisCompany, ThisValue ;
    IF SQLSTATE = err_notfound THEN
        LEAVE CustomerLoop ;
    END IF ;
    IF ThisValue > TopValue THEN
        SET TopValue = ThisValue ;
        SET TopCompany = ThisCompany ;
    END IF ;
END LOOP CustomerLoop ;
CLOSE curThisCust ;
END

```

- 例 2 – 次のプロシージャは、**ELSEIF** 文の使用法を示します。

```

BEGIN
    DECLARE X INT;
    SET X = 1;
    IF X = 1 THEN
        PRINT '1';
    ELSEIF X = 2 THEN
        PRINT '2';
    ELSE
        PRINT 'something else';
    ENDIF
END

```

使用法

(先頭に戻る) (407 ページ)

いずれの *search-condition* も TRUE と評価されず、ELSE 句が存在する場合は、ELSE 句にある *statement-list* が実行されます。いずれの *search-condition* も TRUE と評価されず、ELSE 句も存在しない場合は、式は NULL 値を返します。

実行は END IF の後に記述されている最初の文から再開されます。

IF 文の内部の **SELECT** 文から返された 1 つの値を変数と比較する場合は、最初に **SELECT** の結果を別の変数に割り当てる必要があります。

注意： IF 文の構文と IF 式の構文を混同しないでください。IF 文をネストすることはできません。

標準

(先頭に戻る) (407 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - Transact-SQL の IF 文では、構文が多少異なります。

パーミッション

(先頭に戻る) (407 ページ)

なし

参照：

- BEGIN ... END 文 (108 ページ)

IF 文 [T-SQL]

SAP Sybase IQ の **IF** 文の代わりとして、Transact-SQL の文を条件付きで実行します。

クイックリンク：

「例」 (409 ページ)

「使用法」 (410 ページ)

「標準」 (410 ページ)

「パーミッション」 (410 ページ)

構文

```

IF expression
  ... statement
  ... [ ELSE [ IF expression ] statement ]...

```

例

(先頭に戻る) (409 ページ)

- **例 1** – Transact-SQL の **IF** 文の使用法を示します。

```

IF (SELECT max(id) FROM sysobjects) < 100
  RETURN
ELSE
  BEGIN
    PRINT 'These are the user-created objects'
    SELECT name, type, id
    FROM sysobjects
    WHERE id < 100
  END

```

- **例 2** – Transact-SQL の **ELSEIF** 文の使用法を示します。

```

BEGIN
  DECLARE @X INT
  SET @X = 1

```

SQL 文

```
IF @X = 1
  PRINT '1'
ELSEIF @X = 2
  PRINT '2'
ELSE
  PRINT 'something else'
END
```

使用法

(先頭に戻る) (409 ページ)

Transact-SQL の **IF** 条件と **ELSE** 条件は、それぞれ単一の SQL 文または複合文 (キーワード **BEGIN** と **END** の間) の実行を制御します。

SAP Sybase IQ の **IF** 文とは異なり、Transact-SQL の **IF** 文には **THEN** がありません。また、Transact-SQL の **IF** 文には **ELSEIF** や **END IF** のキーワードもありません。

IF 文の内部の **SELECT** 文から返された 1 つの値を変数と比較する場合は、最初に **SELECT** の結果を別の変数に割り当てる必要があります。

注意： **IF** 文はネストできません。

標準

(先頭に戻る) (409 ページ)

- SQL – ISO/ANSI SQL 文法の Transact-SQL 拡張。
- SAP Sybase Database 製品 - Adaptive Server では、Transact-SQL の **IF** 文がサポートされています。

パーミッション

(先頭に戻る) (409 ページ)

なし

INCLUDE 文 [ESQL]

SQL ソース言語プリプロセッサによってスキャンされたソースプログラムにファイルをインクルードします。

クイックリンク：

「パラメータ」 (411 ページ)

「使用法」 (411 ページ)

「標準」 (411 ページ)

「パーミッション」(411 ページ)

構文

```
INCLUDE filename
```

パラメータ

(先頭に戻る) (410 ページ)

- **filename** – 識別子

使用法

(先頭に戻る) (410 ページ)

INCLUDE 文は、C プリプロセッサの **#include** 指令と非常によく似ています。

ただし、SQL プリプロセッサは指定のファイルを読み込んで、その内容を出力 C ファイルに挿入します。したがって、インクルードファイルに SQL プリプロセッサが必要とする情報がある場合は、Embedded SQL の **INCLUDE** 文を使用して、その情報をインクルードしてください。

具体的には、SQLCA と SQLDA という 2 つのファイル名が認識されます。Embedded SQL を使用する C プログラムは、すべての Embedded SQL 文の前に次の文を指定する必要があります。

```
EXEC SQL INCLUDE SQLCA;
```

この文を、C 言語プログラムの静的変数宣言を指定できる場所に入れます。多くの Embedded SQL 文では、SQLCA INCLUDE 文の位置に、SQL プリプロセッサによって宣言された変数(プログラマには見えない)が必要です。SQLDA を使用している場合は、SQLDA ファイルをインクルードしてください。

標準

(先頭に戻る) (410 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - Open Client/Open Server でサポートされています。

パーミッション

(先頭に戻る) (410 ページ)

なし

INSERT 文

1つのローあるいは選択されたローを、現在のデータベースの任意の場所からテーブルへ挿入します。この命令では別のデータベースからテーブルへ選択されたローを挿入することができます。

クイックリンク：

「パラメータ」 (413 ページ)

「例」 (415 ページ)

「使用法」 (416 ページ)

「標準」 (420 ページ)

「パーミッション」 (420 ページ)

構文

Syntax 1

```
INSERT [ INTO ] [ owner.]table-name [ ( column-name [, ...] ) ]
    ... VALUES ( [ expression | DEFAULT, ... ] )
or
INSERT [ INTO ] [ owner.]table-name DEFAULT VALUES
```

Syntax 2

```
INSERT [ INTO ] [ owner.]table-name [ ( column-name [, ...] ) ]
    ... insert-load-options insert-select-load-options
    ... select-statement
```

Syntax 3

```
INSERT [ INTO ] [ owner.]table-name [ ( column-name [, ...] ) ]
    ... insert-select-load-options insert-select-load-options
    LOCATION 'servername.dbname'
    [ location-options ]
    ... { { select-statement } | 'select statement' }
```

insert-load-options - (back to Syntax 2) or (back to Syntax 3)

```
[ LIMIT number-of-rows ]
[ NOTIFY number-of-rows ]
[ SKIP number-of-rows ]
```

insert-select-load-options - (back to Syntax 2) or (back to Syntax 3)

```
[ WORD SKIP number ]
[ IGNORE CONSTRAINT constraint-type [, ...] ]
[ MESSAGE LOG 'string' ROW LOG 'string' [ ONLY LOG logwhat [, ...] ] ]
[ LOG DELIMITED BY 'string' ]
```

constraint-type - (back to insert-select-load-options)


```

{ CHECK integer
| UNIQUE integer
| NULL integer
| FOREIGN KEY integer
| DATA VALUE integer
} ALL integer
}

logwhat - (back to insert-select-load-options)
{ CHECK
| ALL
| NULL
| UNIQUE
| DATA VALUE
| FOREIGN KEY
| WORD
}

```

パラメータ

(先頭に戻る) (412 ページ)

- **insert-load-options** – ロードを制限するオプション。
 - **LIMIT** – クエリからテーブルに挿入するローの最大数を指定します。デフォルトは 0 で無制限を意味します。最大値は 2GB -1 です。
 - **NOTIFY** – 一定数のローのテーブルへの挿入が正常に実行されるたびに、メッセージで通知されるように指定します。デフォルトは 100,000 ローごとです。
 - **SKIP** – この挿入時にスキップされる入力テーブルの先頭からのロー数を定義します。デフォルトは 0 です。
- **WORD SKIP** – ワードインデックスの作成時に、指定された制限よりも長いデータが検出されてもロードを続行できます。 *number* パラメータはエラーを無視する回数を指定します。このオプションを 0 に設定すると、無制限になります。

ワードが許可されている最大サイズを超えたためにローがロードされない場合、警告メッセージが *.iqmsg* ファイルに書き込まれます。WORD サイズ違反は、必要に応じて MESSAGE LOG ファイルに記録できます。このオプションを指定しない場合、指定された上限より長いワードの最初の発生時まで処理がロールバックします。

- **IGNORE CONSTRAINT** – ロードエンジンが、ロード時に発生した CHECK、UNIQUE、NULL、DATA VALUE、および FOREIGN KEY の整合性制約違反を無視するかどうか、およびロールバックの開始前に無視される最大違反数を決定します。

limit が 0 の場合、無視される CHECK 制約違反数は無制限になります。CHECK が指定されない場合、CHECK 制約違反の初回発生時にロードがロールバック

されます。*limit* がゼロでない場合、CHECK 制約違反が *limit*+1 回発生すると、ロードがロールバックされます。

- **MESSAGE LOG** – ロードエンジンで整合性制約違反が記録されるファイルの名前を指定します。ロードの開始と完了のタイムスタンプは、MESSAGE LOG と ROW LOG の両ファイルに記録されます。MESSAGE LOG と ROW LOG は両方とも指定する必要があります。指定がないと、整合性違反についての情報はログ記録されません。

ONLY LOG 句に指定されたすべてのタイプの整合性制約の違反についての情報、また、キーワード **WORD** が指定されている場合はすべてのワードインデックス長制限の違反についての情報がログに記録されます。ONLY LOG 句を指定しなかった場合は、整合性制約違反に関する情報はログに記録されません。ロードの開始と完了を示すタイムスタンプのみが記録されます。

- **LOG DELIMITED BY** – ROW LOG ファイルのデータ値を区切るセパレータを指定します。デフォルトのセパレータはカンマです。
- **ENCRYPTED PASSWORD** – リモートサーバへの接続時に Open Client Library のデフォルトパスワードの暗号化を使用するように指定します。このパラメータが指定され、リモートサーバが Open Client Library のデフォルトのパスワード暗号化をサポートしない場合、無効なユーザ ID またはパスワードが使用されたことを示すエラーがレポートされます。

SAP Sybase IQ サーバが、暗号化されたパスワードのある jConnect 接続を受け入れるようにするには、jConnect ENCRYPT_PASSWORD 接続プロパティを true に設定します。

- **PACKETSIZE** – TDS の packet-size をバイト数で指定します。デフォルトの TDS packet-size は、ほとんどのプラットフォームで 512 バイトです。パケットサイズが指定されないか、ゼロに指定された場合は、プラットフォームのデフォルトのパケットサイズ値が使用されます。

packet-size には、512 の倍数で、デフォルトのネットワークパケットサイズと同じ値か、ネットワークパケットサイズのデフォルト値から最大値までの値を指定してください。ネットワークパケットサイズの最大値およびデフォルト値は、512 から 524288 バイトまでの 512 の倍数です。ネットワークパケットサイズの最大値は常に、デフォルトのネットワークパケットサイズ以上になります。

- **QUOTED_IDENTIFIER** – リモートサーバで QUOTED_IDENTIFIER オプションを設定します。デフォルト設定は 'OFF' です。SELECT 文の識別子がすべて二重引用符で囲まれている場合のみ (次の例では 'c1')、QUOTED_IDENTIFIER を 'ON' にします。

```
INSERT INTO foo
LOCATION 'ase.database'
QUOTED_IDENTIFIER ON {select "c1" from xxx};
```

- **ISOLATION LEVEL** – リモートサーバへの接続用の独立性レベルを指定します。

独立性レベル	特性
READ UNCOMMITTED	<ul style="list-style-type: none"> • 独立性レベル 0。 • 書き込みロックの有無にかかわらず、ローで読み込みが許可される。 • 読み込みロックは適用されない。 • 同時トランザクションがローを変更しないこと、またはローに対しての変更がロールバックされないことは保証されない。
READ COMMITTED	<ul style="list-style-type: none"> • 独立性レベル 1。 • 書き込みロックのないローでは読み込みのみ許可される。 • 現在のローでの読み込みに対してのみ読み込みロックが取得されて保持されるが、カーソルがローから移動すると解放される。 • トランザクション中にデータが変更されないという保証はない。
SERIALIZABLE	<ul style="list-style-type: none"> • 独立性レベル 3。 • 書き込みロックのない結果内のローに対しては読み込みのみ許可される。 • カーソルが開いているときに取得された読み込みロックは、トランザクションの終了時まで保持される。

注意： insert-select-load-options と location-options、および constraint-type と logwhat の詳細については、「LOAD TABLE 文」を参照してください。

例

(先頭に戻る) (412 ページ)

- **例 1** – データベースに Eastern Sales 部を追加します。

```
INSERT INTO Departments
(DepartmentID, DepartmentName, DepartmentHeadID)
VALUES (600, 'Eastern Sales', 501)
```

- **例 2** – テーブル dept_head に部長とその部の名称を入力します。

```
INSERT INTO dept_head (name, dept)
NOTIFY 20
```

```
SELECT Surname || ' ' || GivenName
AS name,
dept_name
FROM Employees JOIN Departments
ON EmployeeID= DepartmentHeadID
```

- **例3**– リモートサーバ *detroit* 上の SAP Sybase IQ データベース *iqdet* にある *lineitem* テーブルの *l_shipdate* カラムと *l_orderkey* カラムから、現在のデータベースの *lineitem* テーブルの対応するカラムにデータを挿入します。

```
INSERT INTO lineitem
(l_shipdate, l_orderkey)
LOCATION 'detroit.iqdet'
PACKETSIZE 512
' SELECT l_shipdate, l_orderkey
FROM lineitem '
```

- **例4**– **INSERT** 文により、値のリストが許可され、一度に複数のローを挿入できるようになりました。

```
INSERT into t1 values( 10, 20, 30 ), ( 11, 21, 31 ), ( 12, 22,
32 )
```

使用法

(先頭に戻る) (412 ページ)

構文1を使用すると、指定された式の値で、1つのローを挿入できます。カラム名のリストを指定しないと、作成順 (**SELECT** * を使って取り出すときと同じ順序) に値がテーブルカラムの中に挿入されます。ローは、テーブルの任意の位置に挿入されます。(リレーショナルデータベースでは、テーブルは順序付けられません)。

構文2を使用すると、ユーザは、通常とまったく同じ **SELECT** 文の結果を使用したテーブルへの大量挿入を実行できるようになります。**SELECT** 文に **ORDER BY** 句が含まれていない場合、任意の順序で挿入が行われます。select リストのカラムは、カラムリストに指定されたカラムまたは作成順に並べたカラムと順に一致します。

注意： **NUMBER(*)** 関数は **INSERT** 文の構文2を使用してプライマリーキーを発生する場合に役立ちます。

構文3 **INSERT...LOCATION** は構文2の変形であり、Adaptive Server または SAP Sybase IQ データベースからデータを挿入できます。**LOCATION** 句で指定される *servername.dbname* は、**FROM** 句のテーブルのリモートサーバとデータベースを識別します。構文3を使用するには、接続中の Adaptive Server または SAP Sybase IQ のリモートサーバが、ローカルマシン上の SAP Sybase Open Client の *interfaces* ファイルまたは *sql.ini* ファイルに記述されている必要があります。

構文3を使用したクエリでは、挿入可能なローの数は 2147483647 までです。

SELECT 文を区切るには、中カッコまたは一重引用符を使用できます。

注意：中カッコは、ODBC 標準ではエスケープシーケンスの開始と終了を表すため、ODBC または SAP Control Center のコンテキストではエラーが発生する可能性があります。一重引用符を使用して **SELECT** 文を区切ることにより、エラーを回避できます。

ローカルの SAP Sybase IQ サーバは、**LOCATION** 句で指定したサーバとデータベースに接続します。リモートテーブルに対するクエリの結果が返され、ローカルサーバによって結果が現在のデータベースに挿入されます。**LOCATION** 句でサーバ名を指定しない場合、ローカルサーバにある現在のデータベース以外は選択できなくなるため、SAP Sybase IQ はどのデータベース名が指定されたとしてもこれを無視します。

SAP Sybase IQ がリモートサーバに接続するときに、**CREATE EXTERNLOGIN** でリモートログインが作成されており、**CREATE SERVER** 文でリモートサーバが定義されている場合、**INSERT...LOCATION** は現在の接続のユーザ ID にリモートログインを使用できます。リモートサーバが定義されていないか、現在の接続のユーザ ID に対するリモートログインが作成されていない場合、SAP Sybase IQ は現在の接続のユーザ ID とパスワードを使用して接続します。

注意：現在の接続のユーザ ID とパスワードを使用しているときに、ユーザがパスワードを変更した場合、サーバを停止して再起動するまで、新しいパスワードはリモートサーバ上で有効になりません。**CREATE EXTERNLOGIN** で作成されたリモートログインは、デフォルトユーザ ID のパスワード変更の影響は受けません。

CREATE EXTERNLOGIN 文でリモートログインを作成し、**CREATE SERVER** 文でリモートサーバを定義すると、ユーザがどのような状況でもログインとパスワードを使用できるように、**INSERT...LOCATION** の外部ログインとパスワードが設定されます。これによって、ログインまたはパスワードによるアクセス不能が原因で発生する可能性のあるエラーを回避されます。リモートサーバへの接続はこの方法で行うことをおすすめします。

たとえば、ユーザ russid が SAP Sybase IQ データベースに接続して、次の文を実行します。

```
INSERT local_SQL_Types LOCATION 'ase1.ase1ldb'
{SELECT int_col FROM SQL_Types};
```

サーバ ase1 にはユーザ ID ase1user とそのパスワード sybase があります。テーブル SQL_Types の所有者は ase1user です。リモートサーバは、IQ サーバでは次のように定義されます。

```
CREATE SERVER ase1 CLASS 'ASEJDBC'
USING 'system1:4100';
```

外部ログインは、IQ サーバでは次のように定義されます。

```
CREATE EXTERNLOGIN russid TO asel REMOTE LOGIN aseluser IDENTIFIED BY sybase;
```

INSERT...LOCATION は、ユーザ ID `aseluser` と、ユーザ `russid` のパスワード `sybase` を使用して、リモートサーバ `asel` に接続します。

ENCRYPTED PASSWORD パラメータを使用すると、リモートサーバへの接続時に、Open Client ライブラリによるデフォルトのパスワード暗号化の使用を指定できるようになります。**ENCRYPTED PASSWORD** が指定され、リモートサーバが Open Client Library のデフォルトのパスワード暗号化をサポートしない場合、無効なユーザ ID またはパスワードが使用されたことを示すエラーがレポートされます。

SAP Sybase IQ をリモートサーバとして使用した場合、TDS のパスワード暗号化がサポートされます。SAP Sybase IQ サーバは、クライアントが送信した暗号化されたパスワードのある接続を受け入れます。パスワードの暗号化を設定するための接続プロパティについては、Open Server 15.5 の『Open Client Client-Library/C リファレンスマニュアル』>「Client-Library トピックス」>「セキュリティ機能」>SAP Sybase IQ のセキュリティ機能>「セキュリティハンドシェイク：暗号化されたパスワード」を参照してください。

注意：パスワードの暗号化には Open Client 15.0 が必要です。TDS のパスワード暗号化では、Open Client 15.0 ESD #7 以降が必要です。

INSERT...LOCATION が SAP Sybase IQ サーバとリモートの SAP Sybase IQ サーバまたは Adaptive Server サーバとの間でデータを転送する場合、**PACKETSIZE** に 512 バイト以外の値を指定しても、**INSERT...LOCATION TDS PACKETSIZE** パラメータの値は常に 512 バイトになります。

注意：不正なパケットサイズ (512 の倍数でない 933 など) を指定すると、Open Client `ct_connect` "Connection failed" エラーが表示されて接続の試行が失敗します。接続の試行が失敗すると、必ず "Connection failed" という汎用のメッセージが返されます。Adaptive Server のエラーログを調べれば、接続失敗の原因について、より詳細な情報を入手できます。

SAP Sybase IQ は Adaptive Server のデータ型である TEXT をサポートしませんが、長さが 255 バイトを超える IQ の CHAR カラムまたは VARCHAR カラムから

INSERT...LOCATION (構文 3) を実行できます。また、ASE データベースのデータ型 TEXT のカラムからも実行可能です。ASE の TEXT カラムと IMAGE カラムは、SAP Sybase IQ が内部変換をサポートしていれば、他の SAP Sybase IQ データ型のカラムに挿入できます。デフォルトでは、リモートデータカラムの内容が 2GB を超えている場合、SAP Sybase IQ は暗黙的にカラム値を 2GB にトランケートします。

警告！ SAP Sybase IQ では、Adaptive Server データ型 UNICHAR、UNIVARCHAR、または UNITEXT はサポートされていません。INSERT...LOCATION コマンドを使用して、ISO_BINENG 照合で UNICHAR または UNITEXT のカラムから CHAR または CLOB カラムに変換する操作が、エラーなく実行される場合があります。その場合は、カラム内のデータの整合性が失われている可能性があります。この場合にエラーが報告されるのは、変換に失敗した場合にかぎられます。

非構造化データ分析オプションのラージオブジェクト機能を使用するには、正規のライセンスを取得している必要があります。

注意： INSERT...LOCATION を使用して VARBINARY カラムから選択されたデータを挿入する場合は、リモートデータベースの ASE_BINARY_DISPLAY を OFF に設定してください。

INSERT...LOCATION (構文 3) は SELECT 文での変数の使用をサポートしません。

ビューに挿入できるのは、ビューを定義する SELECT 文が FROM 句の中でテーブルを 1 つだけ持ち、GROUP BY 句と集合関数を含まないか、または UNION 演算を伴わない場合です。

テーブルに挿入される文字列は、データベースで大文字と小文字が区別されるかどうかに関係なく、常に入力された大文字と小文字がそのまま格納されます。したがって、文字列 "Value" をテーブルに挿入すると、V は大文字、残りの英字は小文字で、常にデータベースに格納されます。SELECT 文は、'Value' として文字列を返します。ただし、データベースで大文字と小文字が区別されない場合は、すべての比較で 'Value' は 'value'、'VALUE' などと同じとみなされます。さらに、単一カラムのプライマリーキーにエントリ 'Value' がある場合は、プライマリーキーがユニークでなくなるので、INSERT 文による 'value' の追加は拒否されます。

ユーザが INSERT...LOCATION 文を実行すると、SAP Sybase IQ は言語を判断するために必要なローカライゼーション情報、照合順、文字セット、および日付と時刻の形式をロードします。データベースがプラットフォームのデフォルト以外のロケールを使用している場合は、ローカルクライアントに環境変数を設定して、SAP Sybase IQ が正しい情報をロードするようにしてください。

環境変数 LC_ALL を設定すると、SAP Sybase IQ はその値をロケール名として使用します。LC_ALL が設定されていない場合、SAP Sybase IQ は LANG 環境変数の値を使用します。どちらの環境変数も設定されていない場合、SAP Sybase IQ はロケールファイルにあるデフォルトのエントリを使用します。

(DEFAULT)、DEFAULT VALUES または VALUES () 句を使用して、すべてのデフォルト値を含むローを挿入してください。テーブル t2 に 3 カラムあると仮定すると、これらの例はセマンティック上同義です。

```
INSERT INTO t2 values (DEFAULT, DEFAULT, DEFAULT);
```

```
INSERT INTO t2 DEFAULT VALUES;
```

```
INSERT INTO t2 () VALUES ();
```

INSERT...VALUES は複数のローもサポートします。次の例では、テーブル t1 に 3 つのローを挿入します。

```
CREATE TABLE t1(c1 varchar(30));  
INSERT INTO t1 VALUES ('morning'),('afternoon'),  
('evening');
```

SAP Sybase IQ はすべての load/insert を全幅挿入として扱います。load/insert 文で明示的に指定されないカラムでは、ロードされる値は、カラムの DEFAULT 値（定義されている場合）と NULL（DEFAULT 値がカラムに定義されていない場合）のいずれかとなります。

SAP Sybase IQ は、**INSERT...VALUES**、**INSERT...SELECT**、および **INSERT...LOCATION** のカラムの DEFAULT 値をサポートします。カラムに DEFAULT 値が指定されている場合、カラムの値を指定しないすべての **INSERT** 文（または **LOAD** 文）でカラムの値としてこの DEFAULT 値が使用されます。

プロシージャまたは関数で **COMMIT** 文、**ROLLBACK** 文、または一部の **ROLLBACK TO SAVEPOINT** 文を使用している場合は、ストアードプロシージャまたは関数から **INSERT** を使用することはできません。

SELECT...FROM の結果は、**INSERT...SELECT...FROM** の結果と少し異なる場合があります。これは、DOUBLE や NUMERIC などの不正確なデータ型の内部データ変換によって、挿入時に最適化が行われるためです。より正確な結果が必要な場合は、より高い精度の DOUBLE または NUMERIC のデータ型としてカラムを宣言します。

標準

(先頭に戻る) (412 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - Adaptive Server でサポートされています (*insert-load-options* を除く)。

パーミッション

(先頭に戻る) (412 ページ)

そのテーブルに対する INSERT 権限が必要です。

参照：

- CREATE EXTERNLOGIN 文 (166 ページ)
- DELETE 文 (309 ページ)

- [LOAD TABLE 文 \(430 ページ\)](#)

INSTALL JAVA 文

Java クラスをデータベース内で使用できるようにします。

クイックリンク：

[「パラメータ」 \(421 ページ\)](#)

[「例」 \(422 ページ\)](#)

[「使用法」 \(423 ページ\)](#)

[「標準」 \(423 ページ\)](#)

[「パーミッション」 \(423 ページ\)](#)

構文

```
INSTALL JAVA [ install-mode ] [ JAR jar-name ]  
FROM source
```

install-mode - (構文に戻る)

{ **NEW** | **UPDATE** }

source - (構文に戻る)

{ **FILE** *file-name* | **URL** *url-value* }

パラメータ

(先頭に戻る) [\(421 ページ\)](#)

- **NEW** - (デフォルト) 参照される Java クラスが、現在インストールされているクラスの更新ではなく、新しいクラスである必要があります。データベース内に同じ名前のクラスがすでにある場合、NEW **install-mode** 句を使用すると、エラーが発生します。
- **UPDATE** - このインストールモードを指定すると、指定したデータベースにすでにインストールされている Java クラスの代わりとして使用する Java クラスを、参照される Java クラスに含めることができます。
- **JAR** - 255 バイトまでの文字列値です。後続の **INSTALL** 文、**UPDATE** 文、および **REMOVE** 文で保持された JAR を識別するのに使用します。*jar-name* または *text-pointer* には、JAR ファイル、または JAR を含むカラムを指定する必要があります。JAR ファイルは、通常 **.jar** または **.zip** の拡張子を持ちます。

インストールされた JAR および zip ファイルは、圧縮または展開できます。ただし、Sun JDK **jar** ユーティリティによって作成された JAR ファイルはサポート

されていません。それ以外の zip ユーティリティによって作成されたファイルはサポートされています。

JAR オプションを指定すると、JAR の持つクラスがインストールされた後で JAR として保持されます。この JAR は、これらの各クラスに関連付けられた JAR です。JAR 句によってデータベースにインストールされた JAR のセットは、データベースの保持された JAR と呼ばれます。

保持された JAR は、**INSTALL** 文と **REMOVE** 文で参照されます。保持された JAR は、他の Java-SQL クラスの使用には影響を与えません。SQL システムは、指定されたデータに関連付けられたクラスを他のシステムによって要求された場合に、保持された JAR を使用します。要求されたクラスに、関連付けられている JAR がある場合、SQL システムは、個々のクラスではなく、その JAR を提供します。

- **source** – インストールする Java クラスのロケーションを指定します。クラスファイルまたは JAR ファイルを指定する必要があります。

file-name でサポートされるフォーマットは、'c:¥libs¥jarname.jar' や '/usr/u/libs/jarname.jar' のような完全修飾ファイル名と、データベースサーバの現在の作業ディレクトリを基準にした相対ファイル名です。

各クラスのクラス定義は、最初にそのクラスを使用するときに各接続の VM でロードされます。クラスを **INSTALL** (インストール) すると、接続の VM が暗黙的に再起動されます。このため、**INSTALL** が **NEW** または **UPDATE** のどちらの **install-mode** 句を使用するかに関係なく、すぐに新しいクラスにアクセスできません。

他の接続では、新しいクラスは次回、VM が最初にクラスにアクセスしたときにロードされます。クラスがすでに VM によってロードされている場合、VM がその接続のために (たとえば、**STOP JAVA** や **START JAVA** を使用して) 再起動されるまで、その接続から新しいクラスは見えません。

例

(先頭に戻る) (421 ページ)

- **例 1** – クラスのファイル名とロケーションを指定して、ユーザが作成した "Demo" という名前の Java クラスをインストールします。

```
INSTALL JAVA NEW
FROM FILE 'D:¥JavaClass¥Demo.class'
```

インストール後、クラスはその名前で参照されます。元のファイルパスのロケーションは使用されなくなります。たとえば、次の文は前の文でインストールしたクラスを使用します。

```
CREATE VARIABLE d Demo
```

Demo クラスがパッケージ sybase.work のメンバの場合は、次の例のように、完全に修飾されたクラス名を使用してください。

```
CREATE VARIABLE d sybase.work.Demo
```

- **例 2** – zip ファイルに含まれるすべてのクラスをインストールし、それらを JAR ファイル名でデータベース内に関連付けます。

```
INSTALL JAVA
JAR 'Widgets'
FROM FILE 'C:¥Jars¥Widget.zip'
```

zip ファイルのロケーションは保持されません。クラスは完全に修飾されたクラス名 (パッケージ名とクラス名) を参照しなければなりません。

使用法

(先頭に戻る) (421 ページ)

新しい定義を使用するのは、クラスのインストール後に設定された新しい接続か、クラスのインストール後に初めてそのクラスを使用する接続だけです。Java VM がクラス定義をロードすると、クラス定義は接続が閉じるまでメモリに保存されます。

現在の接続で Java クラスまたはクラスを基にしたオブジェクトを使用している場合、新しいクラス定義を使用するには接続をいったん切断し、その後再接続する必要があります。

標準

(先頭に戻る) (421 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Adaptive Server ではサポートされていません。

パーミッション

(先頭に戻る) (421 ページ)

- `MANAGE ANY EXTERNAL OBJECT` システム権限と、ディスク上のファイルで使用可能な新しいバージョンのコンパイル済みクラスファイルまたは JAR ファイルが必要です。
- すべてのインストールされたクラスは、すべてのユーザがすべての方法で参照できます。

参照：

- `REMOVE` 文 (484 ページ)

IQ UTILITIES 文

バッファキャッシュの統計を収集するキャッシュモニタを起動します。

クイックリンク：

「パラメータ」 (424 ページ)

「例」 (427 ページ)

「使用法」 (427 ページ)

「標準」 (428 ページ)

「パーミッション」 (428 ページ)

構文

```
IQ UTILITIES { MAIN | PRIVATE }
  [ INTO ] table-name
  { START MONITOR ['monitor-options' ]
  | STOP MONITOR }
```

monitor-options

```
{ -summary
| {-append | -truncate } -bufalloc
| -cache
| -cache_by_type
| -contention
| -debug
| -file_suffix suffix
| -io
| -interval seconds
| -threads }...
```

パラメータ

(先頭に戻る) (424 ページ)

- **START MONITOR** – IQ バッファキャッシュモニタを開始します。
- **MAIN** – IQ ストアのメインバッファキャッシュ内のテーブルをすべてモニタリングします。
- **PRIVATE** – テンポラリストアのテンポラリバッファキャッシュ内のテーブルをすべてモニタリングします。
- **dummy_table_name** – SAP Sybase IQ の任意のベーステーブルまたはテンポラリテーブルを指定します。他の **IQ UTILITIES** コマンドと構文上の互換性を持たせ

るために、テーブル名を指定する必要があります。最も望ましいのは、モニタリング専用のテーブルを作成することです。

- **monitor_options** – バッファキャッシュモニタ出力を制御します。複数回指定でき、それらを引用符で囲む必要があります。

オプション	説明
-summary	メインとテンポラリの両方のバッファキャッシュの要約情報を表示。モニタオプションを何も指定しない場合は、サマリレポートを表示。使用法: <code>monitor_options -summary</code>
-cache	メインバッファキャッシュまたはテンポラリバッファキャッシュのアクティビティの詳細を表示。重要なフィールドは Finds、HR%、BWait。使用法: <code>monitor_options -cache</code>
-cache_by_type	-cache の結果を IQ ページタイプごとに集計 ([Bwaits] カラムは例外で、合計だけを表示)。この形式は、保守契約を結んでいるサポートセンタに情報を送る場合に非常に有益。使用法: <code>monitor_options -cache_by_type</code>
-file_suffix	名前 <dbname>.<connid>-<main_or_temp>-<suffix> のモニタリング出力ファイルを作成する。オプションのファイル拡張子を指定しない場合は、デフォルトのファイル拡張子 .iqmon を使用。使用法: <code>monitor_options -file_suffix {extension}</code>
-io	指定した期間のメインまたはテンポラリ (プライベート) のバッファキャッシュの I/O 率と圧縮比を表示。これらのカウンタで、サーバのすべてのアクティビティを表す。情報のデバイス別集計はなし。使用法: <code>monitor_options -io</code>
-bufalloc	ソート、ハッシュ、ビットマップなどのオブジェクト用にバッファキャッシュ内の領域を予約する、メインバッファアロケータまたはテンポラリバッファアロケータの情報を表示。使用法: <code>monitor_options -bufalloc</code>

オプション	説明
-contention	多くの重要なバッファキャッシュとメモリマネージャロックを表示。これらのロックカウンタとミューテックスカウンタは、バッファキャッシュおよびヒープメモリ内のアクティビティと、これらのロックがどれだけ迅速に解消されたかを示す。タイムアウト数が20%を超えている場合は、問題の発生を示す。使用法: <code>monitor_options -contention</code>
-threads	処理スレッドマネージャのカウンタを表示。値はサーバワイド(つまり、メインとプライベートのどちらでこのオプションを選択するかは無関係)。使用法: <code>monitor_options -threads</code>
-interval	レポート間隔を秒単位で指定。デフォルトは60秒ごと。最小値は2秒ごと。 通常、クエリの実行中やパフォーマンスに問題があるときに、モニタをデフォルトの間隔で実行すると、有益な結果の取得が可能。間隔が短すぎると、意味のある結果を取得できないことがある。ジョブ時間に見合った間隔を指定する。通常は1分で十分。使用法: <code>monitor_options -interval</code>
-append -truncate	前者は既存の出力ファイルに追加し、後者は既存の出力ファイルをトランケートする。デフォルトは、トランケート。使用法: <code>monitor_options -append -truncate</code>
-debug	同じ情報を扱う標準表示モードの有無にかかわらず、パフォーマンスモニタで使用可能な情報がすべて表示される。-debugは、主に、保守契約を結んでいるサポートセンタへの情報提供に使用される。使用法: <code>monitor_options -debug</code>

- **STOP MONITOR – START MONITOR** とほぼ同じですが、オプションを指定する必要はありません。

注意：

- モニタを簡単に使用するには、ストアドプロシージャを作成してダミーテーブルを宣言し、出力ロケーションを指定して、モニタを起動します。
- 表示する間隔は、ページ単位ではなく、出力行単位です。ただし、次の2つの場合は例外です。-cache_by_type 句と -debug 句では、表示ごとに新しいページが開始されます。

例

(先頭に戻る) (424 ページ)

- **例 1** - バッファキャッシュモニタを開始し、IQ テンポラリバッファキャッシュのアクティビティを記録します。

```
IQ UTILITIES PRIVATE INTO monitor START MONITOR '-cache -interval
20'
```

使用法

(先頭に戻る) (424 ページ)

各バッファキャッシュをモニタリングするには、別々のコマンドを発行します。モニタが結果を収集している間は、これらの各セッションを開いておきます。接続を閉じると、モニタは実行を停止します。1つの接続で最大で2つのモニタの実行に対応できます。1つはメインバッファキャッシュ用で、もう1つはテンポラリバッファキャッシュ用です。

モニタリング出力ファイルのディレクトリ配置を制御するには、`MONITOR_OUTPUT_DIRECTORY` オプションを設定します。このオプションを設定しない場合は、データベースと同じディレクトリに結果が出力されます。モニタを実行している間、すべてのモニタリング出力ファイルが使用されます。モニタの実行が停止した後も、ファイルはそのまま残ります。

マルチプレックスクエリサーバを作成する前に、モニタリングで使用するテンポラリテーブルを宣言するか、新しいデータベースの作成時に永続的なダミーテーブルを作成してください。これによって DDL の変更を回避し、実際の運用稼働時にデータがクエリサーバに保持されるようにします。

UNIX のようなオペレーティングシステムでは、クエリの実行中にモニタリング出力を確認できます。次に例を示します。

次のコマンドでモニタを起動します。

```
iq utilities main into monitor tab
start monitor "-cache -interval 2 -file_suffix iqmon"
```

出力が `dbname.conn#[main|temp]-iqmon` という名前の ASCII ファイルに送信されます。したがって、`iqdemo` データベースでは、バッファモニタから `iqdemo.2-main-iqmon` に結果が送信されます。

バッファキャッシュモニタは、各実行の結果をこれらのログに書き込みます。

- `dbname.connection#-main-iqmon` //for main buffer cache results
- `dbname.connection#-temp-iqmon` //for temp buffer cache results

プレフィクス `dbname.connection#` は、データベース名と接続番号を示します。複数の接続番号が存在して、自分のものがどれであるかが不明の場合は、カタログ

SQL 文

ストアドプロシージャ **sa_conn_info** を実行します。このプロシージャを実行すると、アクティブなデータベース接続のそれぞれについて、接続番号、ユーザ ID などの情報が表示されます。サフィックス `iqmon` を任意のサフィックスに変更するには、`-file_suffic` 句を使用します。テキストエディタを使用して、ファイルを表示または印刷します。同じデータベースと同じ接続番号からモニタを再実行すると、前回の結果が上書きされます。モニタの実行結果を保存する場合は、ファイルを別の場所にコピーするか、`-append` オプションを使用してください。

標準

(先頭に戻る) (424 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Adaptive Server ではサポートされていません。

パーミッション

(先頭に戻る) (424 ページ)

なし

参照：

- `MONITOR_OUTPUT_DIRECTORY` オプション (725 ページ)

LEAVE 文

複合文または **LOOP** を抜けて実行を続けます。

クイックリンク：

「例」 (428 ページ)

「使用法」 (429 ページ)

「標準」 (429 ページ)

「パーミッション」 (430 ページ)

構文

```
LEAVE statement-label
```

例

(先頭に戻る) (428 ページ)

- **例 1** – 次のコードフラグメントは、**LEAVE** 文を使用してループを抜ける方法を示します。

```
SET i = 1;
lbl:
LOOP
  INSERT
  INTO Counters ( number )
  VALUES ( i ) ;
  IF i >= 10 THEN
    LEAVE lbl ;
  END IF ;
  SET i = i + 1
END LOOP lbl
```

- **例 2** – 次のコードフラグメントは、ネストされたループ内で **LEAVE** を使用します。

```
outer_loop:
LOOP
  SET i = 1;
  inner_loop:
  LOOP
    ...
    SET i = i + 1;
    IF i >= 10 THEN
      LEAVE outer_loop
    END IF
  END LOOP inner_loop
END LOOP outer_loop
```

使用法

(先頭に戻る) (428 ページ)

LEAVE は、指定したラベルの複合文または指定したラベルのループを抜けるための制御文です。複合文またはループの後に記述されている最初の文から実行が再開されます。

プロシージャの本文である複合文は、プロシージャの名前と同じ暗黙のラベルを持っています。

標準

(先頭に戻る) (428 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - Adaptive Server ではサポートされていません。
BREAK 文は、Transact-SQL 互換のプロシージャと同様の機能を提供します。

パーミッション

(先頭に戻る) (428 ページ)

なし

参照：

- BEGIN ... END 文 (108 ページ)
- FOR 文 (361 ページ)
- LOOP 文 (455 ページ)

LOAD TABLE 文

外部ファイルからデータベーステーブルにデータをインポートします。

クイックリンク：

「パラメータ」 (431 ページ)

「例」 (444 ページ)

「使用法」 (447 ページ)

「標準」 (450 ページ)

「パーミッション」 (451 ページ)

構文

```
[ INTO ] TABLE [ owner. ] table-name
... ( load-specification [ , ... ] )
... { FROM | USING [ CLIENT ] FILE }
{ 'filename-string' | filename-variable } [ , ... ]
... [ CHECK CONSTRAINTS { ON | OFF } ]
... [ DEFAULTS { ON | OFF } ]
... [ QUOTES OFF ]
... ESCAPES OFF
... [ FORMAT { ascii | binary | bcp } ]
... [ DELIMITED BY 'string' ]
... [ STRIP { OFF | RTRIM } ]
... [ WITH CHECKPOINT { ON | OFF } ]
... [ BYTE ORDER { NATIVE | HIGH | LOW } ]
... [ LIMIT number-of-rows ]
... [ NOTIFY number-of-rows ]
... [ ON FILE ERROR { ROLLBACK | FINISH | CONTINUE } ]
... [ PREVIEW { ON | OFF } ]
... [ ROW DELIMITED BY 'delimiter-string' ]
... [ SKIP number-of-rows ]
... [ HEADER SKIP number [ HEADER DELIMITED BY 'string' ] ]
... [ WORD SKIP number ]
... [ ON PARTIAL INPUT ROW { ROLLBACK | CONTINUE } ]
```

```

... [ IGNORE CONSTRAINT constraint-type [, ...] ]
... [ MESSAGE LOG 'string' ROW LOG 'string' [ ONLY LOG log-what
[, ...] ]
... [ LOG DELIMITED BY 'string' ]

load-specification - (back to Syntax)
{ column-name [ column-spec ]
  | FILLER ( filler-type ) }

column-spec - (back to load-specification)
{ ASCII ( input-width )
  | BINARY [ WITH NULL BYTE ]
  | PREFIX { 1 | 2 | 4 }
  | 'delimiter-string'
  | DATE ( input-date-format )
  | DATETIME ( input-datetime-format )
  | ENCRYPTED ( data-type 'key-string' [, 'algorithm-string' ] )
  | DEFAULT default-value }
[ NULL ( { BLANKS | ZEROS | 'literal', ... } )

filler-type - (back to load-specification)
{ input-width
  | PREFIX { 1 | 2 | 4 }
  | 'delimiter-string'
  }

constraint-type - (back to Syntax)
{ CHECK integer
  | UNIQUE integer
  | NULL integer
  | FOREIGN KEY integer
  | DATA VALUE integer
  | ALL integer
  }

log-what - (back to Syntax)
{ CHECK
  | ALL
  | NULL
  | UNIQUE
  | DATA VALUE
  | FOREIGN KEY
  | WORD
  }

```

パラメータ

(先頭に戻る) (430 ページ)

- **FROM** – データのロード元のファイルを1つ以上指定します。複数のファイルを指定する場合は、各 *filename-string* をカンマで区切ります。*filename-string* は、文字列としてサーバに渡されます。このため、文字列は他の SQL 文字列と同じフォーマット要件に従います。

Windows のディレクトリパスを示すには、円記号 (¥) を 2 つの円記号で表します。したがって、ファイル `c:¥temp¥input.dat` から `Employees` テーブルにデータをロードする文は、次のようになります。

```
LOAD TABLE Employees
FROM 'c:¥¥temp¥¥input.dat' ...
```

パス名は、クライアントアプリケーションではなくデータベースサーバを基準にした相対パスを指定します。別のコンピュータのデータベースサーバ上で文を実行している場合、ディレクトリ名で参照されるのは、クライアントマシンのディレクトリではなく、そのサーバマシンのディレクトリです。マルチプレックスデータベースをロードする場合、すべてのファイル名に絶対パス (完全修飾パス) を使用します。相対パス名は使用しないでください。

リソースの制約があるため、SAP Sybase IQ ではすべてのデータがロードされる保証はありません。リソースの割り付けに失敗した場合、そのロードトランザクション全体がロールバックされます。ファイルは 1 つずつ読み込まれ、FROM 句で指定された順に処理されます。SKIP または LIMIT の値は、ロードの開始時に適用され、各ファイルには適用されません。

LOAD TABLE FROM 句は今後廃止されますが、サーバ上に存在するファイルの指定に使用できます。次に、クライアントコンピュータ上の `a.inp` ファイルからデータをロードする例を示します。

```
LOAD TABLE t1(c1,c2,filler(30))
USING CLIENT FILE 'c:¥¥client-data¥¥a.inp'
QUOTES OFF ESCAPES OFF
IGNORE CONSTRAINT UNIQUE 0, NULL 0
MESSAGE LOG 'c:¥¥client-data¥¥m.log'
ROW LOG 'c:¥¥client-data¥¥r.log'
ONLY LOG UNIQUE
```

- USING** – USING FILE は、サーバから 1 つまたは複数のファイルをロードします。この句は、FROM *filename* 句を指定する場合と同義です。USING CLIENT FILE は、クライアントから 1 つまたは複数のファイルのバルクロードを行います。クライアント側のファイルの文字セットは、サーバ照合と同じである必要があります。SAP Sybase IQ はファイルリスト内のファイルを逐次処理します。各ファイルは処理時に読み込みモードでロックされ、その後で、ロックが解除されます。クライアント側のバルクロードでは、余分なディスク領域、メモリ、ネットワークモニタリングデーモン要件などの管理作業にかかるオーバーヘッドは発生しません。ただし、ファイルごとの単一スレッド処理が強制的に実行されます。

ラージオブジェクトのバルクロードを行う場合は、USING CLIENT FILE 句をプライマリファイルとセカンダリファイルの両方に適用します。

LOAD TABLE 文では、gzip 形式でのみ、圧縮されたクライアントファイルとサーバファイルをロードできます。拡張子 `.gz` または `.gzip` のファイルはす

べて圧縮ファイルとみなされます。圧縮ファイルのロード時に、名前付きパ
イプまたはセカンダリファイルはサポートされません。同一の **LOAD TABLE** 文
で圧縮ファイルと圧縮解除ファイルを指定することができます。1 回のロード
では、各圧縮ファイルが 1 スレッドで処理されます。

クライアント側ロード中に、**IGNORE CONSTRAINT** ログファイルがクライア
ントホスト上で作成されます。ログファイルの作成中にエラーが発生すると、
操作がロールバックします。

クライアント側バルクロードは、Command Sequence プロトコルを使用する
Interactive SQL クライアントと ODBC/JDBC クライアントによってサポートさ
れています。TDS プロトコルを使用するクライアントではサポートされていま
せん。ネットワーク上のデータセキュリティを確保するには、トランスポート
レイヤセキュリティを使用します。クライアント側バルクロードを使用できる
ユーザを制御するには、セキュア機能 (-sf) サーバ起動スイッチ、
ALLOW_READ_CLIENT_FILE データベースオプション、**READCLIENTFILE** アク
セス制御を使用します。

- **CHECK CONSTRAINTS** – 検査制約を評価します。ユーザはこれを無視するこ
とも、ログに記録することもできます。**CHECK CONSTRAINTS** はデフォルト
で ON に設定されます。

CHECK CONSTRAINTS OFF に設定すると、SAP Sybase IQ はすべての検査制約
違反を無視します。これは、たとえばデータベースの再構築時などに便利で
す。テーブルにユーザ定義関数を呼び出す検査制約があり、その関数がまだ作
成されていない場合、このオプションを **OFF** に設定しなければ再構築が失敗し
ます。

このオプションは、次のオプションに対して相互に排他的です。これらのオブ
ションのいずれかが同じロードに指定されていると、エラーが発生します。

- **IGNORE CONSTRAINT ALL**
- **IGNORE CONSTRAINT CHECK**
- **LOG ALL**
- **LOG CHECK**
- **DEFAULTS** – カラムのデフォルト値を使用します。このオプションはデフォ
ルトで ON です。**DEFAULTS** オプションが **OFF** の場合は、カラムリストに存在し
ないカラムに **NULL** が割り当てられます。

DEFAULTS オプションの設定は、**AUTOINCREMENT** を含むすべてのカラムの
DEFAULT 値に適用されます。

- **QUOTES** – 入力文字列が引用符文字で囲まれることを指定します。**QUOTES** は
オプションのパラメータで、デフォルトでは ON です。引用符文字はアポスト
ロフィ (一重引用符) または二重引用符のいずれかです。文字列内で初めて出現

した場合、これらの文字はその文字列の引用符文字列として処理されます。文字列データは、対応する引用符文字で終わっている必要があります。

QUOTES ON の場合、カラムデリミタ文字またはローデリミタ文字をカラム値の一部とすることができます。開始と終端の引用符文字は、値の一部とはみなされず、ロードされるデータ値から取り除かれます。

QUOTES ON が指定されている値に引用符文字を含めるには、2つの引用符を使用します。たとえば、次の行では3番目のカラムの値に一重引用符文字が含まれています。

```
'123 High Street, Anytown', '(715)398-2354', ''''
```

STRIP を ON (デフォルト) に指定すると、後続空白を削除してから値が挿入されます。後続空白が削除されるのは、引用符で囲まれていない文字列だけです。引用符で囲まれた文字列では、後続空白が保持されます。先行空白または TAB 文字は、この設定が ON の場合にのみ削除されます。

データ抽出機能には、引用符を処理するオプション (**TEMP_EXTRACT_QUOTES**、**TEMP_EXTRACT_QUOTES_ALL**、および **TEMP_EXTRACT_QUOTE**) があります。IQ メインストアテーブルにロードされるデータを抽出する場合に、デフォルトの ASCII 抽出で文字列フィールドにカラムデリミタまたはローデリミタが含まれているときは、抽出に **TEMP_EXTRACT_BINARY** オプションを使用し、**LOAD TABLE** に **FORMAT binary** オプションと **QUOTES OFF** オプションを使用します。

制限事項：

- QUOTES ON は、カラムデリミタがある ASCII フィールドにのみ適用されません。
- QUOTES ON の場合、カラムデリミタまたはローターミネータの最初の文字に一重引用符および二重引用符は使用できません。
- QUOTES ON では、指定のファイルの単一スレッド処理が強制的に実行されます。
- QUOTES オプションは、その設定に関係なく、セカンダリファイルからのバイナリラージオブジェクト (BLOB) データまたはキャラクタラージオブジェクト (CLOB) データのロードに適用されません。開始引用符または終了引用符は、CLOB データの一部としてロードされます。引用符で囲まれている2つの連続した引用符は、QUOTES ON オプションを使用すると2つの連続した引用符としてロードされます。
- Adaptive Server BCP は、QUOTES オプションをサポートしていません。フィールドデータはすべて、QUOTES OFF 設定の場合と同様にコピーされます。QUOTES ON は SAP Sybase IQ **LOAD TABLE** 文のデフォルト設定であ

るため、BCP 出力から SAP Sybase IQ テーブルに ASE データをインポートする場合は QUOTES OFF を指定する必要があります。

例外:

- **LOAD TABLE** で、引用符で囲まれたフィールドの終了引用符文字の後に空白でない文字がある場合、次のエラーがレポートされてロード操作はロールバックされます。

```
Non-SPACE text found after ending quote character for
an enclosed field.
```

```
SQLSTATE: QTA14      SQLCODE: -1005014L
```

- **QUOTES ON** で、カラムデリミタの最初の文字として一重引用符または二重引用符が指定された場合、次のエラーがレポートされてロード操作は失敗します。

```
Single or double quote mark cannot be the 1st character
of column delimiter or row terminator with QUOTES option
ON.
```

```
SQLSTATE: QCA90      SQLCODE: -1013090L
```

- **ESCAPES** – ESCAPES が ON (デフォルト) の場合に、入力フィールドの *column-spec* 定義を省略すると、データベースサーバは円記号に続く文字を特殊文字として認識、解釈します。改行文字は ¥n という組み合わせとして、他の文字はタブ文字の ¥x09 のような 16 進の ASCII コードとしてデータに含めることができます。連続した 2 つの円記号 (¥¥) は 1 つの円記号として解釈されます。SAP Sybase IQ では、ESCAPES を OFF に設定する必要があります。
- **FORMAT** – SAP Sybase IQ は ASCII とバイナリの入力フィールドをサポートします。フォーマットは通常、上記の *column-spec* で定義します。カラムに対してこの定義を省略した場合、SAP Sybase IQ はデフォルトとしてこのオプションで定義したフォーマットを使用します。入力行は、ascii (デフォルト) フィールドまたは **binary** フィールドを持ち、1 行あたり 1 ローで構成され、カラムデリミタ文字で値が区切られているものとみなされます。

SAP Sybase IQ は、**LOAD TABLE** コマンドへの入力として BCP 文字ファイルのデータも受け入れます。

- **LOAD TABLE FORMAT BCP** 文を使用して SAP Sybase IQ テーブルにロードされる BCP データファイルは、-c オプションを使用して、プラットフォームを問わないファイルフォーマットでエクスポート (BCP OUT) する必要があります。
- **FORMAT BCP** の場合、**LOAD TABLE** 文のデフォルトのカラムデリミタは <tab> で、デフォルトのローターミネータは <newline> です。

- **FORMAT BCP** では、ローの最後のカラムはカラムデリミタではなくローターミネータで終了します。カラムデリミタがローターミネータの前にある場合、データの一部として扱われます。
- ロード指定で最後のカラムではないカラムのデータは、カラムデリミタだけを使用して区切ります。最後のカラムではないカラムで、ローターミネータがカラムデリミタの前にある場合、そのローターミネータはカラムデータの一部として扱われます。
- カラムデリミタは **DELIMITED BY** 句で指定できます。**FORMAT BCP** の場合、デリミタの長さは 10 文字以下である必要があります。デリミタの長さが 10 を超える場合はエラーが返されます。
- **FORMAT BCP** では、ロード指定にカラム名、**NULL**、**ENCRYPTED** のみを含めることができます。ロード指定にこれ以外のオプションが指定された場合は、エラーが返されます。

たとえば、次の **LOAD TABLE** ロード指定は有効です。

```
LOAD TABLE x( c1, c2 null(blanks), c3 )
FROM 'bcp_file.bcp'
FORMAT BCP
...
```

```
LOAD TABLE x( c1 encrypted(bigint,'KEY-ONE','aes'), c2, c3 )
FROM 'bcp_file.bcp'
FORMAT BCP
...
```

- **DELIMITED BY** – *column-spec* 定義でカラムデリミタを省略した場合は、デフォルトのカラムデリミタ文字であるカンマが使用されます。1 文字の ASCII 文字、または 16 進の文字表現を入力することにより、別のカラムデリミタを指定できます。**DELIMITED BY** 句は次のようになります。

```
... DELIMITED BY '¥x09' ...
```

デリミタとして改行文字を使用する場合は、特殊文字の組み合わせである '¥n' またはその ASCII 値である '¥x0a' を指定できます。*column-spec* の *delimiter-string* には最大 4 文字まで指定できますが、**DELIMITED BY** 句では 1 文字しか指定できません。

- **STRIP** – 引用符で囲まれていない値を挿入する前に、その値の後続空白を削除するかどうかを指定します。**LOAD TABLE** コマンドでは、次の **STRIP** キーワードを指定できます。
 - **STRIP OFF** – 後続空白を削除しません。
 - **STRIP RTRIM** – 後続空白を削除します。
 - **STRIP ON** – 非推奨。STRIP RTRIM を使用してください。

STRIP を **ON** (デフォルト) に指定すると、**SAP Sybase IQ** は後続ブランクを削除してから値を挿入します。これは **VARCHAR** データの場合にのみ有効です。**STRIP OFF** では後続ブランクは保持されます。

後続ブランクが削除されるのは、引用符で囲まれていない文字列だけです。引用符で囲まれた文字列では、後続ブランクが保持されます。ブランクを区別する必要がない場合は、後続スペースをすべて削除する代わりに、**FILLER** オプションを使用して、削除するバイト数をより詳細に指定できます。**SAP Sybase IQ** では、**STRIP OFF** はさらに効率的で、後続ブランクの処理は **ANSI** 規格に準拠します (**CHAR** データでは、常にブランクが埋め込まれます。したがって、この **STRIP** オプションが有効なのは、**VARCHAR** データの場合だけです)。

STRIP オプションは、可変長の非バイナリデータにのみ適用され、**ASCII** 固定幅の挿入には適用されません。たとえば、次のようなスキーマがあるとします。

```
CREATE TABLE t( c1 VARCHAR(3) );
LOAD TABLE t( c1 ',' ) ..... STRIP RTRIM    // trailing blanks
trimmed

LOAD TABLE t( c1 ',' ) ..... STRIP OFF      // trailing blanks
not trimmed

LOAD TABLE t( c1 ASCII(3) ) ... STRIP RTRIM // trailing blanks
not trimmed
LOAD TABLE t( c1 ASCII(3) ) ... STRIP OFF   // trailing blanks
trimmed

LOAD TABLE t( c1 BINARY ) ..... STRIP RTRIM // trailing blanks
trimmed
LOAD TABLE t( c1 BINARY ) ..... STRIP OFF   // trailing blanks
trimmed
```

バイナリデータの後続ブランクは常に削除されます。

- **WITH CHECKPOINT** – **SAP Sybase IQ** がチェックポイントを実行するかどうかを決定します。このオプションは、**SAP Sybase IQ** データベースの **SQL Anywhere** テーブルをロードする場合にのみ役立ちます。

デフォルト設定は **OFF** です。この句を **ON** に設定すると、文が正常に完了し、ロギングされた後にチェックポイントが発行されます。接続がコミットを実行してから次のチェックポイントを実行するまでにサーバに障害が発生した場合、リカバリを正常に完了するには、テーブルのロードに使用されたデータファイルが存在する必要があります。しかし、**WITH CHECKPOINT ON** を指定した後にリカバリが必要になった場合は、リカバリ時にデータファイルは必要ありません。

データベースが破損したため、バックアップを使用して現在のログファイルを適用する必要がある場合は、この句の指定内容にかかわらず、データファイルが必要となります。

警告！ データベースオプション `CONVERSION_ERROR` を `OFF` に設定すると、エラーがレポートされることなく不良データがテーブルにロードされることがあります。WITH CHECKPOINT ON を指定しない場合、データベースのリカバリが必要となったときには、`CONVERSION_ERROR` が `ON` (デフォルト値) であれば、リカバリが失敗することがあります。`CONVERSION_ERROR` を `OFF` に設定し、WITH CHECKPOINT ON が指定されていないときは、テーブルをロードしないことをおすすめします。

詳細については、「`CONVERSION_ERROR` オプション [TSQL]」を参照してください。

- **BYTE ORDER** – 読み込み時のバイトの順序を指定します。このオプションはすべてのバイナリ入力フィールドに適用されます。何も定義されなければ、このオプションは無視されます。SAP Sybase IQ は通常、それ自体が稼働しているコンピュータのネイティブフォーマットでバイナリデータを読み込みます (デフォルトは `NATIVE` です)。次のように指定することもできます。
 - **HIGH**: マルチバイトの値が上位バイト優先である場合に指定します (Sun、IBM AIX、HP などのビッグエンディアンプラットフォーム用)。
 - **LOW**: マルチバイトの値が下位バイト優先である場合に指定します (Windows などのリトルエンディアンプラットフォーム用)。
- **LIMIT** – テーブルに挿入するローの最大数を指定します。デフォルトは 0 で無制限を意味します。ロー数の最大値は $2^{31} - 1$ (2147483647) です。
- **NOTIFY** – 指定した個数のローがテーブルに正常に挿入されるたびに、メッセージが通知されるように指定します。デフォルトは 0 (通知の出力なし) です。このオプションの値は、`NOTIFY_MODULUS` データベースオプションの値を上書きします。
- **ON FILE ERROR** – 入力ファイルが存在しないか、またはファイルを読み込むパーミッションが不適切であるために、ファイルを開くことができない場合の SAP Sybase IQ の動作を指定します。次のいずれかを指定できます。
 - **ROLLBACK** – トランザクション全体をアボートします (デフォルト)。
 - **FINISH** –すでに完了している挿入処理を完了して、ロード処理を終了します。
 - **CONTINUE** – エラーを返しますが、該当するファイルのみをスキップしてロード処理を継続します。

ON FILE ERROR 句は 1 つしか使用できません。

- **PREVIEW** – 各カラムの開始位置、名前、データ型など、ロード先テーブルへの入力のレイアウト情報を表示します。SAP Sybase IQ はロード処理の開始時にこの情報を表示します。ログファイルにログを書き込んでいる場合は、この情報もログに書き込みます。
- **ROW DELIMITED BY delimiter-string** – 入力レコードの末尾を指定する文字列を最大 4 バイト長で指定します。このオプションが使用できるのは、ロー内の全フィールドが次のいずれかである場合に限られます。
 - カラムターミネータで区切られている場合
 - DATE または DATETIME の *column-spec* オプションでデータが定義されている場合
 - ASCII 固定長フィールド

並列ロードを確実にを行うためには、常に ROW DELIMITED BY を使用します。LOAD 指定でこの句を省略すると、SAP Sybase IQ はロードを並列実行ではなく逐次実行します。

入力フィールドにバイナリデータが格納されている場合は、このオプションは使用できません。このオプションを指定すると、ローターミネータにより、不足したフィールドが NULL に設定されます。すべてのローに同じローデリミタがあり、すべてのカラムデリミタと区別する必要があります。ローデリミタ文字列およびフィールドデリミタ文字列に、互いの初期サブセットを指定することはできません。たとえば、フィールドデリミタとして "*" を、ローデリミタとして "##" を指定することはできませんが、フィールドデリミタとして "#" を、ローデリミタとして "*" を指定することはできます。

ローにデリミタがない場合は、SAP Sybase IQ はエラーを返し、ロードトランザクション全体をロールバックします。唯一の例外はファイルの最終レコードです。この場合、そのローがロールバックされて、警告メッセージが返されます。Windows では通常、ローデリミタは改行文字とそれに続く復帰文字によって指定されます。このオプションまたは FILLER では、これらの文字を、上記で説明されている *delimiter-string* として指定しなければならない場合があります。

- **SKIP** – このロードで入力テーブルの開始時にスキップするローの数を定義します。スキップするローの最大数は $2^{31} - 1$ (2147483647) です。デフォルトは 0 です。SKIP は、スキップするローを読み込むときに単一スレッドモードで動作します。
- **HEADER SKIP...HEADER DELIMITED BY – LOAD TABLE** でスキップする、ヘッダローを含むデータファイルの先頭行数を指定します。指定した数のローがスキップされるまで、すべての **LOAD TABLE** カラム指定と他のロードオプションは無視されます。

- スキップする行数には、0 以上の数を指定する必要があります。
- 行は、**HEADER DELIMITED BY** 句で指定した 1～4 文字のデリミタ文字列で区切られます。デフォルトの **HEADER DELIMITED BY** 文字列は、'¥n' 文字です。
- **HEADER DELIMITED BY** 文字列の最大長は 4 文字です。文字列の長さが 5 文字以上または 1 未満の場合は、エラーが返されます。
- ゼロ以外の **HEADER SKIP** 値を指定すると、**HEADER DELIMITED BY** デリミタを含むすべてのデータが、**HEADER SKIP** 句で指定した数のデリミタが検出されるまで無視されます。
- 指定されている数のローがスキップされるまで、すべての **LOAD TABLE** カラム指定と他のロードオプションは無視されます。指定されている数のローがスキップされると、**LOAD TABLE** カラム指定と他のロードオプションが残りのデータに適用されます。
- "ヘッダ" バイトは、データの開始位置でのみ無視されます。USING 句で複数のファイルが指定されていると、後続のファイルに同様のローが含まれている場合でも、**HEADER SKIP** は、指定されている数のヘッダローがスキップされるまで、最初のファイルの最初のローから始まるデータだけを無視します。**LOAD TABLE** は、実際のデータの解析を開始すると、ヘッダを検索しません。
- **HEADER SKIP** で指定されている数のローをスキップする前に、**LOAD TABLE** がすべての入力データを処理しても、エラーはレポートされません。
- **WORD SKIP** – ワードインデックスの作成時に、指定された制限よりも長いデータが検出されてもロードを続行できます。

ワードが許可されている最大サイズを超えたためにローがロードされない場合、警告メッセージが .iqmsg ファイルに書き込まれます。**WORD** サイズ制限の違反は、オプションで **MESSAGE LOG** ファイルに記録され、拒否されたローは、**ROW LOG** ファイルに記録されます。これらのファイルは **LOAD TABLE** 文で指定されます。

 - このオプションが指定されていない場合、**LOAD TABLE** は、指定された制限を超えた最初のワードでエラーをレポートしてロールバックします。
 - *number* は、「最大許容単語長を超える単語はサポートされません。」のエラーを無視する回数を指定します。
 - 0 (ゼロ) は制限がないことを意味します。
- **ON PARTIAL INPUT ROW** – ロード中に部分入力ローがあった場合のアクションを指定します。次のいずれかを指定できます。
 - **CONTINUE** は、警告を発行し、ロード操作を続行します。これはデフォルトです。

- ROLLBACK は、ロード操作全体をアボートし、エラーをレポートします。

```
Partial input record skipped at EOF.
SQLSTATE: QDC32      SQLSTATE: -1000232L
```

- **IGNORE CONSTRAINT** – ロード中に発生した CHECK、UNIQUE、NULL、DATA VALUE、または FOREIGN KEY 整合性制約違反を無視するかどうか、また、違反をいくつ無視してからロールバックを開始するかを決める違反の最大数を指定します。 *constrainttype* の指定に応じて、次のような動作となります。
 - **CHECK limit** – *limit* に 0 を指定すると、CHECK 制約違反は無制限に無視されます。CHECK が指定されなければ、最初に CHECK 制約違反が発生した時点で **LOAD** 文がロールバックされます。 *limit* が 0 でなければ、CHECK 制約違反が *limit*+1 回発生した時点でロードがロールバックされます。
 - **UNIQUE limit** – *limit* に 0 を指定すると、UNIQUE 制約違反は無制限に無視されます。 *limit* が 0 でなければ、UNIQUE 制約違反が *limit*+1 回発生した時点でロードがロールバックされます。
 - **NULL limit** – *limit* に 0 を指定すると、NULL 制約違反は無制限に無視されます。 *limit* が 0 でなければ、NULL 制約違反が *limit*+1 回発生した時点でロードがロールバックされます。
 - **FOREIGN KEY limit** – *limit* に 0 を指定すると、FOREIGN KEY 制約違反は無制限に無視されます。 *limit* が 0 でなければ、FOREIGN KEY 制約違反が *limit*+1 回発生した時点でロードがロールバックされます。
 - **DATA VALUE limit** – データベースオプションに `CONVERSION_ERROR=ON` を指定すると、エラーがレポートされて文がロールバックされます。 *limit* に 0 を指定した場合、DATA VALUE 制約違反 (データ型変換エラー) は無制限に無視されます。 *limit* が 0 でなければ、DATA VALUE 制約違反が *limit*+1 回発生した時点でロードがロールバックされます。
 - **ALL limit** – データベースオプションに `CONVERSION_ERROR = ON` が指定された場合、エラーがレポートされて文がロールバックされます。 *limit* に 0 を指定した場合、すべての整合性制約違反は無制限に無視されます。 *limit* が 0 以外の場合は、UNIQUE、NULL、DATA VALUE、FOREIGN KEY の整合性制約違反を無視した数の累計が *limit* の値を超えた時点で、ロードはロールバックされます。たとえば、次の IGNORE CONSTRAINT オプションを指定したとします。

```
IGNORE CONSTRAINT NULL 50, UNIQUE 100, ALL 200
```

この場合、整合性制約違反の合計数は 200 を超えることができません。同時に、NULL 制約違反は 50、UNIQUE 制約違反は 100 を超えてはなりません。これらのいずれかの制限を超えた時点で、LOAD TABLE 文はロールバックされます。

注意：1つのローに、整合性制約違反が複数ある場合もあります。それぞれの整合性制約違反が、当該の種類違反としてカウントされ、件数が制限に近づきます。

無視された整合性制約違反をロギングする場合は、`IGNORE CONSTRAINT` オプションの `limit` を 0 以外の値に設定します。ロギングする違反の数が多すぎると、ロードのパフォーマンスに影響します。

`IGNORE CONSTRAINT` 句に `CHECK`、`UNIQUE`、`NULL`、または `FOREIGN KEY` を指定しない場合、これらのいずれかのタイプの整合性制約違反が最初に起きた時点でロードがロールバックします。

`IGNORE CONSTRAINT` 句で `DATA VALUE` が指定されていない場合は、このタイプの整合性制約違反が最初に見つかった時点で、ロードはロールバックされます。ただし、データベースオプション `CONVERSION_ERROR = OFF` が設定されている場合は除きます。`CONVERSION_ERROR = OFF` が指定されていると、すべての `DATA VALUE` 制約違反に警告がレポートされ、ロードが継続されます。

ロードが完了すると、整合性制約違反に関する情報メッセージが `.iqmsg` ファイルに記録されます。このメッセージには、ロード中に発生した整合性制約違反の数と、スキップされたローの数が含まれます。

- **MESSAGE LOG** – 整合性制約違反に関する情報を記録するログファイルの名前と、ログに記録する違反のタイプを指定します。ロードの開始と完了を示すタイムスタンプは、`MESSAGE LOG` ファイルと `ROW LOG` ファイルの両方に記録されます。`MESSAGE LOG` と `ROW LOG` は両方とも指定する必要があります。指定がないと、整合性違反についての情報はログ記録されません。
 - `ONLY LOG` 句を指定しなかった場合は、整合性制約違反に関する情報はログに記録されません。ロードの開始と完了を示すタイムスタンプのみが記録されます。
 - `ONLY LOG` 句に指定されたすべてのタイプの整合性制約の違反についての情報、また、キーワード `WORD` が指定されている場合はすべてのワードインデックス長制限の違反についての情報がログに記録されます。
 - 制約違反がログに記録される場合、整合性制約違反が発生するたびに、`MESSAGE LOG` ファイルに必ず 1 行の情報が生成されます。
`MESSAGE LOG` ファイルのローの数 (レポートされたエラーの数) が、`IGNORE CONSTRAINT` オプションの制限を超えることがあります。並行して動作する複数のスレッドによってロードが実行されるためです。制約違反の数が指定された制限を超えたことを、複数のスレッドがレポートする場合もあります。

- 制約違反がログに記録される場合、特定のローに関する情報は、(そのローで整合性制約違反がいくつ発生しようと) ROW LOG ファイルに必ず 1 行で記録されます。
MESSAGE LOG ファイルに記録された個々のエラーの数と、ROW LOG ファイルに記録されたローの数が一致しない場合があります。両者のロー数の差異は MESSAGE LOG の項で説明した、ロードの並行処理によるものです。
- MESSAGE LOG ファイルと ROW LOG ファイルには、ローパーティションや名前付きパイプは使用できません。
- MESSAGE LOG ファイルまたは ROW LOG ファイルがすでに存在する場合、新しい情報がそのファイルに追加されます。
- MESSAGE LOG ファイルまたは ROW LOG ファイルに無効なファイル名を指定すると、エラーが発生します。
- MESSAGE LOG ファイルと ROW LOG ファイルに同じファイル名を指定すると、エラーが発生します。

IGNORE CONSTRAINT オプションと MESSAGE LOG オプションをさまざまに組み合わせて指定すると、さまざまなロギングアクションが実行されます。

表 5 : LOAD TABLE のロギングアクション

IGNORE CONSTRAINT の指定の有無	MESSAGE LOG の指定の有無	アクション
あり	あり	無視されたすべての整合性制約違反が、ロールバックが発生するまでのユーザ指定の limit も含めて記録される。
なし	あり	ロールバックが発生する前の最初の整合性制約違反が記録される。
あり	なし	何もロギングされない。
なし	なし	何もロギングされない。最初の整合性制約違反でロールバックが実行される。

ヒント： 無視された整合性制約違反を記録する場合は、IGNORE CONSTRAINT オプションの limit を 0 以外の値に設定します。1 つのローに複数の整合性制約違反がある場合、MESSAGE LOG ファイルには、各違反がそれぞれ別個のローとして記録されます。ロギングする違反の数が多すぎると、ロードのパフォーマンスに影響します。

- **LOG DELIMITED BY** – ROW LOG ファイルのデータ値を区切るセパレータを指定します。デフォルトのセパレータはカンマです。

SAP Sybase IQ では、FORMAT BCP が **LOAD TABLE** 句として指定された場合でも、エラーメッセージを返さなくなりました。また、次の状態が確認され、対応するエラーメッセージが返されます。

- 指定されたロード形式が ASCII、BINARY、または BCP のいずれでもない場合、SAP Sybase IQ はメッセージ「LOAD のフォーマットとしてサポートされているのは ASCII、BCP および BINARY のみです。」を返します。
- LOAD TABLE** のカラム指定にカラム名、NULL、または ENCRYPTED 以外のものが含まれている場合、SAP Sybase IQ はエラーメッセージ「LOAD ... FORMAT BCP のロードの指定は無効です。」を返します。
- FORMAT BCP ロードのカラムデリミタまたはローターミネータのサイズが 10 文字を超えた場合、SAP Sybase IQ はメッセージ「デリミタ '%2' の長さは 1 から %3 文字にする必要があります。」を返します(ここで、%3 には 10 が入ります)。

FORMAT BCP と FORMAT ASCII で発生する可能性があるエラーまたは警告の状態に対応するメッセージは、どちらのフォーマットでも同じです。

- 指定されるロードデフォルト値が AUTOINCREMENT、IDENTITY、または GLOBAL AUTOINCREMENT の場合、SAP Sybase IQ はエラー「デフォルト値 %2 は LOAD のデフォルト値として使用できません。%1」を返します。
- LOAD TABLE** 指定に、指定されたファイルからロードする必要があるカラムが含まれていない場合、SAP Sybase IQ がエラー「LOAD 文には入力ファイルからロードされるカラムを少なくとも 1 つ含める必要があります。」をレポートし、**LOAD TABLE** 文がロールバックします。
- ロード時に TEXT インデックスがあるテキストドキュメントで単語の数が最大数を超えると、SAP Sybase IQ から次のエラーが返されます。「テキストドキュメントが語数の最大値を超えています。サポートされているドキュメント当たりの最大語数は 4294967295 です。」

例

(先頭に戻る) (430 ページ)

- 例 1** - 1 つのファイルのデータを Windows システム上の Products テーブルにロードします。タブは、Description カラムと Color カラムの後に続くカラムデリミタとして使用されます。

```
LOAD TABLE Products
( ID ASCII(6),
  FILLER(1),
  Name  ASCII(15),
  FILLER(1),
  Description  '¥x09',
  Size  ASCII(2),
```



```

FILLER(1),
Color   '¥x09',
Quantity PREFIX 2,
UnitPrice PREFIX 2,
FILLER(2) )
FROM 'C:¥¥mydata¥¥source1.dmp'
QUOTES OFF
ESCAPES OFF
BYTE ORDER LOW
NOTIFY 1000

```

- **例2**–クライアントコンピュータにあるファイル a.inp からデータをロードします。

```

LOAD TABLE t1(c1,c2,filler(30))
USING CLIENT FILE 'c:¥¥client-data¥¥a.inp'
QUOTES OFF ESCAPES OFF
IGNORE CONSTRAINT UNIQUE 0, NULL 0
MESSAGE LOG 'c:¥¥client-data¥¥m.log'
ROW LOG 'c:¥¥client-data¥¥r.log'ONLY LOG UNIQUE

```

- **例3**–2つのファイルからのデータをUNIXシステム上の product_new テーブル (NULL 値使用可) にロードします。タブ文字がデフォルトカラムデリミタで、改行文字がローデリミタです。

```

LOAD TABLE product_new
( id,
name,
description,
size,
color   '¥x09'   NULL( 'null', 'none', 'na' ),
quantity PREFIX 2,
unit_price PREFIX 2 )
FROM '/s1/mydata/source2.dump',
'/s1/mydata/source3.dump'
QUOTES OFF
ESCAPES OFF
FORMAT ascii
DELIMITED BY '¥x09'
ON FILE ERROR CONTINUE
ROW DELIMITED BY '¥n'

```

- **例4**–ワード長の制限違反を10回は無視し、11回目に新しいエラーを表示してロードをロールバックします。

```

load table PTAB1(
    ck1      ',,' null ('NULL') ,
    ck3fk2c2 ',,' null ('NULL') ,
    ck4      ',,' null ('NULL') ,
    ck5      ',,' null ('NULL') ,
    ck6c1    ',,' null ('NULL') ,
    ck6c2    ',,' null ('NULL') ,
    rid      ',,' null ('NULL') )
FROM 'ri_index_selfRI.inp'
row delimited by '¥n'

```

```
LIMIT 14 SKIP 10
IGNORE CONSTRAINT UNIQUE 2, FOREIGN KEY 8
word skip 10 quotes off escapes off strip
off
```

- **例 5 – FORMAT BCP** ロードオプションを使用して、テーブル t1 を **BCP** 文字ファイル bcp_file.bcp からテーブルにロードします。

```
LOAD TABLE t1 (c1, c2, c3)
FROM 'bcp_file.bcp'
FORMAT BCP
...
```

- **例 6 – DEFAULT** ロードオプションを使用して、デフォルト値 12345 を c1 にロードして、c2 および c3 をデータとともに LoadConst04.dat ファイルからロードします。

```
LOAD TABLE t1 (c1 DEFAULT '12345 ', c2, c3, filler(1))
FROM 'LoadConst04.dat'
STRIP OFF
QUOTES OFF
ESCAPES OFF
DELIMITED BY ',';
```

- **例 7 – FORMAT BCP** ロードオプションを使用して、c1 および c2 をデータとともにファイル bcp_file.bcp からロードし、c3 を値 10 に設定します。

```
LOAD TABLE t1 (c1, c2, c3 DEFAULT '10')
FROM 'bcp_file.bcp'
FORMAT BCP
QUOTES OFF
ESCAPES OFF;
```

- **例 8** – 次のコードフラグメントは、データファイルの先頭のヘッダローを 1 行無視します。ヘッダローは「&&」で区切られています。

```
LOAD TABLE
...HEADER SKIP 1 HEADER DELIMITED by '&&'
```

- **例 9** – 次のコードフラグメントは、データファイルの先頭のヘッダローを 2 行無視します。ヘッダ行のそれぞれは、「¥n」によって区切られています。

```
LOAD TABLE
...HEADER SKIP 2
```

- **例 10** – ファイルを RLV 対応テーブルにロードします。

FORMAT BCP ロードオプションを使用して、**BCP** 文字ファイル bcp_file.bcp から RLV 対応テーブル rvt1 にデータをロードします。

```
LOAD TABLE rvt1 (c1, c2, c3)
FROM 'bcp_file.bcp'
FORMAT BCP
...
```

使用法

(先頭に戻る) (430 ページ)

LOAD TABLE 文を使用すると、ASCII またはバイナリデータのファイルからデータベーステーブルに大量の挿入を効率よく行うことができます。

LOAD TABLE では、整合性制約違反が発生し、違反に関する情報を記録する場合のロードの動作を制御することもできます。

LOAD TABLE はテンポラリテーブルに対して使用できますが、テンポラリテーブルを **ON COMMIT PRESERVE ROWS** で宣言している必要があります。宣言されていないと、次の **COMMIT** 時にロードしたローが削除されます。

LOAD TABLE は、ラージオブジェクト (LOB) データのロードをサポートしていません。

SAP Sybase IQ は、ASCII データとバイナリデータの両方からのロードをサポートし、また、固定長と可変長の両方のフォーマットをサポートしています。これらのフォーマットのすべてを処理するには、*load-specification* を指定して、ソースファイルの各「カラム」またはフィールドからロードされるデータの種類を SAP Sybase IQ に指示する必要があります。*column-spec* を使用すると、次のフォーマットを定義できます。

- 固定長バイトの ASCII。*input-width* 値は、各レコードの入力フィールドの固定幅のバイト数を示す整数値です。
- 入力の長さの指定に **PREFIX** のバイト数 (1、2、または 4) を使用するバイナリフィールドと非バイナリフィールド。

PREFIX 句に関連する部分は、次の 2 つです。

- Prefix 値 - 常にバイナリ値。
- 関連付けられるデータバイト - 常に文字フォーマット。バイナリフォーマットは使用されません。

TEMP_EXTRACT_BINARY オプションを **ON** に設定し、抽出機能を使用してデータがアンロードされる場合は、バイナリデータのロード時に **BINARY WITH NULL BYTE** パラメータを各カラムに使用する必要があります。

- セパレータで区切られた可変長文字列。ターミネータを 16 進の ASCII 文字として指定できます。*delimiter-string* には、4 文字までの任意の文字列を使用できます。これには任意に組み合わせた印刷可能文字、および印刷されない文字を表す任意の 8 ビットの 16 進 ASCII コードなどが含まれます。たとえば、次のように指定します。
 - ターミネータとなるタブを表す '¥x09'。
 - null ターミネータ '¥x00' ("C" の文字列と同様に表示されないターミネータ)。

- ターミネータとなる改行文字の '¥x0a'。特殊文字の組み合わせである '¥n' を使用して改行を表すこともできます。

注意：デリミタ文字列の長さには 1～4 文字を指定できますが、**DELIMITED BY** 句には 1 文字しか指定できません。**BCP** では、10 文字までのデリミタが使用できます。

- ASCII 文字としての DATE または DATETIME 文字列。SAP Sybase IQ でサポートされる日付または日付時刻データ型に対応するフォーマットのいずれかを使用して、文字列の *input-date-format* または *input-datetime-format* を定義する必要があります。日付値には **DATE** を、日付/時刻、および時刻の値には **DATETIME** を使用します。

表 6：日付と時刻のフォーマット

オプション	意味
yyyy または YYYY yy または YY	年を表す。デフォルトは現在の年。
mm または MM	月を表す。5月には「05」というように、必要に応じて必ず先行0またはブラックを使用する。DATE 値には月を含める必要がある。たとえば、 DATE 値として「1998」と入力すると、エラーになる。「03」と入力した場合、SAP Sybase IQ はデフォルトの年と日付を適用して、「1998-03-01」に変換する。
dd または DD jjj または JJJ	日付を表す。デフォルトの日付は 01。1桁の日付は、日付の先頭に0を付ける。たとえば、最初の日は「01」となる。Jまたはjは年のユリウス日付(1～366)を示す。
hh HH	時間を示す。時間は24時間表記に基づく。1桁の時刻は、その前に0または空白を付ける。たとえば、午前1時は「01」となる。「00」は有効な値で、午前0時(深夜)を表す。
nn	分を示す。1桁の分は、分の前に0を付ける。たとえば、8分は「08」となる。
ss[.sssss]	秒数とコンマ以下の秒数を示す。
aa	午前または午後を示す。
pp	必要な場合にのみ、午後を示す(このオプションはバージョン12.0より前のSAP Sybase IQとは互換性がない。以前は「pp」と「aa」は同義であった)。
hh	SAP Sybase IQ は、分と秒をゼロとみなす。たとえば、入力した DATETIME 値が「03」の場合、SAP Sybase IQ では、「03:00:00.0000」に変換される。

オプション	意味
hh:nn または hh:mm	SAP Sybase IQ は秒をゼロとみなす。たとえば、入力した時刻値が「03:25」の場合、SAP Sybase IQ では「03:25:00.0000」に変換される。

表 7 : DATE と DATETIME フォーマットのオプションの例

入力データ	フォーマット仕様
12/31/98	DATE ('MM/DD/YY')
19981231	DATE ('YYYYMMDD')
123198140150	DATETIME ('MMDDYYhhnnss')
14:01:50 12-31-98	DATETIME ('hh:nn:ss MM-DD-YY')
18:27:53	DATETIME ('hh:nn:ss')
12/31/98 02:01:50AM	DATETIME ('MM/DD/YY hh:nn:ssaa')

SAP Sybase IQ には、一般的な日付、時刻および日付時刻フォーマット用のロードの最適化が装備されています。ロードするデータがこれらのフォーマットのいずれかに当てはまる場合は、正しいフォーマットを使用することによってロード時間を大幅に削減できます。

date/time フィールドを ASCII 固定幅フィールド (上記参照) として指定し、FILLER(1) オプションを使用してカラムデリミタをスキップすることもできます。

column-spec の NULL の部分は、テーブルのカラムにデータをロードするときに、特定の入力値を NULL として処理する方法を示します。NULL として処理される文字には、BLANKS、ZEROS、または定義したその他のリテラルのリストなどがあります。NULL 値を指定するか、またはソースファイルから NULL 値を読み込む場合は、ロード先カラムに NULL を格納する必要があります。

ZEROS は次のように解釈されます。入力データ (ASCII の場合は変換前) がすべてバイナリのゼロ (文字のゼロではない) の場合にかぎり、セルは NULL に設定されます。

- 入力データが文字のゼロの場合は、次のようになります。
 1. NULL (ZEROS) を指定しても、セルに NULL がセットされることはない。
 2. NULL ('0') を指定すると、セルに NULL がセットされる。
- 入力データがバイナリのゼロ (全ビットがクリア) の場合は、次のようになります。
 1. NULL (ZEROS) を指定すると、セルに NULL がセットされる。
 2. NULL ('0') を指定しても、セルに NULL がセットされることはない。

たとえば、**LOAD** 文に `col1 date('yymmdd') null(zeros)` が含まれ、日付が 000000 であると、000000 を DATE(4) に変換できないことを示すエラーが表示され

ます。データが 000000 である場合に、**LOAD TABLE** で NULL 値が col1 に挿入されるようにするには、NULL 句を null('000000') のように記述するか、データを同等のバイナリのゼロに修正して NULL(ZEROS) を使用します。

VARCHAR セルの長さがゼロで、セルが NULL 以外の場合は、長さゼロのセルが作成されます。その他のデータ型については、セルの長さがゼロの場合、SAP Sybase IQ によって NULL が挿入されます。これは ANSI 準拠の動作です。ANSI 以外で長さゼロの文字データを処理するには、**NON_ANSI_NULL_VARCHAR** データベースオプションを設定します。

DEFAULT オプションを使用して、デフォルトのロードカラム値を指定します。カラムにテーブルスキーマで定義されたデフォルト値がない場合でも、デフォルト値をカラムにロードできるようになります。この機能によって、ロード時の柔軟性が向上します。

- **LOAD TABLE** 文で指定されたデフォルト値を使用するには、**LOAD TABLE DEFAULTS** オプションを ON にする必要があります。**DEFAULTS** オプションが OFF の場合、指定されたロードデフォルト値は使用されず、代わりに NULL 値がカラムに挿入されます。
- **LOAD TABLE** コマンドには、**LOAD TABLE** コマンドで指定されたファイルからロードする必要があるカラムが 1 つ以上含まれている必要があります。そうしないと、エラーが報告され、ロードは実行されません。
- 指定されたデフォルトのロード値は、カラムのサポート対象のデフォルト値、およびデフォルト値の制約と一致している必要があります。**LOAD TABLE DEFAULT** オプションでは、デフォルトのロード値として **AUTOINCREMENT**、**IDENTITY**、および **GLOBAL AUTOINCREMENT** はサポートされません。
- **LOAD TABLE DEFAULT default-value** は、データベースと同じ文字セットである必要があります。
- **LOAD TABLE DEFAULT** 句で指定されたロードデフォルト値では、デフォルト値の暗号化はサポートされていません。
- 指定されたロードデフォルト値の評価によって発生した制約性違反は、テーブルに挿入されたローごとにカウントされます。

load-specification のもう 1 つの重要部分は、**FILLER** オプションです。このオプションを指定すると、ソース入力ファイル内の指定したフィールドをスキップします。たとえば、ローの末尾の文字や入力ファイルのフィールド全体を、テーブルに追加する必要がない場合があります。**FILLER** では、*column-spec* 定義と同様に、ASCII 固定長バイト、セパレータで区切った可変長文字列、PREFIX バイトを使用するバイナリフィールドを指定できます。

標準

(先頭に戻る) (430 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - なし。

パーミッション

(先頭に戻る) (430 ページ)

LOAD TABLE 文の実行に必要なパーミッションは、データベースサーバの **-gl** コマンドラインオプションに応じて次のように異なります。

- **-gl ALL** – テーブルの所有者であるか、テーブルに対する ALTER または LOAD パーミッションが付与されているか、ALTER ANY TABLE、LOAD ANY TABLE、または ALTER ANY OBJECT システム権限が付与されている必要があります。
- **-gl DBA** – ALTER ANY TABLE、LOAD ANY TABLE、または ALTER ANY OBJECT システム権限が必要です。
- **-gl NONE** – **LOAD TABLE** 文の実行は許可されません。

-gl コマンドラインオプションの詳細については、ユーティリティガイド > 「start_iq データベースサーバ起動ユーティリティ」 > 「start_iq サーバオプション」を参照してください。

また、**LOAD TABLE** はテーブルに対する書き込みロックが必要です。

参照：

- INSERT 文 (412 ページ)
- LOAD_ZEROLENGTH_ASNULL オプション (704 ページ)
- NON_ANSI_NULL_VARCHAR オプション (732 ページ)

記憶領域サイズ

文字データの記憶領域サイズ、所定のカラム定義サイズと入力データサイズは次のとおりです。

表 8：文字データの記憶領域サイズ

データ型	カラム定義	入力データ	記憶領域
CHARACTER、CHAR	幅 (32K - 1) バイト	(32K - 1) バイト	(32K - 1) バイト
VARCHAR、CHARACTER VARYING	幅 (32K - 1) バイト	(32K - 1) バイト	(32K - 1) バイト

LOCK TABLE 文

他の同時トランザクションが指定の時間内にテーブルにアクセスしたり変更したりするのを防ぎます。

クイックリンク：

「パラメータ」 (452 ページ)

「例」 (453 ページ)

「使用法」 (453 ページ)

「標準」 (454 ページ)

「パーミッション」 (455 ページ)

構文

```
LOCK TABLE table-list [ WITH HOLD ]
    IN { SHARE | WRITE | EXCLUSIVE } MODE [ WAIT time ]

table-list
    [ owner. ] table-name [ , [ owner. ] table-name, ...]
```

パラメータ

(先頭に戻る) (452 ページ)

- table-name** – ビューではなくベーステーブルである必要があります。WRITE モードが有効なのは、IQ ベーステーブルの場合だけです。**LOCK TABLE** はテーブルリストにあるすべてのテーブルをロックするか、1 つもロックしないかのどちらかです。テーブルは、ローレベルバージョン管理 (RLV) に対応してはいけません。SQL Anywhere テーブルのロックを取得する場合、または **SHARE** か **EXCLUSIVE** のどちらかのロックを取得するときに指定できるのは、1 つのテーブルだけです。*table-name* の解析には、標準の SAP Sybase IQ オブジェクト修飾ルールが使用されます。
- WITH HOLD** – ロックは、接続の終わりまで保持されます。この句を指定しない場合は、現在のトランザクションがコミットまたはロールバックされた時点でロックは解放されます。**WRITE MODE** を持つ同じ文での **WITH HOLD** 句の使用はサポートされていなく、エラー SQLCODE=-131, ODBC 3 State="42000" が返されます。
- SHARE** – 他の同時トランザクションによるテーブルの変更はできませんが、読み込みアクセスは許可します。このモードでは、他のトランザクションが間

接的に、または **LOCK TABLE** を使用して明示的に修正中のローをロックしていない限り、テーブルのデータを修正できます。

- **WRITE** – 他のトランザクションによるテーブルのリストの変更はできません。接続の一番外側のトランザクションを無条件にコミットします。トランザクションが使用するスナップショットバージョンは、**LOCK TABLE IN WRITE MODE** 文ではなく、SAP Sybase IQ が処理する次のコマンドの実行により確立されます。

WRITE モードロックは、トランザクションがコミットされるかロールバックされると、または接続が切断されたときに解放されます。

- **EXCLUSIVE** – 他のトランザクションによるテーブルへのアクセスはできません。このモードでは、他のトランザクションはテーブルに対してクエリの発行、あらゆる修正、その他のあらゆる操作を実行できません。
- **WAIT time** – 全種類のロックの最長ブロック時間を指定します。この句はロックモードが **WRITE** の場合に必須です。time 引数が指定された場合、サーバは指定時間内に使用できる場合に限り、指定されたテーブルをロックします。この time 引数は、hh:nn:ss:sss の形式で指定できます。日付部分が指定されても、サーバはそれを無視し、その引数をタイムスタンプに変換します。time 引数が指定されない場合は、**WRITE** ロックが使用できるようになるか、または割り込みが発生するまで、サーバはいつまでも待ちます。

例

(先頭に戻る) (452 ページ)

- **例 1** – 次の文は、Customers と Employees の各テーブルが 5 分 3 秒以内に使用できるようになる場合、それぞれのテーブルの **WRITE** ロックを取得します。

```
LOCK TABLE Customers, Employees IN WRITE MODE WAIT
'00:05:03'
```

- **例 2** – Customers と Employees の各テーブルの **WRITE** ロックが使用できるようになるか、または割り込みが発生するまで、いつまでも待ちます。

```
LOCK TABLE Customers, Employees IN WRITE MODE WAIT
```

使用法

(先頭に戻る) (452 ページ)

コーディネータ上の IQ メインストアのテーブルに対して **LOCK TABLE** 文を実行しても、セカンダリサーバ上の接続からこれらのテーブルへのアクセスは影響を受けません。次に例を示します。

コーディネータ接続では、次のコマンドを発行します。

SQL 文

```
LOCK TABLE coord1 WITH HOLD IN EXCLUSIVE MODE
```

コーディネータ上で **sp_iqlocks** を実行すると、テーブル coord1 に排他 (E) ロックが存在することを確認できます。

セカンダリサーバ上の接続に対して **sp_iqlocks** を実行した結果には、テーブル coord1 に対する排他ロックが示されません。この接続上のユーザは、コーディネータ上のテーブル coord1 に対する更新を表示できます。

コーディネータ上の他の接続からは、coord1 に対する排他ロックを表示できません。コーディネータ上の別の接続からテーブル coord1 を選択しようとする、と、「User DBA has the row in coord1 locked」が返されます。

ビューでの **LOCK TABLE** はサポートされていません。ビューをロックしようすると、コマンドで指定されたモードにかかわらず、共有のスキーマロックが取得されます。共有のスキーマロックでは、他のトランザクションによるテーブルスキーマの変更はできません。

Transact-SQL (T-SQL) ストアドプロシージャ言語は、**LOCK TABLE** をサポートしません。たとえば、次の文は、「Syntax error near LOCK」を返します。

```
CREATE PROCEDURE tproc ()
AS
BEGIN
COMMIT;
LOCK TABLE t1 IN SHARE MODE
INSERT INTO t1 VALUES (30)
END
```

Watcom-SQL ストアドプロシージャ言語は、**LOCK TABLE** をサポートします。デフォルトでは、コマンドのデリミタはセミコロン (;) です。次に例を示します。

```
CREATE PROCEDURE tproc ()
AS
BEGIN
COMMIT;
LOCK TABLE t1 IN SHARE MODE
INSERT INTO t1 VALUES (30)
END
```

標準

(先頭に戻る) (452 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Adaptive Server でサポートされています。WITH HOLD 句は、Adaptive Server ではサポートされていません。Adaptive Server には WAIT 句がありますが、これは SQL Anywhere ではサポートされていません。

パーミッション

(先頭に戻る) (452 ページ)

SHARE モードでテーブルをロックするには、SELECT 権限が必要です。

EXCLUSIVE モードでテーブルをロックするには、そのテーブルの所有者であるか以下のシステム権限のいずれかが必要です。

- ALTER ANY OBJECT
- INSERT ANY TABLE
- UPDATE ANY TABLE
- DELETE ANY TABLE
- ALTER ANY TABLE
- LOAD ANY TABLE
- TRUNCATE ANY TABLE

参照：

- SELECT 文 (523 ページ)

LOOP 文

文リストの実行を繰り返します。

クイックリンク：

「例」 (455 ページ)

「使用法」 (456 ページ)

「標準」 (456 ページ)

「パーミッション」 (456 ページ)

構文

```
[ statement-label: ]  
... [ WHILE  
           search-condition ] LOOP  
... statement-list  
... END  
           LOOP [ statement-label ]
```

例

(先頭に戻る) (455 ページ)

SQL 文

- **例 1** – プロシージャ中の **WHILE** のループ

```
...
SET i = 1 ;
WHILE i <= 10 LOOP
    INSERT INTO Counters( number ) VALUES ( i ) ;
    SET i = i + 1 ;
END LOOP ;
...
```

- **例 2** – プロシージャ中の指定されたラベルのループ

```
SET i = 1;
lbl:
LOOP
    INSERT
    INTO Counters( number )
    VALUES ( i ) ;
    IF i >= 10 THEN
        LEAVE lbl ;
    END IF ;
    SET i = i + 1 ;
END LOOP lbl
```

使用法

(先頭に戻る) (455 ページ)

WHILE 文と **LOOP** 文は制御文です。これを使用して、*search-condition* が TRUE と評価するかぎり、SQL 文のリストを繰り返し実行できます。**LEAVE** 文を使用して、**END LOOP** に続く最初の文から実行を再開できます。

終了の *statement-label* を指定する場合は、開始の *statement-label* と一致させる必要があります。

標準

(先頭に戻る) (455 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - Adaptive Server ではサポートされていません。**WHILE** 文は、Transact-SQL ストアドプロシージャにループ機能を提供します。

パーミッション

(先頭に戻る) (455 ページ)

なし

参照:

- FOR 文 (361 ページ)

- LEAVE 文 (428 ページ)
- WHILE 文 [T-SQL] (583 ページ)

MESSAGE 文

メッセージを表示します。これには、任意の式を指定できます。句は、メッセージが表示される場所を指定します。

クイックリンク：

「パラメータ」 (457 ページ)

「例」 (459 ページ)

「使用法」 (459 ページ)

「標準」 (460 ページ)

「パーミッション」 (460 ページ)

構文

```
MESSAGE expression, ...
  [ TYPE { INFO | ACTION | WARNING | STATUS } ]
  [ TO { CONSOLE
        | CLIENT [ FOR { CONNECTION conn_id [ IMMEDIATE ] | ALL } ]
        | [ EVENT | SYSTEM ] LOG }
  [ DEBUG ONLY ] ]
```

パラメータ

(先頭に戻る) (457 ページ)

- **FOR** – このメッセージに関する通知を受け取る接続を指定します。
 - **CONNECTION *conn_id*** – メッセージの受信者の接続 ID。
 - **IMMEDIATE** – SQL 文がいつ実行されるかに関係なく、接続はメッセージを数秒以内に受信します。

通常、宛先の接続がデータベースサーバ要求を実行していない場合でも、IMMEDIATE 句を使用して送信されたメッセージは 5 秒未満で配信されます。クライアント接続が 1 秒間に複数の要求を実行した場合、クライアント接続が非常に大きな BLOB データを受信した場合、またはクライアントのメッセージコールバックの実行に 1 秒以上かかった場合、メッセージの配信が遅れる可能性があります。また、複数の IMMEDIATE メッセージを 1 つの接続に 2 秒ごとに送信すると、メッセージの配信が遅れたり、エラーメッセージが生成されたりすることがあります。クライアント接続が切断

されると、MESSAGE...IMMEDIATE 文が正常に配信されないことがあります。

- **ALL** – 開いているすべての接続がメッセージを受信します。

FOR 句を使用すると、サーバで検出されたイベントを別のアプリケーションに通知できるため、アプリケーションの側でイベントを明示的にチェックする必要がなくなります。FOR 句を使用した場合、受信者は次回 SQL 文を実行したときにメッセージを受け取ります。受信者が現在 SQL 文を実行中であるときは、文の完了時にメッセージを受信します。実行中の文がストアードプロシージャの呼び出しである場合、呼び出しの完了前にメッセージを受信します。

アプリケーションで、メッセージの送信後短時間のうちに通知を必要とし、接続で SQL 文が実行されていない場合は、2 つ目の接続を使用できます。この接続は 1 つ以上の **WAITFOR DELAY** 文を実行できます。これらの文は、サーバやネットワークのリソースを膨大に消費することはありませんが (ポーリング方式で実行されるため)、アプリケーションはメッセージの送信後ただちに通知を受け取ることができます。

- **TYPE** – 有効なのは、メッセージがクライアントに送信される場合だけです。クライアントアプリケーションで、メッセージの処理方法を指定する必要があります。Interactive SQL は、次のロケーションにメッセージを表示します。
 - **INFO** – (デフォルト) メッセージウィンドウ。
 - **ACTION** – [OK] ボタンのあるメッセージボックス。
 - **WARNING** – [OK] ボタンのあるメッセージボックス。
 - **STATUS** – メッセージウィンドウ枠。
- **TO** – メッセージの送信先を指定します。
 - **CONSOLE** – (デフォルト) メッセージをデータベースサーバウィンドウに送信します。
 - **CLIENT** – メッセージをクライアントアプリケーションに送信します。アプリケーションではメッセージの処理方法を定める必要があります。また、この決定のベースとなる情報として **TYPE** 句を使用できます。
 - **LOG** – メッセージを -o オプションで指定されたサーバログファイルに送信します。
- **DEBUG ONLY** – ストアドプロシージャに追加されたデバッグメッセージの有効または無効を、**DEBUG_MESSAGES** オプションの設定を変えることによって制御します。DEBUG ONLY が指定されている場合は、**DEBUG_MESSAGES** オプションが ON に設定されているときにだけ、**MESSAGE** 文が実行されます。

注意： DEBUG ONLY メッセージは、**DEBUG_MESSAGES** オプションが OFF になっていれば低コストなため、通常、実稼働システムのストアードプロシージャ

内に残されます。ただし、頻繁に実行される場所ではパフォーマンスがやや低下する可能性があるため、慎重に使用してください。

例

(先頭に戻る) (457 ページ)

- **例 1** – データベースサーバのメッセージウィンドウに、文字列 The current date and time, と、現在の日付と時刻を表示します。

```
CREATE PROCEDURE message_test ()
BEGIN
MESSAGE 'The current date and time: ', Now();
END;
CALL message_test();
```

- **例 2** – ODBC でコールバックを登録するには、最初にメッセージハンドラを宣言します。

```
void SQL_CALLBACK my_msgproc(
    void * _sqlca,
    unsigned char    msg_type,
    long             code,
    unsigned short   len,
    char*            msg )
{ ... }
```

SQLSetConnectAttr 関数を呼び出して、宣言されたメッセージハンドラをインストールします。

```
rc = SQLSetConnectAttr(
    dbc,
    ASA_REGISTER_MESSAGE_CALLBACK,
    (SQLPOINTER) &my_msgproc, SQL_IS_POINTER );
```

使用法

(先頭に戻る) (457 ページ)

MESSAGE ... TO CLIENT 文を発行するプロシージャは、接続に関連付けられている必要があります。

たとえば、次の例ではイベントが接続の外部で発生するため、メッセージボックスは表示されません。

```
CREATE EVENT CheckIdleTime TYPE ServerIdle
WHERE event_condition( 'IdleTime' ) > 100
HANDLER
BEGIN
MESSAGE 'Idle engine' type warning to client;
END;
```

一方、次の例では、メッセージがサーバコンソールに書き込まれます。

SQL 文

```
CREATE EVENT CheckIdleTime TYPE ServerIdle
WHERE event_condition( 'IdleTime' ) > 100
HANDLER
BEGIN
  MESSAGE 'Idle engine' type warning to console;
END;
```

有効な式には、引用符で囲まれた文字列などの定数、または変数、関数が含まれます。ただし、本来ならクエリも式の 1 つですが、**MESSAGE** 文の出力にクエリを含めることはできません。

ESQL クライアントと ODBC クライアントは、メッセージコールバック関数をおしてメッセージを受信します。どちらの場合も、関数の登録が必要です。ESQL メッセージハンドラを登録するには、**db_register_callback** 関数を使用します。

ODBC クライアントの場合は、**SQLSetConnectAttr** 関数を使用してコールバック関数を登録してください。

標準

(先頭に戻る) (457 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Adaptive Server ではサポートされていません。Transact-SQL の **PRINT** 文は、同様な機能を提供し、SQL Anywhere で使用できません。

パーミッション

(先頭に戻る) (457 ページ)

データベースに接続しておく必要があります。

- **FOR** 句 - 次のいずれかが必要です。
 - **SERVER OPERATOR** システム権限。
 - **DROP CONNECTION** システム権限。
- **TO EVENT LOG** 句または **TO SYSTEM LOG** 句 - **SERVER OPERATOR** システム権限が必要です。

参照：

- **CREATE PROCEDURE** 文 (210 ページ)
- **WAITFOR** 文 (579 ページ)
- **DEBUG_MESSAGES** オプション (643 ページ)

OPEN 文 [ESQL] [SP]

事前に宣言したカーソルをオープンし、データベースからの情報にアクセスします。

クイックリンク：

「パラメータ」 (461 ページ)

「例」 (461 ページ)

「使用法」 (462 ページ)

「標準」 (462 ページ)

「パーミッション」 (463 ページ)

構文

```
OPEN cursor-name
... [ USING [ DESCRIPTOR { sqllda-name | host-variable [, ...] } ] ]
... [ WITH
      HOLD ]
```

パラメータ

(先頭に戻る) (461 ページ)

- **cursor-name** – 識別子またはホスト変数

カーソル名が識別子または文字列によって指定される場合、C プログラムでは **OPEN** の前に対応する **DECLARE CURSOR** 文を置く必要があります。ホスト変数によってカーソル名を指定する場合、**DECLARE CURSOR** 文を **OPEN** 文の前に実行する必要があります。

- **sqllda-name** – 識別子

- **WITH HOLD** – 次に実行されるトランザクションのためにカーソルを開いたままにします。カーソルは現在の接続が終了するまで、または明示的な **CLOSE** 文が実行されるまでオープンしたままです。接続が終了すると、カーソルは自動的に閉じます。

- **USING** – カーソルが宣言されている **SELECT** 文のプレースホルダバインド変数にバインドされるホスト変数を指定します。

例

(先頭に戻る) (461 ページ)

- **例 1** – Embedded SQL での **OPEN** の使用法:

```
EXEC SQL OPEN employee_cursor;
```

および

```
EXEC SQL PREPARE emp_stat FROM 'SELECT EmployeeID, Surname FROM
Employees WHERE name like ?'; EXEC SQL DECLARE employee_cursor
CURSOR FOR emp_stat; EXEC SQL OPEN employee_cursor USING :pattern;
```

- **例 2** – プロシージャからの例

```
BEGIN
DECLARE cur_employee CURSOR FOR
  SELECT Surname
  FROM Employees ;
DECLARE name CHAR(40) ;
OPEN cur_employee;
LOOP
FETCH NEXT cur_employee into name ;
  ...
END LOOP
CLOSE cur_employee;
END
```

使用法

(先頭に戻る) (461 ページ)

デフォルトで、すべてのカーソルは現在のトランザクションの最後 (**COMMIT** または **ROLLBACK**) で自動的に閉じます。

カーソルは、最初のローの前に置かれます。

FOR READ ONLY 句を使用して宣言されたカーソルからは、最初に **FETCH** 文を実行したときのテーブルのバージョンではなく、カーソルを開いたときにカーソルの宣言に使用されるテーブルのバージョンが表示されます。

USING DESCRIPTOR *sqlda-name*、**host-variable**、**BLOCK n** の各句を使用できるのは、Embedded SQL だけです。

OPEN 文が正常に実行された後で、**SQLCA (SQLIOESTIMATE)** の *sqlerrd[3]* フィールドに、クエリのすべてのローをフェッチするのに必要な入出力操作の数の推定値が格納されます。同様に、**SQLCA (SQLCOUNT)** の *sqlerrd[2]* フィールドに、カーソル内にある実際のロー数 (0 以上の値)、またはその推定値 (絶対値が推定値である負の数) が入ります。データベースサーバが、ローをカウントしなくてもこの値を計算できる場合、*sqlerrd[2]* フィールドは実際のローの数になります。

標準

(先頭に戻る) (461 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - 単純な **OPEN cursor-name** 構文は Adaptive Server でサポートされています。それ以外の句はいずれも Adaptive Server ストアドプロシージャでサポートされていません。Open Client/Open Server は、**USING** 記述子またはホスト名変数構文をサポートします。

パーミッション

(先頭に戻る) (461 ページ)

- **SELECT** 文のすべてのテーブルに対する **SELECT** パーミッション、または **CALL** 文のプロシージャに対する **EXECUTE** パーミッションが必要です。
- カーソルが **CALL** 文にあるとき、**OPEN** は最初の結果セット (**INTO** 句がない **SELECT** 文)が見つかるまでプロシージャを実行します。プロシージャが完了しても結果セットが見つからない場合は、**SQLSTATE_PROCEDURE_COMPLETE** 警告が設定されます。

参照：

- **CLOSE** 文 [ESQL] [SP] (122 ページ)
- **DECLARE CURSOR** 文 [ESQL] [SP] (299 ページ)
- **FETCH** 文 [ESQL] [SP] (357 ページ)
- **PREPARE** 文 [ESQL] (469 ページ)
- **RESUME** 文 (495 ページ)

OUTPUT 文 [Interactive SQL]

現在のクエリで取得した情報をファイルに書き込みます。

クイックリンク：

「パラメータ」 (464 ページ)

「例」 (466 ページ)

「使用法」 (467 ページ)

「標準」 (467 ページ)

「パーミッション」 (467 ページ)

構文

```
OUTPUT TO filename
  [ APPEND ] [ VERBOSE ]
  [ FORMAT output-format ]
```

```
[ ESCAPE CHARACTER character ]
[ DELIMITED BY string ]
[ QUOTE string [ ALL ] ]
[ COLUMN WIDTHS ( integer, ... ) ]
[ HEXADECIMAL { ON | OFF | ASIS } ]
[ ENCODING encoding ]
[ WITH COLUMN NAMES ]
```

output-format

TEXT | FIXED | HTML | SQL | XML

パラメータ

(先頭に戻る) (463 ページ)

- **FORMAT** – 出力フォーマット。FORMAT 句を指定しないと、Interactive SQL **OUTPUT_FORMAT** データベースオプションの設定が使用されます。
- **TEXT** – TEXT フォーマットでファイルに出力し、1 行につき 1 ローを書き込みます。すべての値がカンマで区切られ、文字列はアポストロフィ (一重引用符) で囲まれます。デリミタと引用符文字列は、DELIMITED BY 句と QUOTE 句を使って変更できます。QUOTE 句で ALL 句を指定すると、文字列だけでなく、すべての値が引用符で囲まれます。TEXT はデフォルトの出力フォーマットです。

この他に 3 つの特別なシーケンスが認識されます。2 文字の文字列 \backslashn は改行文字を表し、 $\backslash\text{¥}$ は単一の円記号 (¥) を表します。 $\backslashx\text{DD}$ のようなシーケンスは 16 進コード DD の文字を表します。

文字列の値を返す Java メソッドをエクスポートする場合は、HEXADECIMAL OFF 句を使用する必要があります。

- **FIXED** – それぞれのカラムが固定幅を持つ固定フォーマットで出力します。各カラムの幅は COLUMN WIDTHS 句を使用して指定できます。このフォーマットでは、カラム見出しは出力されません。

COLUMN WIDTHS 句を省略すると、各カラムの幅はデータ型から計算され、そのデータ型のどのような値でも十分に保持できる大きさになります。ただし、LONG VARCHAR と LONG BINARY 型のデータは例外で、デフォルトで 32 KB になります。

- **HTML** – Hyper Text Markup Language フォーマットで出力します。
- **SQL** – テーブル内の情報を再作成するのに必要な Interactive SQL **INPUT** 文を出力します。

注意： SAP Sybase IQ は、**INPUT** 文はサポートしていません。この出力を使ってデータをロードバックするためには、**INPUT** 文を有効な **LOAD TABLE** (または **INSERT**) 文に編集する必要があります。

- **XML** – UTF-8 でエンコードされ、DTD が埋め込まれた XML ファイルに出力します。バイナリ値は、2 桁の 16 進数文字列として表されるバイナリデータとして CDATA ブロック内にエンコードされます。**LOAD TABLE** 文のファイルフォーマットに XML を指定することはできません。
- **APPEND** – クエリの結果を、既存の出力ファイルに記述されている内容に上書きするのではなく、ファイルの末尾に追加します。**APPEND** 句を使用しないと、**OUTPUT** 文はデフォルトで出力ファイルの内容を上書きします。

APPEND 句は、出力フォーマットが **TEXT**、**FIXED**、または **SQL** の場合に有効です。

- **VERBOSE** – クエリに関するエラーメッセージ、データの選択に使用された SQL 文、およびデータ自体が出力ファイルに書き込まれます。**VERBOSE** 句を省略すると (つまり、デフォルト)、データだけがファイルに書き込まれます。**VERBOSE** 句は、出力フォーマットが **TEXT**、**FIXED**、または **SQL** の場合に有効です。
- **ESCAPE CHARACTER** – 16 進のコードおよび記号として格納されている文字に使用するデフォルトのエスケープ文字は、円記号 (¥) です。たとえば、¥x0A は改行文字です。

このデフォルトのエスケープ文字は、**ESCAPE CHARACTER** 句を使用して変更できます。たとえば、感嘆符 (!) をエスケープ文字として使用するには、次のように入力します。

```
... ESCAPE CHARACTER '!'
```

- **DELIMITED BY** – 出力フォーマットが **TEXT** の場合にのみ使用できます。コラム間にデリミタ文字列 (デフォルトはカンマ) が挿入されます。
- **QUOTE** – 出力フォーマットが **TEXT** の場合にのみ使用できます。文字列値を引用符で囲みます。デフォルトは一重引用符です。**QUOTE** 句で **ALL** を指定すると、文字列だけでなく、すべての値が引用符文字列で囲まれます。
- **COLUMN WIDTHS** – **FIXED** フォーマットの出力のコラム幅を指定します。
- **HEXADECIMAL** – バイナリデータのアンロード方法を指定します。**TEXT** フォーマットにのみ適用されます。この句を **ON** に設定すると、バイナリデータは 0xabcd のようなフォーマットでアンロードされます。この句を **OFF** に設定すると、バイナリデータはアンロード時にエスケープされます (¥xab¥xcd)。この句を **ASIS** に設定すると、値はたとえ制御文字を含んでいても、そのまま

(すなわち、エスケープなしで)書き込まれます。ASIS は、タブや復帰改行などのフォーマット記号を含むテキストに適しています。

- **ENCODING** – ファイルの書き込み時に使用するエンコードを指定します。ENCODING 句は、TEXT フォーマットでのみ使用できます。文字列または識別子を指定できます。

ENCODING 句を指定しないと、Interactive SQL は、ファイルの書き込みに使用するコードページを次の設定に基づいて決定します。この場合、最初の項目のコードページ値が後の項目のコードページ値よりも優先されます。

- **DEFAULT_ISQL_ENCODING** オプションで指定されたコードページ (このオプションが設定されている場合)
- Interactive SQL を実行しているコンピュータのデフォルトのコードページ

例

(先頭に戻る) (463 ページ)

- **例 1** – Employees テーブルの内容をテキストファイルに出力します。

```
SELECT * FROM Employees;
OUTPUT TO employees.txt FORMAT TEXT
```

- **例 2** – Employees テーブルの内容を既存のファイルの終わりに追加します。また、クエリに関するメッセージも同じファイルに含めます。

```
SELECT * FROM Employees;
OUTPUT TO employees.txt APPEND VERBOSE
```

- **例 3** – 改行文字が埋め込まれた値をエクスポートします。改行文字のコード値は 10 です。これは、SQL 文では文字列 '¥x0a' として表現することができます。

HEXADECIMAL ON を指定して、この文を実行します。

```
SELECT 'line1¥x0aline2'; OUTPUT TO file.txt HEXADECIMAL ON
```

次のテキストを含む 1 行のファイルが取得されます。

```
line10x0aline2
```

HEXADECIMAL OFF を指定して、同じ文を実行します。

```
line1¥x0aline2
```

HEXADECIMAL を ASIS に設定すると、次の 2 行から成るファイルが出力されます。

```
'line1
line2'
```

ASIS を使用すると 2 行が生成されるのは、埋め込まれた改行文字が、2 桁の 16 進表現に変換されることも、プレフィクスを付加されることもなく、エクスポートされるためです。

使用法

(先頭に戻る) (463 ページ)

現在のクエリは、[結果] ウィンドウ枠の [結果] タブに表示される情報を生成した **SELECT** 文または **LOAD TABLE** 文です。現在のクエリがない場合、**OUTPUT** 文はエラーをレポートします。

注意： **OUTPUT** 文は、特にクエリやレポートの結果を別のアプリケーションで利用するときには便利ですが、バルクオペレーションには向いていません。大量のデータを移動する場合は、**SELECT** 文で ASCII および BINARY のデータ抽出機能を使用してください。これらの抽出機能を使用すると、格段に高いパフォーマンスで大量のデータを移動し、出力ファイルを作成してロードに使用できます。

関連する動作

- Interactive SQL では、現在のクエリの結果だけが [結果] タブに表示されます。前回のクエリの結果は、現在のクエリの結果にすべて置き換えられます。

標準

(先頭に戻る) (463 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - なし。

パーミッション

(先頭に戻る) (463 ページ)

なし

参照：

- SELECT 文 (523 ページ)
- DEFAULT_ISQL_ENCODING オプション [Interactive SQL] (648 ページ)

PARAMETERS 文 [Interactive SQL]

Interactive SQL (**dbisql**) コマンドファイルにパラメータを指定します。

クイックリンク：

SQL 文

「例」 (468 ページ)

「使用法」 (468 ページ)

「標準」 (468 ページ)

「パーミッション」 (468 ページ)

構文

```
PARAMETERS parameter1, parameter2, ...
```

例

(先頭に戻る) (467 ページ)

- **例 1** - 次の **dbisql** コマンドは 2 つのパラメータを取ります。

```
PARAMETERS department_id, file ;  
SELECT Surname  
FROM Employees  
WHERE DepartmentID = {department_id}  
>#{file}.dat;
```

使用法

(先頭に戻る) (467 ページ)

PARAMETERS は、指定したコマンドファイルに対するパラメータ数を指定します。また、これらのパラメータに名前を与え、そのコマンドファイル内で後から参照できるようにします。

パラメータは、指定したパラメータを置き換えるコマンドファイルに、そのパラメータを次のように入れることで参照されます。

```
{parameter1}
```

大カッコとパラメータ名の間には、スペースを入れないでください。

コマンドファイルを呼び出すときに、すべての必要なパラメータを指定しないと、**dbisql** は不足しているパラメータの値を要求するメッセージを表示します。

標準

(先頭に戻る) (467 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - なし。

パーミッション

(先頭に戻る) (467 ページ)

なし

参照：

- READ 文 [Interactive SQL] (478 ページ)

PREPARE 文 [ESQL]

後から実行する文、またはカーソルに使用する文を準備します。

クイックリンク：

「パラメータ」 (469 ページ)

「例」 (470 ページ)

「使用法」 (470 ページ)

「標準」 (471 ページ)

「パーミッション」 (471 ページ)

構文

```
PREPARE statement-name
FROM statement [ FOR { READ ONLY | UPDATE [ OF column-name-list ] } ]
... [ DESCRIBE describe-type INTO [ [ SQL ] DESCRIPTOR ] descriptor ]
... [ WITH EXECUTE ]
```

describe-type

```
{ ALL
| BIND VARIABLES
| INPUT
| OUTPUT
| SELECT LIST } ... { LONG NAMES [ [ OWNER.]TABLE.]COLUMN ]
| WITH VARIABLE RESULT }
```

パラメータ

(先頭に戻る) (469 ページ)

- **statement-name** – 文の実行時に参照されます。また、文が **SELECT** 文の場合、カーソルを開きます。*statement-name* は、自動的に格納される `sqlca.h` ヘッダファイルの中で定義される `a_sql_statement_number` 型のホスト変数である場合もアリアンズ。識別子を *statement-name* で使用する場合、この *statement-name* を使用して各モジュールに準備できるのは、それぞれ 1 文だけです。
- **FOR UPDATE | FOR READ ONLY** – カーソルで文が使用される場合のカーソル更新可能性を定義します。FOR READ ONLY カーソルは **UPDATE** (位置付け) 操作や **DELETE** (位置付け) 操作には使用できません。FOR READ ONLY がデフォ

ルトです。FOR UPDATE を指定したカーソル要求への応答の場合、SAP Sybase IQ は value-sensitive カーソルまたは sensitive カーソルを提供します。insensitive カーソルと asensitive カーソルは更新できません。

- **DESCRIBE INTO DESCRIPTOR** – 準備文は指定した記述子に記述されます。記述タイプとして、**DESCRIBE** 文で許容されるいずれかのものを使用できません。

DESCRIBE INTO DESCRIPTOR 句を使用すると、クライアントとサーバとの間で必要な通信が少なくなるので、パフォーマンスが向上する場合があります。

- **WITH EXECUTE – CALL** 文または **SELECT** 文ではなく、ホスト変数が含まれていない場合にのみ、文が実行されます。正常に実行された後、文はその場で削除されます。文の **PREPARE** と **DESCRIBE** (存在する場合) が正常に終了しても、文が実行できない場合、警告 SQLCODE 111、SQLSTATE 01W08 が設定され、文は削除されません。

WITH EXECUTE 句を使用すると、クライアントとサーバとの間で必要な通信が少なくなるので、パフォーマンスが向上する場合があります。

- **WITH VARIABLE RESULT** – 複数の結果セットと異なる数または型のカラムを持つプロシージャを記述します。WITH VARIABLE RESULT 句を使用する場合、記述後にデータベースサーバによって SQLCOUNT 値に次のいずれかの値が設定されます。
 - **0** – 結果セットは変更される場合があります。各 **OPEN** 文の後でプロシージャコールを再度記述してください。
 - **1** – 結果セットは固定です。再度記述する必要はありません。

例

(先頭に戻る) (469 ページ)

- **例 1** – 簡単なクエリを準備します。

```
EXEC SQL PREPARE employee_statement FROM
'SELECT Surname FROM Employees';
```

使用法

(先頭に戻る) (469 ページ)

PREPARE 文は、*statement* から SQL 文を作成し、作成した文を *statement-name* と関連付けます。

ホスト変数を *statement-name* に使用する場合は、short int 型にする必要があります。sqlca.h 内には、a_sql_statement_number というこの型の型定義があります。この型は SQL プリプロセッサで認識されるので、**DECLARE** セクション

の中で使用できます。**PREPARE** 文の実行中に、データベースによってホスト変数が入力されるため、プログラマが初期化する必要はありません。

次は、PREPARED の対象になる文のリストです。

- **ALTER**
- **CALL**
- **COMMENT ON**
- **CREATE**
- **DELETE**
- **DROP**
- **GRANT**
- **INSERT**
- **REVOKE**
- **SELECT**
- **SET OPTION**

互換性保持のため、**COMMIT** 文、**PREPARE TO COMMIT** 文、**ROLLBACK** 文の準備は引き続きサポートされます。ただし、特定のアプリケーション環境で必要とされるため、トランザクション管理操作はすべて、静的 Embedded SQL を使用して実行します。また、他の Embedded SQL システムは、動的トランザクション管理操作をサポートしません。

注意： 使用後の文は、必ず削除してください。削除しないと、文が使用したメモリを再使用できません。

関連する動作：

- 以前に同じ名前で作成した文が失われます。

標準

(先頭に戻る) (469 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - Open Client/Open Server でサポートされています。

パーミッション

(先頭に戻る) (469 ページ)

なし

参照：

- DECLARE CURSOR 文 [ESQL] [SP] (299 ページ)
- DESCRIBE 文 [ESQL] (314 ページ)

SQL 文

- DROP 文 (319 ページ)
- EXECUTE 文 [ESQL] (348 ページ)
- OPEN 文 [ESQL] [SP] (461 ページ)

PRINT 文 [T-SQL]

データベースサーバのメッセージウィンドウにメッセージを表示します。

クイックリンク：

「例」 (472 ページ)

「使用法」 (473 ページ)

「標準」 (473 ページ)

「パーミッション」 (473 ページ)

構文

```
PRINT format-string [, arg-list]
```

例

(先頭に戻る) (472 ページ)

- **例 1** – サーバのメッセージウィンドウにメッセージを表示します。

```
CREATE PROCEDURE print_test
AS
PRINT 'Procedure called successfully'
```

この文は、クライアントに "Procedure called successfully" という文字列を返します。

```
EXECUTE print_test
```

- **例 2** – PRINT 文でプレースホルダを使用します。これらの文をプロシージャ内で実行します。

```
DECLARE @var1 INT, @var2 INT
SELECT @var1 = 3, @var2 = 5
PRINT 'Variable 1 = %1!, Variable 2 = %2!', @var1, @var2
```

- **例 3** – RAISERROR を使用して接続を禁止します。

```
CREATE procedure DBA.login_check()
begin
  // Allow a maximum of 3 concurrent connections
  IF( db_property('ConnCount') > 3 ) then
    raiserror 28000
    'User %1! is not allowed to connect -- there are
```

```
        already %2! users logged on',
        current user,
        cast(db_property('ConnCount') as int)-1;
    ELSE
    call sp_login_environment;
    end if;
end
go
grant execute on DBA.login_check to PUBLIC
go
set option PUBLIC.Login_procedure='DBA.login_check'
go
```

これ以外の接続禁止方法では、**LOGIN_PROCEDURE** オプションまたは **sp_iqmodifylogin** システムストアードプロシージャを使用します。

使用法

(先頭に戻る) (472 ページ)

PRINT 文は、Open Client アプリケーションまたは JDBC アプリケーションから接続している場合、クライアントウィンドウにメッセージを返します。Embedded SQL または ODBC アプリケーションから接続している場合、このメッセージはデータベースサーバのウィンドウに表示されます。

フォーマット文字列は、オプション引数リスト (arg-list) にある引数のプレースホルダを含むことができます。プレースホルダの形式は *%nn!* で、*nn* は 1 ~ 20 の整数です。

標準

(先頭に戻る) (472 ページ)

- SQL – ISO/ANSI SQL 文法の Transact-SQL 拡張。
- SAP Sybase Database 製品 - Adaptive Server でサポートされています。

パーミッション

(先頭に戻る) (472 ページ)

データベースに接続しておく必要があります。

参照：

- MESSAGE 文 (457 ページ)
- LOGIN_PROCEDURE オプション (708 ページ)

PUT 文 [ESQL]

指定されたカーソルにローを挿入します。

クイックリンク：

「パラメータ」 (474 ページ)

「例」 (474 ページ)

「使用法」 (475 ページ)

「標準」 (475 ページ)

「許可」 (475 ページ)

構文

```
PUT cursor-name [ USING DESCRIPTOR sqlda-name  
| FROM hostvar-list ] [ INTO { DESCRIPTOR into-sqlda-name  
| into-hostvar-list } ] [ ARRAY :nnn ]
```

パラメータ

(先頭に戻る) (474 ページ)

- **cursor-name** – 識別子またはホスト変数
- **sqlda-name** – 識別子
- **hostvar-list** – インジケータ変数を含む場合がある
- **ARRAY** – 使用すると、ワイド挿入が実行できます。このワイド挿入では複数のローが一度に挿入され、パフォーマンスが向上します。値 *nnn* は、挿入するローの数です。SQLDA には $nnn * (\text{ローあたりのカラム数})$ 変数を指定する必要があります。最初のローは SQLDA の変数 0 から (ローあたりのカラム数) - 1 に入り、以後のローも同様です。

注意： スクロール (値依存) カーソルでは、新しいローが WHERE 句と一致し、キーセットカーソルがデータ移植が完了していない場合に、挿入されたローが表示されます。動的なカーソルでは、挿入されたローが WHERE 句と一致すると表示されます。Insensitive カーソルは更新できません。

例

(先頭に戻る) (474 ページ)

- **例 1** – Embedded SQL での **PUT** の使用法:

```
EXEC SQL PUT cur_employee FROM :EmployeeID, :Surname;
```

使用法

(先頭に戻る) (474 ページ)

指定されたカーソルにローを挿入します。カラムの値は、最初の **SQLDA** から取得されます。または、**INSERT** 文に指定したカラム (**INSERT** カーソル用) または **select** リストのカラム (**SELECT** カーソル用) と 1 対 1 で対応するホスト変数リストからカラムの値を取得します。

PUT 文は、**FROM** 句の単一テーブルを参照する、または単一のベーステーブルで構成される更新可能なビューを参照する **INSERT** 文または **SELECT** 文のカーソルのみで使用することができます。

SQLDA 内の **sqldata** ポインタが **NULL** ポインタである場合、そのカラムには値が指定されません。カラムに関連付けられている **DEFAULT VALUE** が存在する場合はそれが使用され、その他の場合は **NULL** 値が使用されます。

2 番目の **SQLDA** またはホスト変数のリストには、**PUT** 文の結果が格納されていません。

データベースへの **LONG VARCHAR** 値または **LONG BINARY** 値の挿入については、「**SET** 文 [ESQL]」を参照してください。

関連する動作：

- ローを **value-sensitive** (キーセット駆動型) カーソルに挿入すると、挿入されたローは結果セットの最後に表示されます。ローがクエリの **WHERE** 句に一致しない場合も、**ORDER BY** 句の通常どおりにローが結果セットの別の場所に配置される場合も同様です。

標準

(先頭に戻る) (474 ページ)

- **SQL - ISO/ANSI SQL** 準拠。
- **SAP Sybase Database 製品 - Open Client/Open Server** でサポートされています。

パーミッション

(先頭に戻る) (474 ページ)

INSERT 権限が必要です。

参照：

- DELETE (位置付け) 文 [ESQL] [SP] (312 ページ)
- INSERT 文 (412 ページ)
- SET 文 [ESQL] (534 ページ)
- UPDATE 文 (566 ページ)
- UPDATE (位置付け) 文 [ESQL] [SP] (571 ページ)

RAISERROR 文 [T-SQL]

ユーザ定義エラーを通知し、クライアントにメッセージを送信することができます。

クイックリンク：

「パラメータ」 (476 ページ)

「例」 (477 ページ)

「使用法」 (477 ページ)

「標準」 (477 ページ)

「許可」 (477 ページ)

構文

```
RAISERROR error-number [ format-string ] [, arg-list ]
```

パラメータ

(先頭に戻る) (476 ページ)

- **error-number** – は 17000 よりも大きい 5 桁の整数です。このエラー番号は、グローバル変数 @@error に格納されます。
- **format-string** – 指定しないか、空にすると、エラー番号を使用してシステムテーブルのエラーメッセージが確認されます。Adaptive Server は、17000 ~ 19999 のメッセージを SYSMESSAGES テーブルから取得します。SAP Sybase IQ では、このテーブルは空のビューであるため、この範囲のエラーはフォーマット文字列を提供します。エラー番号が 20000 以上のメッセージは、SYS.SYSUSERMESSAGES テーブルから取得されます。

format-string の最大長は 255 バイトです。これは、Adaptive Server の場合と同じです。

フォーマット文字列には、オプション引数リストにある引数のプレースホルダを含めることができます。このプレースホルダの形式は %nn! で、nn は 1 ～ 20 の整数です。

例

(先頭に戻る) (476 ページ)

- **例 1** – エラー 99999 を発行します。これは、ユーザ定義エラーの範囲内にあり、クライアントにメッセージが送信されます。

```
RAISERROR 99999 'Invalid entry for this  
column: %1!', @val
```

使用法

(先頭に戻る) (476 ページ)

error-number パラメータと *format-string* パラメータの間には、カンマはありません。カンマの後にある最初の項目は、引数リストの最初の項目として解釈されます。

SQL Server または Adaptive Server の **RAISERROR** 文でサポートされる拡張値は、SAP Sybase IQ ではサポートされていません。

中間レベルの **RAISERROR** のステータスとコード情報は、プロシージャの終了後に失われます。結果を返す際に **RAISERROR** とともにエラーが発生した場合は、エラー情報が返され、**RAISERROR** 情報は失われます。アプリケーションでは、異なるいくつかの実行ポイントでグローバル変数 @@error を調べることによって、中間の **RAISERROR** ステータスを問い合わせることができます。

標準

(先頭に戻る) (476 ページ)

- SQL – ISO/ANSI SQL 文法の Transact-SQL 拡張。
- SAP Sybase Database 製品 - Adaptive Server でサポートされています。

パーミッション

(先頭に戻る) (476 ページ)

データベースに接続しておく必要があります。

参照：

- CONTINUE_AFTER_RAISERROR オプション [TSQL] (624 ページ)
- ON_TSQL_ERROR オプション [TSQL] (737 ページ)

READ 文 [Interactive SQL]

Interactive SQL (**dbisql**) 文をファイルから読み込みます。

クイックリンク：

「パラメータ」 (478 ページ)

「例」 (479 ページ)

「標準」 (480 ページ)

「パーミッション」 (480 ページ)

構文

```
READ [ ENCODING encoding ] filename [ parameter ] ...
```

パラメータ

(先頭に戻る) (478 ページ)

- **ENCODING** – ファイルの読み込みに使用するエンコードを指定するための識別子または文字列。

READ 文は、ファイルを読み込むときにエスケープ文字を処理しません。ファイル全体が指定されたコードであると想定します。Interactive SQL を実行するとき、データの読み込みに使用されるエンコードは、次の順序で決定されます。

1. ENCODING 句で指定されたエンコード (この句が指定されている場合)。
 2. ファイル内でバイトオーダーマーク (BOM) で指定されたエンコード (BOM が指定されている場合)。
 3. default_isql_encoding オプションで指定されたエンコード (このオプションが設定されている場合)。
 4. 実行しているプラットフォームのデフォルトのエンコード。英語版 Windows コンピュータでは、デフォルトのエンコードは 1252 です。
- **filename** – *filename* にファイル拡張子がない場合、Interactive SQL は同じファイル名で拡張子 .sql を持つものを検索します。

filename に絶対パスが含まれていない場合でも、Interactive SQL はファイルを検索します。filename の場所は、READ 文の場所に基づいて次のように決定されます。

- READ 文が Interactive SQL で直接実行された場合、Interactive SQL は、まず、Interactive SQL の実行場所のディレクトリを基準に filename のパスを解決しようとします。これで解決できなかった場合、Interactive SQL は、環境変数 SQLPATH で指定されたディレクトリで filename を検索し、次に環境変数 PATH で指定されたディレクトリで検索を行います。
- READ 文が外部ファイル(.sql ファイルなど)に含まれている場合、Interactive SQL は、まず、その外部ファイルのロケーションを基準に、filename のパスを解決しようとします。これで解決できなかった場合、Interactive SQL は、Interactive SQL の実行場所のディレクトリを基準に、filename を検索します。引き続き失敗した場合、Interactive SQL は、環境変数 SQLPATH で指定されたディレクトリで検索を行い、さらに環境変数 PATH で指定されたディレクトリで検索を行います。
- **parameters** – SQL スクリプトファイル名の後にリストできます。これらのパラメータは、文ファイルの先頭の PARAMETERS 文で指定したパラメータに対応します。

パラメータ名は角カッコで囲む必要があります。Interactive SQL は、ソースファイルに含まれている { *parameter-name* } (*parameter-name* は該当するパラメータ名)を検出するたびに、対応するパラメータを代入します。

スクリプトファイルに渡すパラメータには、識別子、数値、引用符付きの識別子、または文字列を指定できます。パラメータが引用符で囲まれている場合は、代入時に引用符がテキストに挿入されます。識別子、数値、または文字列以外のパラメータ (スペースまたはタブを含む) は、角カッコ ([]) で囲む必要があります。このようにすることで、スクリプトファイル内の任意のテキストを置き換えることができます。

十分なパラメータがスクリプトファイルに渡されない場合、Interactive SQL は足りないパラメータの値を要求するメッセージを表示します。

Interactive SQL で reload.sql ファイルを実行する場合は、パラメータとして暗号化キーを指定する必要があります。READ 文にキーを指定していない場合、Interactive SQL でキーの入力を要求するプロンプトが表示されます。

例

(先頭に戻る) (478 ページ)

- **例 1** – この例では、架空のファイル status.rpt と birthday.sql を読み込み、パラメータ値をファイル内の変数に渡します。

```
READ status.rpt '160'
```

```
READ birthday.sql [>= '1988-1-1'] [<= '1988-1-30']
```

SQL 文

- **例2**—この例では、PARAMETERS 句を使用してパラメータをスクリプトファイルに渡します。

```
[test1.sql]
PARAMETERS par1, par2;

BEGIN
DECLARE v_par1 int;
DECLARE v_par2 varchar(200)

SET v_par1 = {par1};
SET v_par2 = {par2};

MESSAGE STRING('PAR1 Value: ', v_par1 ) TO CLIENT;
MESSAGE STRING('PAR2 Value: ', v_par2 ) TO CLIENT;

END;

(USR1)> READ test1.sql 123 '041028'
PAR1 Value: 123
PAR2 Value: 041028
```

注意：2つ目のパラメータ値の041028は引用符で囲みます。これは、*v_par2*が文字データ型として宣言されているためです。

標準

(先頭に戻る) (478 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - なし。

パーミッション

(先頭に戻る) (478 ページ)

なし

参照：

- DEFAULT_ISQL_ENCODING オプション [Interactive SQL] (648 ページ)
- PARAMETERS 文 [Interactive SQL] (467 ページ)

REFRESH TEXT INDEX 文

テキストインデックスを更新します。

クイックリンク：

「パラメータ」 (481 ページ)

「例」 (482 ページ)

「使用法」 (482 ページ)

「標準」 (483 ページ)

「パーミッション」 (483 ページ)

構文

```
REFRESH TEXT INDEX text-index-name ON [ owner.]table-name
  [ WITH {
    ISOLATION LEVEL isolation-level
    | EXCLUSIVE MODE
    | SHARE MODE } ]
  [ FORCE { BUILD | INCREMENTAL } ]
```

パラメータ

(先頭に戻る) (480 ページ)

- **WITH** – 更新時に基礎となるベーステーブルに対して取得するロックの種類を指定します。取得したロックの種類によって、テキストインデックスの格納方法とトランザクションの同時実行性への影響が決まります。**WITH** 句を指定しなかった場合、デフォルトは、接続に設定されている独立性レベルに関係なく、**WITH ISOLATION LEVEL READ UNCOMMITTED** になります。

次の **WITH** 句オプションを指定できます。

- **ISOLATION LEVEL 独立性レベル** – **WITH ISOLATION LEVEL** は、更新操作の実行時に独立性レベルを変更する場合に使用します。接続の元の独立性レベルは、文の実行終了時にリストアされます。
- **EXCLUSIVE MODE** – **WITH EXCLUSIVE MODE** は、独立性レベルを変更しないが、基礎となるテーブルにコミットされたデータと矛盾しないようにデータを確実に更新する場合に使用します。**WITH EXCLUSIVE MODE** を使用すると、基礎となるベーステーブルに排他テーブルロックが設定され、更新操作が完了するまで、基本となるテーブルに対する問い合わせ、更新、その他のアクションを、他のトランザクションから実行できなくなります。テーブルのロックを取得できない場合、更新操作は失敗し、エラーが返されます。
- **SHARE MODE** – **WITH SHARE MODE** を使用すると、更新操作の実行中に他のトランザクションからの基礎となるテーブルへの読み込みアクセスが許可されます。この句を指定すると、更新操作が実行される前に基本となるベーステーブルに対する共有テーブルロックが取得され、更新操作が完了するまで保持されます。

- **FORCE** – この句は、更新方法の指定に使用します。この句を指定しなかった場合、データベースサーバは、テーブルの変更の割合に基づいて、増分更新するか、または完全再構築するかどうかを決定します。
- **FORCE BUILD** – テキストインデックスを再作成することで更新します。この句は、テキストインデックスを強制的に完全再構築する場合に使用します。
- **FORCE INCREMENTAL** – 基礎となるテーブルでの変更内容にのみ基づいてテキストインデックスを更新します。基本となるテーブルに加えられた更新の量が多くない場合、増分更新は短時間で完了します。この句を使用すると、テキストインデックスが強制的に増分更新されます。

増分更新では、エントリが削除されていても、テキストインデックスからは削除されません。このため、テキストインデックスのサイズは、現在のデータと履歴データを含めた予想サイズよりも大きくなる場合があります。通常、この問題は、常に **FORCE INCREMENTAL** 句を使用して手動で更新されているテキストインデックスで発生します。自動的に更新されるテキストインデックスでは、履歴データは、テキストインデックスの合計サイズの 50 % に達すると、自動的に削除されます。

例

(先頭に戻る) (480 ページ)

- **例 1** – `MarketingTextIndex` という名前の架空のテキストインデックスを更新し、その再構築を強制します。

```
REFRESH TEXT INDEX MarketingTextIndex ON
GROUPO.MarketingInformation
FORCE BUILD;
```

使用法

(先頭に戻る) (480 ページ)

この文は、**MANUAL REFRESH** または **AUTO REFRESH** と定義されているテキストインデックスでのみ使用できます。

FORCE 句を使用した場合は、`sa_text_index_stats` システムプロシージャの結果を確認して、完全再構築 (**FORCE BUILD**) または増分更新 (**FORCE INCREMENTAL**) のどちらが最適かを判定できます。

IMMEDIATE REFRESH として定義されているテキストインデックスに対して **REFRESH TEXT INDEX** 文を実行することはできません。

MANUAL REFRESH テキストインデックスの場合は、`sa_text_index_stats` システムプロシージャを使用してテキストインデックスを更新するかどうかを判断します。`pending_length` を `doc_length` で除算して求められるパーセンテージを、再表示が必

要かどうかを判断するうえでの目安にします。必要な再構築の種類を判断する場合も、`deleted_length` と `doc_count` を使って同じ処理を行います。

文またはトランザクションのスナップショットを使用する、`WITH HOLD` 句を使用して開かれたカーソルが存在する場合、この文は実行できません。

標準

(先頭に戻る) (480 ページ)

ANSI SQL – 準拠レベル：Transact-SQL 拡張。

パーミッション

(先頭に戻る) (480 ページ)

次のいずれかが必要です。

- ALTER ANY INDEX システム権限
- ALTER ANY OBJECT システム権限
- そのテーブルに対する REFERENCES 権限
- テーブルを所有していること

RELEASE SAVEPOINT 文

現在のトランザクション内でセーブポイントを解放します。

クイックリンク：

「パラメータ」 (483 ページ)

「使用法」 (484 ページ)

「標準」 (484 ページ)

「許可」 (484 ページ)

構文

```
RELEASE SAVEPOINT [ savepoint-name ]
```

パラメータ

(先頭に戻る) (483 ページ)

- **savepoint-name** – 現在のトランザクションの **SAVEPOINT** 文で指定された識別子です。 *savepoint-name* を省略すると、最新のセーブポイントが解放されます。

使用法

(先頭に戻る) (483 ページ)

セーブポイントを解放すると、どの種類の **COMMIT** も実行されません。現在アクティブなセーブポイントのリストから、そのセーブポイントが削除されるだけです。

標準

(先頭に戻る) (483 ページ)

- SQL — ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Adaptive Server ではサポートされていません。ネストされたトランザクションを使用すると、Adaptive Server と互換性のある方法で同様の機能を使用できます。

パーミッション

(先頭に戻る) (483 ページ)

現在のトランザクション内の **SAVEPOINT** に対応するパーミッションを持っている必要があります。

参照：

- ROLLBACK TO SAVEPOINT 文 (519 ページ)
- SAVEPOINT 文 (521 ページ)

REMOVE 文

クラス、パッケージ、または JAR ファイルをデータベースから削除します。削除されたクラスは、変数型として利用できなくなります。クラス、パッケージ、または JAR を削除するには、それがインストールされている必要があります。

クイックリンク：

「パラメータ」 (485 ページ)

「例」 (485 ページ)

「標準」 (485 ページ)

「許可」 (485 ページ)

構文

```
REMOVE JAVA classes_to_remove
```



```

classes_to_remove
{ CLASS java_class_name [, java_class_name ]...
| PACKAGE java_package_name [, java_package_name ]...
| JAR jar_name [, jar_name ]... [ RETAIN CLASSES ] }

```

パラメータ

(先頭に戻る) (484 ページ)

- **java_class_name** – 削除される 1 つまたは複数の Java クラスの名前です。これらのクラスは、現在のデータベースにインストールされている必要があります。
- **java_package_name** – 削除される 1 つまたは複数の Java パッケージの名前です。これらのパッケージは現在のデータベース内にあるパッケージ名でなければなりません。
- **jar_name** – 最大長 255 の文字列値です。 *jar_name* のそれぞれは、現在のデータベースに保持されている JAR の *jar_name* と同じある必要があります。 *jar_name* と同じかどうかは、SQL システムの文字列比較ルールによって決定されます。
- **RETAIN CLASSES** – 指定した JAR がデータベースに保持されなくなり、保持されたクラスは JAR の関連付けがなくなります。 **RETAIN CLASSES** を指定した場合、これが **REMOVE** 文の唯一のアクションになります。

例

(先頭に戻る) (484 ページ)

- **例 1** – 現在のデータベースから "Demo" という名前の Java クラスを削除します。

```
REMOVE JAVA CLASS Demo
```

標準

(先頭に戻る) (484 ページ)

- SQL — ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Adaptive Server ではサポートされていません。ネストされたトランザクションを使用すると、Adaptive Server と互換性のある方法で同様の機能を使用できます。

パーミッション

(先頭に戻る) (484 ページ)

次のいずれかが必要です。

- **MANAGE ANY EXTERNAL OBJECT** システム権限

- オブジェクトの所有者である。

RESIGNAL 文

例外条件の信号を送り返します。

クイックリンク：

「例」 (486 ページ)

「使用法」 (486 ページ)

「標準」 (486 ページ)

「許可」 (487 ページ)

構文

```
RESIGNAL [ exception-name ]
```

例

(先頭に戻る) (486 ページ)

- **例1**–次のコードフラグメントは、カラムが見つからない (Column Not Found) を除く、すべての例外をアプリケーションに返します。

```
...  
DECLARE COLUMN_NOT_FOUND EXCEPTION  
        FOR SQLSTATE '52003';  
...  
EXCEPTION  
WHEN COLUMN_NOT_FOUND THEN  
    SET message='Column not found' ;  
WHEN OTHERS THEN  
    RESIGNAL ;
```

使用法

(先頭に戻る) (486 ページ)

例外ハンドラの中で **RESIGNAL** を使用して、まだアクティブな例外を持つ複合文を終了するか、別の指定された例外のレポートを終了できます。例外は、別の例外ハンドラによって処理されるか、またはアプリケーションに返されます。

RESIGNAL の前の例外ハンドラによる動作は取り消されます。

標準

(先頭に戻る) (486 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - Adaptive Server ではサポートされていません。
Transact-SQL プロシージャのエラー処理は、**RAISERROR** 文を使用して実行されます。

パーミッション

(先頭に戻る) (486 ページ)

なし

参照：

- BEGIN ... END 文 (108 ページ)
- SIGNAL 文 (550 ページ)

RESTORE DATABASE 文

1 つまたは複数のアーカイブデバイスから SAP Sybase IQ データベースバックアップをリストアします。

クイックリンク：

「パラメータ」 (488 ページ)

「例」 (490 ページ)

「使用法」 (492 ページ)

「標準」 (494 ページ)

「許可」 (494 ページ)

構文

構文 1

```
RESTORE DATABASE 'db_file'
    'archive_device' [ FROM 'archive_device' ]...
... [ CATALOG ONLY ]
... [ KEY key_spec ]
... [ [ RENAME logical-dbfile-name TO 'new-dbspace-path']...
    | VERIFY [ COMPATIBLE ] ]
```

構文 2

```
RESTORE DATABASE 'database-name'
    [ restore-option ... ]
    FROM 'archive_device' ...
```

restore-option

```

READONLY dbspace-or-file [, ... ]
KEY key_spec
RENAME file-name TO new-file-path ...

```

パラメータ

(先頭に戻る) (487 ページ)

- **db_file** – リストアするデータベースの相対パスまたは絶対パス。カタログストアファイルの場合、元のロケーションか、または新しいロケーションを指定できます。
- **key_spec** – 大文字と小文字を混ぜた英数字、および特殊文字で構成される引用符文字列。キーをコマンドシェルによる解読または変更から保護するために必要です。
- **FROM** – リストアに使用する *archive_device* の名前を、一重引用符で囲んで指定します。複数のアーカイブデバイスを使用している場合、それぞれに別々の **FROM** 句を使用して指定します。カンマで区切ったリストは使用できません。アーカイブデバイスは異なるものでなければなりません。入力デバイスに関して SAP Sybase IQ が試みる並行処理の量は、**FROM** 句の数によって決まります。

バックアップ/リストア API の DLL 版では、アーカイブデバイスのオープン時に DLL に渡す引数を指定できます。サードパーティの実装の場合は、*archive_device* 文字列は次のフォーマットになります。

```
'DLLIdentifier::vendor_specific_information'
```

具体例：

```
'spsc::workorder=12;volname=ASD002'
```

archive_device 文字列の最大長は 1023 バイトです。DLLIdentifier 部分は 1 から 30 バイトの長さで、英数字とアンダースコアの文字だけを使用できます。文字列の *vendor_specific_information* 部分は、内容の検査なしでサードパーティの実装に渡されます。

注意： この構文を SAP Sybase IQ で使用して動作確認されているサードパーティ製品は、ごく一部に限られます。使用上のその他の指示や制約については、『リリースノート』を参照してください。サードパーティ製品を使用して SAP Sybase IQ データベースをバックアップする場合は、その製品が動作確認されたものであるかどうかを事前に確認してください。『リリースノート』、または Technical Documents にある SAP Sybase IQ 製品に対する SAP Sybase Certification Reports を参照してください。

バックアップ/リストア API の SAP Sybase IQ の実装の場合、テープデバイス名またはファイル名以外の情報を指定する必要はありません。ただし、ディスクデバイスを使用する場合、バックアップ上で指定したものと同数のアクティブデバイスをリストアで指定する必要があります。同数を指定しない場合、

バックアップの実行に使用した数と異なる数のリストアデバイスが存在することになります。UNIX システム用に非リワインディングテープデバイスを指定する SAP Sybase IQ API DLL 用のアーカイブデバイスの具体例を示します。

```
'/dev/rmt/0n'
```

- **CATALOG ONLY** – アーカイブメディアからバックアップヘッダレコードだけをリストアします。
- **RENAME** – SAP Sybase IQ データベースファイルを新しい場所にリストアします。SYSFILE テーブル内と同じ表記で、移動する各 *dbspace-name* を指定します。また、*new-dbspace-path* を新しいローパーティションとして、またはその DB 領域の新しいフルパス名または相対パス名として指定します。

相対パスを使用しデータベースファイルを作成した場合、ファイルはデフォルトでカタログストアファイル (SYSTEM DB 領域) に相対的なロケーションにリストアされ、RENAME 句は必要ありません。絶対パスを使用してデータベースファイルを作成し、ファイルに RENAME 句を指定しない場合、元のロケーションにリストアされます。

RENAME 句の相対パス名は、データベースまたは DB 領域を作成したときと同じように機能します。メイン IQ ストアの DB 領域、テンポラリストアの DB 領域、メッセージログは、*db_file* (カタログストア) のロケーションを基準にしてリストアされます。また、ユーザが作成した IQ ストアの DB 領域は、メイン IQ DB 領域が格納されたディレクトリを基準にしてリストアされます。

RENAME 句を使用して、カタログストアを保持する SYSTEM DB 領域を移動しないでください。カタログストアと、カタログストアの場所に基づいて作成されたファイルのうち、RENAME 句に指定されていないファイルを移動するには、*db_file* パラメータで新しい場所を指定します。

- **VERIFY [COMPATIBLE** – 指定された SAP Sybase IQ データベースバックアップアーカイブについて、フルバックアップ、インクリメンタルバックアップ、フルバックアップ以降のインクリメンタルバックアップ、または仮想バックアップを検証するようサーバに指示します。バックアップは SAP Sybase IQ バージョン 12.6 以降である必要があります。検証プロセスでは、リストアプロセスでチェックされても書き込み操作が実行されないエラーと同じエラーがないかどうか、指定されたアーカイブをチェックします。ステータスメッセージと検出されたエラーは、すべてサーバログファイルに書き込まれます。

RENAME 句を VERIFY 句と一緒に使用することはできません。エラーがレポートされます。

バックアップ検証プロセスは、データベースのホストとは異なるホストで実行できます。**RESTORE DATABASE VERIFY** を実行するには、BACKUP DATABASE システム権限が必要です。

VERIFY 句とともに COMPATIBLE 句が指定されている場合、インクリメンタルアーカイブと既存のデータベースファイルとの互換性がチェックされます。**RESTORE DATABASE...VERIFY COMPATIBLE** が呼び出されたシステム上にデータベースファイルが存在しない場合は、エラーが返されます。フルバックアップの検証時に COMPATIBLE を指定した場合、このキーワードは無視されます。フルバックアップのリストア中に、互換性チェックを実行する必要はありません。

フルバックアップ内の読み込み専用 DB 領域のバックアップを検証するには、データベースとログファイル (.db と .log) が必要です。これらのファイルがない場合は、READONLY *dbspace* 句を指定せずに **RESTORE DATABASE...VERIFY** を実行して、バックアップ全体を検証します。

注意： バックアップアーカイブの検証は、データベース一貫性チェッカ (DBCC) の verify モード (sp_iqcheckdb 'verify...') とは異なります。

RESTORE DATABASE VERIFY では、バックアップアーカイブの一貫性を検証して、そのアーカイブをリストアできるかどうかを確認するのに対し、DBCC ではデータベースのデータの一貫性を検証します。

sp_iqcheckdb 'verify...' を実行してから、バックアップしてください。一貫性のないデータベースをバックアップし、同じバックアップアーカイブからリストアした場合、**RESTORE DATABASE VERIFY** で検証の正常完了が報告された場合でも、データは一貫性のない状態のままになっています。

例

(先頭に戻る) (487 ページ)

- 例 1** – 次の UNIX の例は、iqdemo データベースを、Sun Solaris プラットフォームのテープデバイス /dev/rmt/0 と /dev/rmt/2 からリストアします。Solaris では、テープからリストアするときにリワインディングデバイスを指定する必要があります。したがって、デバイス名に続けて n の文字を入力しないでください。入力すると、"no rewind on close" 機能が指定されます。**RESTORE DATABASE** でこの機能を指定するには、使用する UNIX プラットフォームに適した命名規則を使用してください (Windows はこの機能をサポートしていません)。

```
RESTORE DATABASE 'iqdemo'
FROM '/dev/rmt/0'
FROM '/dev/rmt/2'
```

- 例 2** – marvin という名前を持ち、キー *is!seCret* で暗号化されたデータベースをリストアします。

```
RESTORE DATABASE 'marvin'
FROM 'marvin_bkup_file1'
```

```
FROM 'marvin_bkup_file2'
FROM 'marvin_bkup_file3'
KEY 'is!seCret'
```

- **例 3** – 次の例は、**BACKUP DATABASE** 文と 2 種類の **RESTORE DATABASE** 文の構文を示します(この例では、説明のために iqdemo データベースのオブジェクトを使用します。iqdemo には、iq_main というサンプルのユーザ DB 領域が含まれていますが、この DB 領域はお使いのデータベースには存在しない場合があります)。

次の **BACKUP DATABASE** 文があるとします。

```
BACKUP DATABASE READONLY DBSPACES iq_main
TO '/system1/IQ16/demo/backup/iqmain'
```

DB 領域 iq_main をリストアするには、次のいずれかの **RESTORE DATABASE** 文を使用します。

```
RESTORE DATABASE 'iqdemo' READONLY DBSPACES iq_main
FROM '/system1/IQ16/demo/backup/iqmain'
```

または

```
RESTORE DATABASE 'iqdemo'
FROM '/system1/IQ16/demo/backup/iqmain'
```

選択的バックアップによって、すべての **READWRITE** DB 領域、または指定した読み込み専用 DB 領域または dbfile がバックアップされます。選択的バックアップは、フルバックアップまたはインクリメンタルバックアップのサブタイプです。

注意事項：

- **READONLY** 選択的バックアップを使用すると、(上記の 2 番目の例のように) このバックアップからすべてのオブジェクトをリストアできます。
- 包括的なバックアップを使用すると、読み込み専用ファイルと DB 領域ファイルを選択的にリストアできます。
- 複数の読み込み専用ファイルと DB 領域の **READONLY** 選択的バックアップを使用すると、読み込み専用ファイルと DB 領域のサブセットを選択的にリストアできます。「パーミッション」を参照してください。
- 読み込み専用バックアップをリストアできるのは、読み込み専用ファイルがバックアップ以降に変更されていない場合に限られます。DB 領域を読み込み/書き込みに再度設定すると、データベースの読み込み/書き込みの全部分を DB 領域が読み込み専用で設定されていた時点でリストアする場合を除き、読み込み専用のバックアップは無効になります。
- どちらのバックアップサブタイプ(選択的または非選択的)を使用するかを決定し、決定したサブタイプを継続的に使用します。非選択的バックアップから選択的バックアップに、またはその逆に切り替える場合は、すべて

の変更がバックアップされるようにするために、新しいサブタイプに切り替える前に必ず非選択的フルバックアップを行う必要があります。

- **例 4** – 次の例は、書き込み操作を実行せずに、**VERIFY** 句を使用してデータベースアーカイブを検証する構文を示します。

```
RESTORE DATABASE <database_name.db>
FROM '/sys1/dump/dmp1'
FROM '/sys1/dump/dmp2'
VERIFY
```

検証する場合は、「データベース名がユニークではありません。」エラーが発生しないようにするため、別のデータベース名を指定します。たとえば、元のデータベースが iqdemo.db の場合、代わりに iq_demo_new.db を使用します。

```
RESTORE DATABASE iqdemo_new.db FROM iqdemo.bkp VERIFY
```

使用法

(先頭に戻る) (487 ページ)

RESTORE DATABASE コマンドを実行するには、**SERVER OPERATOR** システム権限を持つユーザがデータベースに排他的アクセスを行う必要があります。この排他的アクセスは **-gd** スイッチを DBA に設定することで実行できます。この設定はサーバエンジンの起動時のデフォルトです。

データベースを起動する前に **RESTORE DATABASE** コマンドを発行します (utility_db データベースに接続する必要があります)。バックアップの種類に応じた **RESTORE DATABASE** コマンドの指定を完了すると、そのデータベースは使用できる状態になります。データベースは、リストアした最新バックアップ内の最初の暗黙的な **CHECKPOINT** の終了時の状態になります。これで、**START DATABASE** を指定すると、他のユーザがリストアしたデータベースにアクセスできるようになります。

RESTORE DATABASE コマンド全体の最大サイズは、すべての句を含めて 32KB です。

ローデバイスにリストアするときは、リストアする DB 領域を書き込めるだけの大きさがそのデバイスにあるかどうかを確認します。SAP Sybase IQ の **RESTORE DATABASE** はローデバイスのサイズをチェックし、DB 領域をリストアするために十分なサイズがなければエラーを返します。

BACKUP DATABASE では、フルバックアップまたはインクリメンタルバックアップを指定できます。インクリメンタルバックアップには、2 種類の選択肢があります。INCREMENTAL は最後のバックアップ (インクリメンタルまたはフル) 以降に変更またはコミットしたブロックのみをバックアップします。INCREMENTAL SINCE FULL は最後のフルバックアップ以降に変更があったブロックをすべて

バックアップします。フルバックアップのリストアの後に、いずれかのタイプのインクリメンタルバックアップが続く場合、連続する **RESTORE DATABASE** コマンド間でデータベースの変更を行うことはできません。この規則により、クラッシュリカバリを必要とするデータベース、または変更されたデータベース上で、インクリメンタルバックアップからのリストアは実行できません。ただし、そのようなデータベースに対しては、フルバックアップからリストアを実行してデータベースを上書きできます。

フルリストアを開始する前に、カタログストアファイル (デフォルト名 `dbname.db`) とトランザクションログファイル (デフォルト名 `dbname.log`) の 2 つのファイルを削除する必要があります。

インクリメンタルバックアップをリストアする場合、**RESTORE DATABASE** は、バックアップメディアセットへのアクセスが正しい順序で行われることを保証します。つまり、最後のフルバックアップテープセットを最初にリストアし、次に最初のインクリメンタルバックアップテープセット、次に新しいセットというような順序で、最新のインクリメンタルバックアップテープセットまで、順にリストアされます。SERVER OPERATOR システム権限を持つユーザが、INCREMENTAL SINCE FULL バックアップを発行した場合は、フルバックアップリケーションテープセットと最新 INCREMENTAL SINCE FULL バックアップテープセットのみが必要になりますが、INCREMENTAL SINCE FULL バックアップ以降に実行された INCREMENTAL バックアップが存在する場合はそれも適用する必要があります。

SAP Sybase IQ では、リストアの順序が正しいことが保証されます。正しくない場合はエラーが発生します。リストア中にこれ以外のエラーが発生した場合、データベースは破損 (corrupt)、使用不可 (unusable) とマーク付けされます。そのような破損したデータベースを修復するには、フルバックアップからのリストアを行ってから、その後の追加のインクリメンタルバックアップからのリストアを実行します。これらのバックアップのどれかで破損が発生した可能性があるため、新しいほうのバックアップを無視し、古いほうのセットを使用しなければならない場合があります。

アーカイブバックアップから読み込み専用ファイルまたは DB 領域をリストアする場合、**RESTORE DATABASE** 文の発行時に、データベースが実行中であっても構いません。また、管理者がデータベースに接続することもできます。バックアップに含まれている名前は、データベースシステムのテーブル情報と一致していれば、読み込み専用ファイルのパス名と一致している必要はありません。

READWRITE FILES ONLY またはすべてのファイルのバックアップを FULL、INCREMENTAL SINCE FULL、または INCREMENTAL でリストアするときは、データベースが実行することはできません。読み込み専用ファイルのバックアップをリストアする場合は、データベースの実行が可能な場合も、そうでない場合もあります。読み込み専用 DB 領域の指定のファイルをリストアする場合は、DB

領域がオフラインである必要があります。読み書き DB 領域の読み込み専用ファイルをリストアする場合は、DB 領域はオンラインまたはオフラインのどちらでもかまいません。リストアによって読み込み専用ファイルがクローズされ、ファイルのリストアを行い、リストアの終了時にこれらのファイルが再オープンされません。

DB 領域が同じ読み込み専用状態になっているかぎり、選択的リストアを使用し、読み込み専用 DB 領域をリストアできます。

RESTORE DATABASE のその他の注意点：

- ディスクへの **RESTORE DATABASE** では、アーカイブデバイスとしてローデバイスはサポートされません。
- SAP Sybase IQ はテープをその使用前にリワインドしません。リワインディングテープデバイスでは、使用後にテープをリワインドします。リストアを開始する前に各テープを SAP Sybase IQ データの開始位置に合わせます。
- バックアップやリストアの実行時に、アーカイブデバイスをオープンできず (メディアのロードが必要な場合など)、ATTENDED オプションが ON である場合、SAP Sybase IQ は次のテープをドライブに挿入できるよう 10 秒間待機してから再試行します。デバイスが正常に挿入されるか、[Ctrl + C] キーを押して操作を終了しないかぎり、無制限に再試行されます。
- [Ctrl] キーを押しながら、[C] キーを押すと、**RESTORE DATABASE** は失敗し、データベースがリストア開始前の状態に戻されます。
- ディスクストライピングを使用した場合、ストライプされたディスクは単一のデバイスとして処理されます。
- SYSTEM DB 領域の SYSFILE システムテーブルの file_name カラムは、リストア時に更新されません。SYSTEM DB 領域では、データベース作成時の名前が file_name カラムに常に反映されます。SYSTEM DB 領域のファイル名は、データベースファイルの名前です。

標準

(先頭に戻る) (487 ページ)

- SQL — ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Adaptive Server ではサポートされていません。

パーミッション

(先頭に戻る) (487 ページ)

この文を実行するのに必要なパーミッションは、以下のように、**-gu** サーバコマンドラインオプションを使用して設定します。

- **NONE** – この文を発行できるユーザはいません。
- **DBA** – SERVER OPERATOR システム権限が必要です。
- **UTILITY_DB** – utility_db データベースに接続できるユーザのみがこの文を発行できます。

参照：

- BACKUP DATABASE 文 (100 ページ)

RESUME 文

結果セットを返すプロシージャの実行を再開します。

クイックリンク：

「パラメータ」 (495 ページ)

「例」 (495 ページ)

「使用法」 (496 ページ)

「標準」 (496 ページ)

「許可」 (496 ページ)

構文

構文 1 – **dbisqlc** でサポートされています。

```
RESUME cursor-name
```

構文 2 – **dbisql** でサポートされています。

```
RESUME [ ALL ]
```

パラメータ

(先頭に戻る) (495 ページ)

- **cursor-name** – 識別子またはホスト変数

例

(先頭に戻る) (495 ページ)

- **例 1** – embedded SQL の例を示します。

```
EXEC SQL RESUME cur_employee;
```

および

```
EXEC SQL RESUME :cursor_var;
```

- **例 2 – dbisql** の例を示します。

```
CALL sample_proc() ;  
RESUME ALL;
```

使用法

(先頭に戻る) (495 ページ)

プロシージャは、次の結果セット (INTO 句のない **SELECT** 文) が見つかるまで実行されます。プロシージャが完了しても結果セットが見つからない場合は、**SQLSTATE_PROCEDURE_COMPLETE** 警告が設定されます。この警告は、**SELECT** 文に対してカーソルの **RESUME** を実行するときにも設定されます。

構文 1 – **dbisqlc** ではサポートされますが、**dbisql** (Interactive SQL) の場合や、SQL Anywhere JDBC ドライバを使用してデータベースに接続している場合はサポートされません。

構文 2 – **dbisql** でサポートされています。現在のプロシージャを再開します。ALL を指定しないで **RESUME** を実行すると、次の結果セットが表示されるか、またはほかに返される結果セットがないときは、プロシージャが完了します。**dbisql** では、**RESUME ALL** 文が、プロシージャ内のすべての結果セットを表示しないで循環し、その後プロシージャを完了します。これは主にプロシージャをテストするときに便利です。

標準

(先頭に戻る) (495 ページ)

- SQL — ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Adaptive Server ではサポートされていません。

パーミッション

(先頭に戻る) (495 ページ)

カーソルを事前にオープンしておく必要があります。

参照：

- DECLARE CURSOR 文 [ESQL] [SP] (299 ページ)

RETURN 文

関数またはプロシージャを無条件で終了し、オプションで値を返します。

RETURN に続く文は実行されません。

クイックリンク：

「パラメータ」 (497 ページ)

「例」 (497 ページ)

「使用法」 (498 ページ)

「標準」 (498 ページ)

「許可」 (498 ページ)

構文

```
RETURN [ ( expression ) ]
```

パラメータ

(先頭に戻る) (497 ページ)

- **expression** – 指定すると、関数またはプロシージャの値として *expression* の値を返します。

関数内では、この式のデータ型を関数の RETURN データ型と同じにします。

例

(先頭に戻る) (497 ページ)

- **例 1** – 3 つの数値の積を返します。

```
CREATE FUNCTION product ( a numeric,  
                          b numeric ,  
                          c numeric)  
RETURNS numeric  
BEGIN  
    RETURN ( a * b * c ) ;  
END
```

- **例 2** – 3 つの数値の積を計算します。

```
SELECT product (2, 3, 4)
```

```
product (2,3,4)  
24
```

SQL 文

- **例 3** – 無益で複雑なクエリの実行を回避します。

```
CREATE PROCEDURE customer_products
( in customer_id integer DEFAULT NULL)
RESULT ( id integer, quantity_ordered integer )
BEGIN
  IF customer_id NOT IN (SELECT ID FROM Customers)
  OR customer_id IS NULL THEN
    RETURN
  ELSE
    SELECT ID, sum(
      SalesOrderItems.Quantity )
    FROM Products,
      SalesOrderItems,
      SalesOrders
    WHERE SalesOrders.CustomerID = customer_id
    AND SalesOrders.ID = SalesOrderItems.ID
    AND SalesOrderItems.ProductID = Products.D
    GROUP BY Products.ID
  END IF
END
```

使用法

(先頭に戻る) (497 ページ)

プロシージャ内では、**RETURN** は Transact-SQL との互換性を確保するためのもので、整数のエラーコードを返すために使用されます。

標準

(先頭に戻る) (497 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - Transact-SQL プロシージャは、RETURN 文を使用して、整数のエラーコードを返します。

パーミッション

(先頭に戻る) (497 ページ)

なし

参照:

- BEGIN ... END 文 (108 ページ)
- CREATE PROCEDURE 文 (210 ページ)

REVOKE CHANGE PASSWORD 文

パスワードおよびシステム権限を管理する権限をユーザから削除します。

クイックリンク：

[「パラメータ」 \(499 ページ\)](#)

[「例」 \(499 ページ\)](#)

[「使用法」 \(500 ページ\)](#)

[「標準」 \(500 ページ\)](#)

[「パーミッション」 \(500 ページ\)](#)

構文

```
REVOKE [ ADMIN OPTION FOR ] CHANGE PASSWORD
  [(target_user_list
    | ANY
    | ANY WITH ROLES target_role_list )]
FROM userID [,...]
```

パラメータ

[\(先頭に戻る\) \(499 ページ\)](#)

- **target_user_list** – 被付与者が同一化の対象とする可能性のあるユーザ。このリストは、ログインパスワードを持つ既存のユーザまたはユーザ拡張ロールで構成される必要があります。リスト内の userID はカンマで区切ります。
- **ANY** – ログインパスワードを持つすべてのデータベースユーザが、各被付与者のパスワードを管理する潜在的ターゲットユーザになります。
- **ANY WITH ROLES target_role_list** – 各被付与者のターゲットロールのリスト。いずれかのターゲットロールを付与されたユーザは、各被付与者の潜在的ターゲットユーザになります。target_role_list は既存のロールで構成される必要があります。前述のロールを付与されたユーザはログインパスワードを持つデータベースユーザで構成される必要があります。複数の userID を指定する場合は、カンマで区切ります。
- **userID** – ログインパスワードを持つ既存のユーザまたはロールの名前になります。複数の userID はカンマで区切ります。

例

[\(先頭に戻る\) \(499 ページ\)](#)

- **例 1** – Joe が Sally または Bob のパスワードを管理できる権限を削除します。

```
REVOKE CHANGE PASSWORD (Sally, Bob) FROM Joe
```

- **例 2** – CHANGE PASSWORD システム権限が WITH ADMIN OPTION 句付きで Sam に付与されていた場合、この例では、Sam から、別のユーザに CHANGE PASSWORD システム権限を付与できる権限は削除されますが、元の **GRANT CHANGE PASSWORD** 文で指定されているユーザのパスワードの管理権限は Sam に残ります。ただし、CHANGE PASSWORD システム権限が WITH ADMIN ONLY OPTION 句付きで Sam に付与されていた場合、この例では、Sam からシステム権限のすべてのパーミッションが削除されます。

```
REVOKE ADMIN OPTION FOR CHANGE PASSWORD FROM Sam
```

使用法

(先頭に戻る) (499 ページ)

CHANGE PASSWORD システム権限が最初にどのように付与されているかによって、ADMIN OPTION FOR 句付きで CHANGE PASSWORD システム権限を取り消した場合、異なる結果になります。CHANGE PASSWORD システム権限が WITH ADMIN OPTION 句付きで付与されている場合、ADMIN OPTION FOR 句を REVOKE 文に含めると、CHANGE PASSWORD システム権限を管理する権限 (つまり、システム権限を別のユーザに付与できる権限) のみが取り消されます。他のユーザのパスワードを実際に管理する権限は残ります。ただし、CHANGE PASSWORD システム権限が WITH ADMIN ONLY OPTION 付きで付与されている場合、ADMIN OPTION FOR 句を REVOKE 文に含めることは、CHANGE PASSWORD システム権限全体を取り消すこととセマンティック上同じになります。最後に、CHANGE PASSWORD システム権限が WITH NO ADMIN OPTION 句付きで付与されている場合、ADMIN OPTION FOR 句を REVOKE 文に含めても、元々管理権限が付与されていないので、何も取り消されません。

付与されているユーザとロールの任意の組み合わせから CHANGE PASSWORD システム権限を取り消すことができます。

標準

(先頭に戻る) (499 ページ)

ANSI SQL – 準拠レベル： Transact-SQL 拡張。

パーミッション

(先頭に戻る) (499 ページ)

CHANGE PASSWORD システム権限が管理権限付きで付与されている必要があります。

参照：

- GRANT CHANGE PASSWORD 文 (383 ページ)

REVOKE CONNECT 文

ユーザをデータベースから削除します。

クイックリンク：

「パラメータ」 (501 ページ)

「使用法」 (501 ページ)

「標準」 (501 ページ)

「パーミッション」 (502 ページ)

構文

```
REVOKE CONNECT
FROM userID [,...]
```

パラメータ

(先頭に戻る) (501 ページ)

- **userID** – ログインパスワードを持つ既存のユーザまたはロールの名前になります。複数の userID はカンマで区切ります。

使用法

(先頭に戻る) (501 ページ)

ユーザ ID を追加および削除するには、**GRANT** 文または **REVOKE** 文ではなく、システムプロシージャまたは **CREATE USER** 文および **DROP USER** 文を使用します。

ユーザがテーブルなどのデータベースオブジェクトを所有している場合、そのユーザの接続権限を取り消すことはできません。**REVOKE** 文、または **sp_droplogin** ストアドプロシージャあるいは **sp_iqdroplogin** ストアドプロシージャでこれを行おうとすると、「ランタイムシステムでは、テーブルを所有しているユーザを削除することはできません。」のようなエラーが返されます。

標準

(先頭に戻る) (501 ページ)

ANSI SQL – 準拠レベル: Transact-SQL 拡張。

パーミッション

(先頭に戻る) (501 ページ)

MANAGE ANY USER システム権限が必要です。

注意: 別のユーザの **CONNECT** パーミッションまたはテーブルパーミッションを取り消すと、そのユーザはそのデータベースに接続できなくなります。

参照:

- GRANT CONNECT 文 (385 ページ)

REVOKE CREATE 文

指定されたユーザ ID から、指定された DB 領域に対する CREATE 権限を削除します。

クイックリンク:

「パラメータ」 (502 ページ)

「例」 (502 ページ)

「標準」 (503 ページ)

「パーミッション」 (503 ページ)

構文

```
REVOKE CREATE ON dbspace-name
FROM userID [, ...]
```

パラメータ

(先頭に戻る) (502 ページ)

- **userID** – ログインパスワードを持つ既存のユーザまたはロールの名前になります。複数の userID はカンマで区切ります。

例

(先頭に戻る) (502 ページ)

- **例 1** – ユーザ Smith から DB 領域 DspHist に対する CREATE 権限を取り消します。

```
REVOKE CREATE ON DspHist FROM Smith
```

- **例 2** – データベースからユーザ ID `fionat` の DB 領域 `DspHist` に対する `CREATE` 権限を取り消します。

```
REVOKE CREATE ON DspHist FROM fionat
```

標準

(先頭に戻る) (502 ページ)

ANSI SQL – 準拠レベル：Transact-SQL 拡張。

パーミッション

(先頭に戻る) (502 ページ)

`MANAGE ANY DBSPACE` システム権限が必要です。

参照：

- `GRANT CREATE` 文 (388 ページ)

REVOKE EXECUTE 文

`GRANT` 文を使用して付与された `EXECUTE` パーミッションを削除します。

クイックリンク：

「パラメータ」 (503 ページ)

「標準」 (503 ページ)

「パーミッション」 (504 ページ)

構文

```
REVOKE EXECUTE ON [ owner. ] procedure-name  
FROM userID [, ...]
```

パラメータ

(先頭に戻る) (503 ページ)

- **userID** – ログインパスワードを持つ既存のユーザまたはロールの名前になります。複数の `userID` はカンマで区切ります。

標準

(先頭に戻る) (503 ページ)

SQL 文

- SQL – 構文は永続的ストアドモジュール機能です。
- SAP Sybase Database 製品- 構文は Adaptive Server でサポートされています。ユーザ管理とセキュリティモデルは、Adaptive Server と SAP Sybase IQ では異なります。

パーミッション

(先頭に戻る) (503 ページ)

次のいずれかが必要です。

- プロシージャを所有している
- `MANAGE ANY OBJECT PRIVILEGE` システム権限が付与されている

参照：

- `GRANT EXECUTE` 文 (391 ページ)

REVOKE INTEGRATED LOGIN 文

`GRANT` 文を使用して付与された `INTEGRATED LOGIN` パーミッションを削除します。

クイックリンク：

「パラメータ」 (504 ページ)

「標準」 (504 ページ)

「パーミッション」 (505 ページ)

構文

```
REVOKE INTEGRATED LOGIN  
FROM userID [, ...]
```

パラメータ

(先頭に戻る) (504 ページ)

- `userID` – ログインパスワードを持つ既存のユーザまたはロールの名前になります。複数の `userID` はカンマで区切ります。

標準

(先頭に戻る) (504 ページ)

ANSI SQL – 準拠レベル：Transact-SQL 拡張。

パーミッション

(先頭に戻る) (504 ページ)

MANAGE ANY USER システム権限が必要です。

参照：

- GRANT INTEGRATED LOGIN 文 (392 ページ)

REVOKE KERBEROS LOGIN 文

GRANT 文を使用して付与された KERBEROS LOGIN パーミッションを削除します。

クイックリンク：

「パラメータ」 (505 ページ)

「標準」 (505 ページ)

「パーミッション」 (505 ページ)

構文

```
REVOKE KERBEROS LOGIN  
FROM userID [, ...]
```

パラメータ

(先頭に戻る) (505 ページ)

- **userID** – ログインパスワードを持つ既存のユーザまたはロールの名前になります。複数の userID はカンマで区切ります。

標準

(先頭に戻る) (505 ページ)

ANSI SQL – 準拠レベル： Transact-SQL 拡張。

パーミッション

(先頭に戻る) (505 ページ)

MANAGE ANY USER システム権限が必要です。

参照：

- GRANT KERBEROS LOGIN 文 (393 ページ)

REVOKE オブジェクトレベル権限文

GRANT 文を使用して付与されたオブジェクトレベル権限を削除します。

クイックリンク：

「パラメータ」 (506 ページ)

「例」 (507 ページ)

「標準」 (507 ページ)

「パーミッション」 (507 ページ)

構文

```
REVOKE { object-level-privilege [, ... ]
        [ owner.]table-name
        FROM userID [, ... ]
```

object-level-privilege

```
ALL [ PRIVILEGES ]
| ALTER
| DELETE
| INSERT
| LOAD
| REFERENCE [ ( column-name [, ... ] ) ]
| SELECT [ ( column-name [, ... ] ) ]
| TRUNCATE
| UPDATE [ ( column-name, ... ) ] }
```

パラメータ

(先頭に戻る) (506 ページ)

- **userID** – 既存ユーザまたは不変ロールの名前になります。リストは、ログインパスワードを持つ既存ユーザで構成される必要があります。リスト内の **userID** はカンマで区切ります。
- **ALL** – すべての権限をユーザに付与します。
- **ALTER** – ユーザが **ALTER TABLE** 文を使用してテーブルを変更できます。この権限は、ビューには使用できません。
- **DELETE** – ユーザがテーブルまたはビューからローを削除できます。
- **INSERT** – ユーザが指定のテーブルまたはビューにローを挿入できます。
- **LOAD** – ユーザが指定のテーブルまたはビューにデータをロードできます。

- **REFERENCES** – ユーザが、指定したテーブルのインデックス、および、指定したテーブルを参照する外部キーを作成できます。カラム名を指定した場合は、ユーザは指定したカラムだけを参照できます。カラムの **REFERENCES** 権限は、ビューに対しては付与できません。テーブルに対してのみ付与できます。
- **SELECT** – ユーザがビューまたはテーブルの情報を参照できます。カラム名を指定すると、ユーザは指定したカラムだけを参照できます。カラムの **SELECT** パーミッションは、ビューに対しては付与できません。テーブルに対してのみ付与できます。
- **TRUNCATE** – ユーザが指定のテーブルまたはビューをトランケートできます。
- **UPDATE** – ユーザがビューまたはテーブルのローを更新できます。カラム名を指定すると、ユーザは指定したカラムだけを更新できます。カラムの **UPDATE** 権限は、ビューに対しては付与できません。テーブルに対してのみ付与できます。テーブルを更新するには、ユーザはそのテーブルに対する **SELECT** 権限と **UPDATE** 権限の両方を保有している必要があります。

例

(先頭に戻る) (506 ページ)

- **例 1** – ユーザ Dave が Employees テーブルに挿入できないようにします。

```
REVOKE INSERT ON Employees FROM Dave
```

- **例 2** – ユーザ Dave が Employees テーブルを更新できないようにします。

```
REVOKE UPDATE ON Employees FROM Dave
```

標準

(先頭に戻る) (506 ページ)

- SQL - 構文は初級レベル機能です。
- SAP Sybase Database 製品 - 構文は Adaptive Server でサポートされています。

パーミッション

(先頭に戻る) (506 ページ)

次のいずれかが必要です。

- テーブルを所有している
- **GRANT OPTION** 句付きで **MANAGE ANY OBJECT PRIVILEGE** システム権限が付与されている

参照：

- GRANT オブジェクトレベル権限文 (389 ページ)

REVOKE ROLE 文

ロールのユーザメンバーシップ、またはロールを管理する権限を削除します。

クイックリンク：

「パラメータ」 (509 ページ)

「例」 (509 ページ)

「標準」 (509 ページ)

「パーミッション」 (509 ページ)

構文

```
REVOKE [ ADMIN OPTION FOR ] ROLE role_name [, ...]
      FROM grantee [, ...]
```

```
role_name
  dbo†††
  | diagnostics†††
  | PUBLIC†††
  | rs_systabgroup†††
  | SA_DEBUG†††
  | SYS†††
  | SYS_AUTH_SA_ROLE
  | SYS_AUTH_SSO_ROLE
  | SYS_AUTH_DBA_ROLE
  | SYS_AUTH_RESOURCE_ROLE
  | SYS_AUTH_BACKUP_ROLE
  | SYS_AUTH_VALIDATE_ROLE
  | SYS_AUTH_WRITEFILE_ROLE
  | SYS_AUTH_WRITEFILECLIENT_ROLE
  | SYS_AUTH_READFILE_ROLE
  | SYS_AUTH_READFILECLIENT_ROLE
  | SYS_AUTH_PROFILE_ROLE
  | SYS_AUTH_USER_ADMIN_ROLE
  | SYS_AUTH_SPACE_ADMIN_ROLE
  | SYS_AUTH_MULTIPLEX_ADMIN_ROLE
  | SYS_AUTH_OPERATOR_ROLE
  | SYS_AUTH_PERMS_ADMIN_ROLE
  | SYS_REPLICATE_ADMIN_ROLE†††
  | SYS_RUN_REPLICATE_ROLE†††
  | SYS_SPATIAL_ADMIN_ROLE†††
  | user-defined role name
```

^{†††}ADMIN OPTION FOR 句はシステムロールに対してはサポートされていません。

パラメータ

(先頭に戻る) (508 ページ)

- **role_name** – データベースにすでに存在している必要があります。複数のロール名を指定するときはカンマで区切ります。
- **userID** – ログインパスワードを持つ既存のユーザまたはロールの名前になります。複数の userID はカンマで区切ります。
- **ADMIN OPTION FOR** – 各 *userID* に、指定の *role_name* に対する管理権限が付与されている必要があります。

注意： そのロールが元々 WITH ADMIN ONLY OPTION 句で付与されていない限り、この句では、ロールのメンバーシップではなく、ロールの管理権限のみを取り消します。WITH ADMIN ONLY OPTION 句付きで付与されているロールに対しては、ADMIN OPTION FOR 句は、ロール内のメンバーシップを全部取り消すこととセマンティック上同じなので、この句は省略可能です。

例

(先頭に戻る) (508 ページ)

- **例 1** – ユーザ定義 (スタンドアロン) ロール Role1 を User1 から取り消します。

```
REVOKE ROLE Role1 FROM User1
```

このコマンドを実行した後、User1 には、Role1 に付与されているシステム権限を使用して承認済みタスクを実行する権限がなくなります。

- **例 2** – User1 から、互換ロール SYS_AUTH_WRITEFILE_ROLE を管理する権限を取り消します。

```
REVOKE ADMIN OPTION FOR ROLE SYS_AUTH_WRITEFILE_ROLE FROM User1
```

SYS_AUTH_WRITEFILE_ROLE によって付与されている認証済みタスクを実行する権限は User1 に残ります。

標準

(先頭に戻る) (508 ページ)

- SQL – その他の構文は、ISO/ANSI SQL 文法のベンダ拡張です。
- SAP Sybase Database 製品 - 構文は Adaptive Server でサポートされています。

パーミッション

(先頭に戻る) (508 ページ)

SQL 文

以下のロールを取り消すには `MANAGE ROLES` システム権限が必要です。

- `diagnostics`
- `dbo`
- `PUBLIC`
- `rs_systabgroup`
- `SA_DEBUG`
- `SYS`
- `SYS_RUN_REPLICATE_ROLE`
- `SYS_SPATIAL_ADMIN_ROLE`

以下のロールを取り消すには、そのロールに対する管理権限が必要です。

- `SYS_AUTH_SA_ROLE`
- `SYS_AUTH_SSO_ROLE`
- `SYS_AUTH_DBA_ROLE`
- `SYS_AUTH_RESOURCE_ROLE`
- `SYS_AUTH_BACKUP_ROLE`
- `SYS_AUTH_VALIDATE_ROLE`
- `SYS_AUTH_WRITEFILE_ROLE`
- `SYS_AUTH_WRITEFILECLIENT_ROLE`
- `SYS_AUTH_READFILE_ROLE`
- `SYS_AUTH_READFILECLIENT_ROLE`
- `SYS_AUTH_PROFILE_ROLE`
- `SYS_AUTH_USER_ADMIN_ROLE`
- `SYS_AUTH_SPACE_ADMIN_ROLE`
- `SYS_AUTH_MULTIPLEX_ADMIN_ROLE`
- `SYS_AUTH_OPERATOR_ROLE`
- `SYS_AUTH_PERMS_ADMIN_ROLE`
- <ユーザ定義ロール名>

参照：

- `GRANT ROLE` 文 (394 ページ)
- `GRANT` システム権限文 (402 ページ)

REVOKE SET USER 文

別のユーザに同一化できる権限と、`SET USER` システム権限を管理する権限を削除します。

クイックリンク：

「パラメータ」 (511 ページ)

「例」 (511 ページ)

「使用法」 (512 ページ)

「標準」 (512 ページ)

「パーミッション」 (512 ページ)

構文

```
REVOKE [ ADMIN OPTION FOR ] SETUSER
    (target_user_list
     | ANY
     | ANY WITH ROLES target_role_list )
FROM userID [,...]
```

パラメータ

(先頭に戻る) (510 ページ)

- **target_user_list** – ログインパスワードを持つ既存のユーザで構成される必要があります。被付与者ユーザによる同一化の対象ではなくなった可能性のあるターゲットユーザの潜在的リストです。リスト内のユーザ ID はカンマで区切ります。
- **ANY** – 各被付与者のターゲットユーザの潜在的リストは、ログインパスワードを持つすべてのデータベースユーザで構成されます。
- **ANY WITH ROLES target_role_list** – *target_role_list* は既存のロールで構成される必要があります。各被付与者のターゲットユーザの潜在的リストは、*target_role_list* 内のロールのサブセットが付与されているログインパスワードを持つデータベースユーザで構成される必要があります。リスト内のロールはカンマで区切ります。
- **userID** – 各 *userID* は、既存のユーザまたは不変ロールの名前になります。リストは、ログインパスワードを持つ既存ユーザで構成される必要があります。リスト内の *userID* はカンマで区切ります。

例

(先頭に戻る) (510 ページ)

- **例 1** – Bob が Sally または Bob に同一化できないようにします。

```
REVOKE SET USER (Sally, Bob) FROM Bob
```

- **例 2** – SET USER システム権限が WITH ADMIN OPTION 句付きで Sam に付与されていた場合、この例により、Sam から、別のユーザに SET USER システム権

限を付与できる権限は削除されますが、指定されているユーザへの同一化権限は Sam に残ります。ただし、SET USER システム権限が WITH ADMIN ONLY OPTION 句付きで Sam に付与されていた場合、この例により、Sam からシステム権限のすべてのパーミッションが削除されます。

```
REVOKE ADMIN OPTION FOR SET USER FROM Sam
```

使用法

(先頭に戻る) (510 ページ)

SET USER システム権限が最初にどのように付与されているかによって、ADMIN OPTION FOR 句付きで SET USER システム権限を取り消した場合、異なる結果になります。SET USER システム権限が WITH ADMIN OPTION 句付きで付与されている場合、ADMIN OPTION FOR 句を REVOKE 文に含めると、SET USER システム権限を管理する権限 (つまり、システム権限を別のユーザに付与できる権限) のみを取り消されます。別のユーザーに実際に同一化する権限は残ります。ただし、SET USER システム権限が WITH ADMIN ONLY OPTION 付きで付与されている場合、ADMIN OPTION FOR 句を REVOKE 文に含めることは、SET USER システム権限全体を取り消すこととセマンティック上同じになります。最後に、SET USER システム権限が WITH NO ADMIN OPTION 句付きで付与されている場合、ADMIN OPTION FOR 句を REVOKE 文に含めても、元々管理システム権限が付与されていないので、何も取り消されません。

標準

(先頭に戻る) (510 ページ)

ANSI SQL – 準拠レベル：Transact-SQL 拡張。

パーミッション

(先頭に戻る) (510 ページ)

SET USER システム権限が管理権限付きで付与されている必要があります。

参照：

- GRANT SET USER 文 (399 ページ)

REVOKE システム権限文

特定のユーザから特定のシステム権限を削除し、権限を管理する権限も削除します。

クイックリンク：

「パラメータ」 (513 ページ)

「例」 (513 ページ)

「使用法」 (514 ページ)

「標準」 (514 ページ)

「パーミッション」 (514 ページ)

構文

```
REVOKE [ ADMIN OPTION FOR ] system_privilege_name [,...]
FROM userID [,...]
```

パラメータ

(先頭に戻る) (512 ページ)

- **system_privilege_name** – 既存のシステム権限になります。
- **userID** – ログインパスワードを持つ既存のユーザまたはロールの名前になります。複数の userID はカンマで区切ります。
- **ADMIN OPTION FOR** – 各 *system_privilege* が、管理権限付きで指定の各 *userID* に現在付与されている必要があります。

注意： この句では、システム権限の管理権限のみが取り消され、システム権限そのものは残ります。ただし、システム権限が **WITH ADMIN ONLY OPTION** 句付きで付与されていた場合は、**ADMIN OPTION FOR** 句により、システム権限が完全に取り消されます。このシナリオでは、管理権限を取り消すために **ADMIN OPTION FOR** 句の使用は必須ではありません。

例

(先頭に戻る) (512 ページ)

- **例 1** – ユーザ Jim の BACKUP DATABASE システム権限を取り消します。

```
REVOKE BACKUP DATABASE FROM Jim
```

- **例 2** – BACKUP DATABASE システム権限がユーザ Jim に **WITH ADMIN OPTION** 句付きで付与されている場合、この例により、ユーザ Jim から、BACKUP DATABASE システム権限を管理する権限が取り消されます。システム権限で承認されているタスクを実行する権限は残ります。ただし、BACKUP DATABASE システム権限が **WITH ADMIN ONLY OPTION** 句付きで Jim に付与されていた場合、この例では、ユーザ Jim からシステム権限のすべてのパーミッションが削除されます。

使用法

(先頭に戻る) (512 ページ)

システム権限が最初にどのように付与されているかによって、ADMIN OPTION FOR 句付きでシステム権限を取り消した場合、異なる結果になります。システム権限が WITH ADMIN OPTION 句付きで付与されている場合、ADMIN OPTION FOR 句を REVOKE 文に含めると、システム権限を管理する権限 (つまり、システム権限を別のユーザに付与できる権限) のみが取り消されます。システム権限を実際に使用する権限は残ります。ただし、システム権限が WITH ADMIN ONLY OPTION 付きで付与されている場合、ADMIN OPTION FOR 句を REVOKE 文に含めることは、システム権限全体を取り消すこととセマンティック上同じになります。最後に、システム権限が WITH NO ADMIN OPTION 句付きで付与されている場合、ADMIN OPTION FOR 句を REVOKE 文に含めても、元々管理システム権限が付与されていないので、何も取り消されません。

標準

(先頭に戻る) (512 ページ)

- SQL – その他の構文は、ISO/ANSI SQL 文法のベンダ拡張です。
- SAP Sybase Database 製品 - 構文は Adaptive Server でサポートされていません。

パーミッション

(先頭に戻る) (512 ページ)

取り消すシステム権限に対する管理権限が必要です。

参照:

- GRANT ROLE 文 (394 ページ)
- GRANT システム権限文 (402 ページ)

すべてのシステム権限のリスト

すべてのシステム権限のリスト。

システム権限は、ユーザが承認済みのデータベースタスクを実行する権限を制御します。

次に、使用可能なシステム権限のリストを示します。

- ACCESS SERVER LS
- ALTER ANY INDEX

- ALTER ANY MATERIALIZED VIEW
- ALTER ANY OBJECT
- ALTER ANY OBJECT OWNER
- ALTER ANY PROCEDURE
- ALTER ANY SEQUENCE
- ALTER ANY TABLE
- ALTER ANY TEXT CONFIGURATION
- ALTER ANY TRIGGER
- ALTER ANY VIEW
- ALTER DATABASE
- ALTER DATATYPE
- BACKUP DATABASE
- CHANGE PASSWORD
- CHECKPOINT
- COMMENT ANY OBJECT
- CREATE ANY INDEX
- CREATE ANY MATERIALIZED VIEW
- CREATE ANY OBJECT
- CREATE ANY PROCEDURE
- CREATE ANY SEQUENCE
- CREATE ANY TABLE
- CREATE ANY TEXT CONFIGURATION
- CREATE ANY TRIGGER
- CREATE ANY VIEW
- CREATE DATATYPE
- CREATE EXTERNAL REFERENCE
- CREATE MATERIALIZED VIEW
- CREATE MESSAGE
- CREATE PROCEDURE
- CREATE PROXY TABLE
- CREATE TABLE
- CREATE TEXT CONFIGURATION
- CREATE VIEW
- DEBUG ANY PROCEDURE
- DELETE ANY TABLE
- DROP ANY INDEX
- DROP ANY MATERIALIZED VIEW
- DROP ANY OBJECT
- DROP ANY PROCEDURE

SQL 文

- DROP ANY SEQUENCE
- DROP ANY TABLE
- DROP ANY TEXT CONFIGURATION
- DROP ANY VIEW
- DROP CONNECTION
- DROP DATATYPE
- DROP MESSAGE
- EXECUTE ANY PROCEDURE
- LOAD ANY TABLE
- INSERT ANY TABLE
- MANAGE ANY DBSPACE
- MANAGE ANY EVENT
- MANAGE ANY EXTERNAL ENVIRONMENT
- MANAGE ANY EXTERNAL OBJECT
- MANAGE ANY LDAP SERVER
- MANAGE ANY LOGIN POLICY
- MANAGE ANY MIRROR SERVER
- MANAGE ANY OBJECT PRIVILEGES
- MANAGE ANY SPATIAL OBJECT
- MANAGE ANY STATISTICS
- MANAGE ANY USER
- MANAGE ANY WEB SERVICE
- MANAGE AUDITING
- MANAGE MULTIPLEX
- MANAGE PROFILING
- MANAGE REPLICATION
- MANAGE ROLES
- MONITOR
- READ CLIENT FILE
- READ FILE
- REORGANIZE ANY OBJECT
- SELECT ANY TABLE
- SERVER OPERATOR
- SET ANY PUBLIC OPTION
- SET ANY SECURITY OPTION
- SET ANY SYSTEM OPTION
- SET ANY USER DEFINED OPTION
- SET USER (管理権限のみで付与)
- TRUNCATE ANY TABLE

- UPDATE ANY TABLE
- UPGRADE ROLE
- USE ANY SEQUENCE
- VALIDATE ANY OBJECT
- WRITE CLIENT FILE
- WRITE FILE

REVOKE USAGE ON SEQUENCE 文

指定されたシーケンスに対する USAGE 権限を削除します。

クイックリンク：

[「パラメータ」 \(517 ページ\)](#)

[「標準」 \(517 ページ\)](#)

[「パーミッション」 \(517 ページ\)](#)

構文

```
REVOKE USAGE ON SEQUENCE sequence-name  
FROM userID [, ...]
```

パラメータ

(先頭に戻る) (517 ページ)

- **userID** – ログインパスワードを持つ既存のユーザまたはロールの名前になります。複数の userID はカンマで区切ります。

標準

(先頭に戻る) (517 ページ)

- SQL – 構文は永続的ストアドモジュール機能です。
- SAP Sybase Database 製品 - セキュリティモデルは Adaptive Server と SAP Sybase IQ では異なるため、他の構文も異なります。

パーミッション

(先頭に戻る) (517 ページ)

次のいずれかが必要です。

- **MANAGE ANY OBJECT PRIVILEGE** システム権限。

SQL 文

- そのシーケンスを所有している。

ROLLBACK 文

最後の **COMMIT** または **ROLLBACK** の後に加えられた変更を取り消します。

クイックリンク：

「使用法」 (518 ページ)

「標準」 (518 ページ)

「許可」 (518 ページ)

構文

ROLLBACK [**WORK**]

使用法

(先頭に戻る) (518 ページ)

ROLLBACK は、作業 (トランザクション) の論理単位を終了し、このトランザクションの間にデータベースに加えられたすべての変更を取り消します。トランザクションは、1つのデータベース接続上で **COMMIT** 文または **ROLLBACK** 文の間で行われるデータベース作業です。

関連する動作：

- **WITH HOLD** 句を指定してオープンしなかったカーソルをすべてクローズします。
- **ROLLBACK** を実行して、トランザクションによって保持されているロックを開放します。

標準

(先頭に戻る) (518 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - Adaptive Server でサポートされています。

パーミッション

(先頭に戻る) (518 ページ)

なし。ただし、ユーザはデータベースに接続する必要があります。

参照：

- COMMIT 文 (129 ページ)
- ROLLBACK TO SAVEPOINT 文 (519 ページ)

ROLLBACK TO SAVEPOINT 文

セーブポイントの確定後以降に加えられた変更を取り消します。セーブポイントの前に加えられた変更は取り消されず、保留中の状態で残ります。

クイックリンク：

「パラメータ」 (519 ページ)

「標準」 (519 ページ)

「許可」 (519 ページ)

構文

```
ROLLBACK
      TO
      SAVEPOINT [ savepoint-name ]
```

パラメータ

(先頭に戻る) (519 ページ)

- **savepoint-name** – 現在のトランザクション内の **SAVEPOINT** 文で指定された識別子。 *savepoint-name* を省略すると、最新のセーブポイントが使用されます。指定したセーブポイントの後にあるセーブポイントは自動的に解放されます。

標準

(先頭に戻る) (519 ページ)

- SQL — ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - セーブポイントは Adaptive Server でサポートされていません。 Adaptive Server と互換性のある方法で同様の機能を実装するには、トランザクションのネストを使用することができます。

パーミッション

(先頭に戻る) (519 ページ)

なし。ただし、現在のトランザクション内に対応する **SAVEPOINT** が必要です。

参照：

- RELEASE SAVEPOINT 文 (483 ページ)
- ROLLBACK 文 (518 ページ)
- SAVEPOINT 文 (521 ページ)

ROLLBACK TRANSACTION 文 [T-SQL]

SAVE TRANSACTION を使用して確定されたセーブポイント以降に加えられた変更を取り消します。**SAVE TRANSACTION** の前に加えられた変更は取り消されず、保留中の状態で残ります。

クイックリンク：

「パラメータ」 (520 ページ)

「例」 (520 ページ)

「標準」 (521 ページ)

「許可」 (521 ページ)

構文

```
ROLLBACK TRANSACTION [ savepoint-name ]
```

パラメータ

(先頭に戻る) (520 ページ)

- **savepoint-name** – 現在のトランザクション内の **SAVE TRANSACTION** 文で指定された識別子。*savepoint-name* を省略した場合、未確定の変更はすべてロールバックされます。指定したセーブポイントの後にあるセーブポイントは自動的に解放されます。

例

(先頭に戻る) (520 ページ)

- **例 1** – 値 10、20 などを持つ 5 つのローが返されます。**ROLLBACK TRANSACTION** 文では、削除の作用は取り消されますが、それ以前の挿入や更新の作用は取り消されません。

```
BEGIN
  SELECT row_num INTO #tmp
  FROM sa_rowgenerator( 1, 5 )
  SAVE TRANSACTION before_delete
  UPDATE #tmp SET row_num=row_num*10
  DELETE FROM #tmp WHERE row_num >= 3
```

```
ROLLBACK TRANSACTION before_delete
SELECT * FROM #tmp
END
```

標準

(先頭に戻る) (520 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。

パーミッション

(先頭に戻る) (520 ページ)

なし。ただし、現在のトランザクション内に対応する **SAVE TRANSACTION** が必要です。

参照：

- BEGIN TRANSACTION 文 [T-SQL] (113 ページ)
- SAVE TRANSACTION 文 [T-SQL] (522 ページ)

SAVEPOINT 文

現在のデータベース内でセーブポイントを確立します。

クイックリンク：

「パラメータ」 (521 ページ)

「使用法」 (522 ページ)

「標準」 (522 ページ)

「許可」 (522 ページ)

構文

```
SAVEPOINT [ savepoint-name ]
```

パラメータ

(先頭に戻る) (521 ページ)

- **savepoint-name** – **RELEASE SAVEPOINT** 文または **ROLLBACK TO SAVEPOINT** 文で使用できる識別子です。

使用法

(先頭に戻る) (521 ページ)

トランザクションが終了すると、すべてのセーブポイントは自動的に解放されません。

トリガまたはアトミックな複合文の実行中に確立されたセーブポイントは、アトミック操作が終了すると自動的に解放されます。

標準

(先頭に戻る) (521 ページ)

- SQL — ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Adaptive Server ではサポートされていません。Adaptive Server と互換性のある方法で同様の機能を実装するには、トランザクションのネストを使用することができます。

パーミッション

(先頭に戻る) (521 ページ)

なし

参照：

- RELEASE SAVEPOINT 文 (483 ページ)
- ROLLBACK TO SAVEPOINT 文 (519 ページ)

SAVE TRANSACTION 文 [T-SQL]

現在のデータベース内でセーブポイントを確立します。

クイックリンク：

「パラメータ」 (523 ページ)

「例」 (523 ページ)

「標準」 (523 ページ)

「許可」 (523 ページ)

構文

```
SAVE TRANSACTION [ savepoint-name ]
```

パラメータ

(先頭に戻る) (522 ページ)

- **savepoint-name – ROLLBACK TRANSACTION** 文で使用できる識別子です。トランザクションが終了すると、すべてのセーブポイントは自動的に解放されます。

例

(先頭に戻る) (522 ページ)

- **例 1** – 値 10、20 などを持つ 5 つのローが返されます。**ROLLBACK TRANSACTION** 文では、削除の作用は取り消されますが、それ以前の挿入や更新の作用は取り消されません。

```
BEGIN
    SELECT row_num INTO #tmp
    FROM sa_rowgenerator( 1, 5 )
    UPDATE #tmp SET row_num=row_num*10
    SAVE TRANSACTION before_delete
    DELETE FROM #tmp WHERE row_num >= 3
    ROLLBACK TRANSACTION before_delete
    SELECT * FROM #tmp
END
```

標準

(先頭に戻る) (522 ページ)

- SQL — ISO/ANSI SQL 文法のベンダ拡張。

パーミッション

(先頭に戻る) (522 ページ)

なし

参照:

- BEGIN TRANSACTION 文 [T-SQL] (113 ページ)
- ROLLBACK TRANSACTION 文 [T-SQL] (520 ページ)

SELECT 文

データベースから情報を取り出します。

クイックリンク:

「パラメータ」 (525 ページ)

「例」 (531 ページ)

「使用法」 (532 ページ)

「標準」 (533 ページ)

「許可」 (534 ページ)

構文

```

SELECT [ ALL | DISTINCT ] [ row-limitation-option1 ] select-list
... [ INTO { host-variable-list | variable-list | table-name } ]
... [ INTO LOCAL TEMPORARY TABLE { table-name } ]
... [ FROM table-list ]
... [ WHERE search-condition ]
... [ GROUP BY [ expression [, ...]
                  | ROLLUP ( expression [, ...] )
                  | CUBE ( expression [, ...] ) ] ]
... [ HAVING search-condition ]
... [ ORDER BY { expression | integer } [ ASC | DESC ] [, ...] ]
| [ FOR JSON json-mode ]
... [ row-limitation-option ]

```

select-list - (構文に戻る)

```

{ column-name
| expression [ [ AS ] alias-name ]
| * }

```

row-limitation-option1 - (構文に戻る)

```

FIRST
| TOP { ALL | limit-expression } [ START AT startat-expression ]

```

limit-expression - (back to row-limitation-option1) or (back to row-limitation-option2)

simple-expression

startat-expression - (back to row-limitation-option1)

simple-expression

row-limitation-option2 - (構文に戻る)

```

LIMIT { [ offset-expression, ] limit-expression
| limit-expression OFFSET offset-expression }

```

offset-expression - (back to row-limitation-option2)

simple-expression

simple-expression - (back to startat-expression) or (back to offset-expression) or (back to limit-expression)

```

integer
| variable
| ( simple-expression )
| ( simple-expression { + | - | * } simple-expression )

```


パラメータ

(先頭に戻る) (523 ページ)

- **ALL または DISTINCT** – クエリ結果をフィルタします。どちらも指定しない場合、**SELECT** 文の句を満たすすべてのローが取得されます。**DISTINCT** を指定すると、重複した出力ローが削除されます。これは文の結果の射影と呼ばれます。多くの場合、**DISTINCT** を指定すると、文の実行時間が非常に長くなります。そのため、**DISTINCT** を使用するのには必要な場合だけにしてください。

DISTINCT を使用する場合、**DISTINCT** パラメータを持つ集合関数を文に含めることはできません。

- **row-limitation-option1** – クエリから返されるローの数を指定します。**FIRST** を指定すると、クエリから選択された最初のローが返されます。**TOP** を指定すると、指定した数のローがクエリから返されます。*number-of-rows* は 1 ~ 2147483647 の値です。整数の定数または整数の変数を指定することができます。

注意： 同じクエリ内で **TOP** と **LIMIT** を使用することはできません。

FIRST と **TOP** は、主に **ORDER BY** 句で使用されます。これらのキーワードを **ORDER BY** 句以外で使用した場合、同じクエリを実行しても、最適化が異なるクエリプランを選択する可能性があるため、そのたびに結果が変わる可能性があります。

FIRST と **TOP** は、クエリの最上位レベルの **SELECT** でのみ使用できます。したがって、派生テーブルやビュー定義で使用することはできません。**FIRST** や **TOP** をビュー定義で使用すると、クエリがビューで実行されたときに、このキーワードが無視される可能性があります。

FIRST の使用は、**ROW_COUNT** データベースオプションを 1 に設定するのと同じです。**TOP** の使用は、**ROW_COUNT** オプションをロー数と同じ数に設定するのと同じです。**TOP** と **ROW_COUNT** を両方とも設定した場合、**TOP** の値が優先されます。

グローバル変数、システム関数、またはプロキシテーブルが関連するクエリで **ROW_COUNT** オプションを使用した場合、一貫性のない結果が生成される可能性があります。詳細については、「**ROW_COUNT** オプション」を参照してください。

- **select-list** – データベースから取得する対象を指定する、カンマで区切られた式のリストです。アスタリスク (*) を指定すると、**FROM** 句に記述された全テーブルの全カラムが選択されます (*table-name* は、指定したテーブルのすべてのカラムを選択します)。*select-list* には、集合関数と統計関数を使用できます。

注意： SAP Sybase IQ では、SQL Anywhere や Adaptive Server と同様に、スカラサブクエリ (ネストされた select) を最上位レベルの SELECT の select リストで使用できます。サブクエリを条件値の式の内部 (**CASE** 文の内部など) で使用することはできません。

WHERE 句や HAVING 句の述部 (サポートされる述部のタイプのいずれか) でサブクエリを使用することもできます。ただし、WHERE 句または HAVING 句の内部であっても、値の式の内部や CONTAINS または LIKE 述部の内部でサブクエリを使用することはできません。外部ジョインの ON 句や、GROUP BY 句でサブクエリを使用することはできません。

- **alias-names** – クエリ全体で使用することができ、エイリアスの式を表します。Interactive SQL によって、エイリアス名も、**SELECT** 文から出力された各カラムの最上部に表示されます。オプションの *alias-name* を式の後で指定しないと、Interactive SQL は式を表示します。カラムのエイリアスにカラム名と同じ名前、または式を使用する場合、名前はテーブルカラム名でなく、エイリアスカラムとして処理されます。
- **INTO host-variable-list** – **SELECT** 文の結果が移動する場所を指定します。select-list 内の項目ごとに *host-variable* 項目が 1 つずつ必要です。select リスト項目は、順番にホスト変数に格納されます。インジケータホスト変数も各 *host-variable* とともに使用でき、プログラムは select リスト項目が NULL であったかどうかを通知することができます。Embedded SQL のみで使用されます。
- **INTO variable-list** – **SELECT** 文の結果が移動する場所を指定します。select リスト内の項目ごとに変数が 1 つずつ必要です。select リスト項目は、順番に変数に格納されます。プロシージャのみで使用されます。
- **INTO table-name** – テーブルを作成し、そのテーブルにデータを挿入します。

テーブル名が # で始まる場合、テーブルはテンポラリテーブルとして作成されます。それ以外の場合、テーブルは永久ベーステーブルとして作成されます。永久テーブルを作成するには、クエリが次のいずれかの条件を満たしている必要があります。

- *select-list* に複数の項目が含まれ、INTO ターゲットが 1 つの *table-name* 識別子である。
- *select-list* に * が含まれ、INTO ターゲットが *owner.table* として指定されている。

カラムが 1 つの永久テーブルを作成するには、テーブル名を *owner.table* として指定する必要があります。所有者の指定を省略すると、テンポラリテーブルが作成されます。

この文では、テーブル作成に関連する動作として、実行前に **COMMIT** が行われます。この文を実行するには、CREATE TABLE システム権限が必要です。新

しいテーブルに対するパーミッションは与えられません。この文は、**CREATE TABLE** とそれに続く **INSERT... SELECT** の省略形です。

ストアドプロシージャまたは関数から **SELECT INTO** を使用することはできません。これは、**SELECT INTO** がアトミックな文であり、アトミックな文内では **COMMIT** 文、**ROLLBACK** 文、および一部の **ROLLBACK TO SAVEPOINT** 文を実行できないためです。

この文を使用して作成されたテーブルには、プライマリキーが定義されていません。**ALTER TABLE** を使用してプライマリキーを追加できます。プライマリキーは、テーブルに更新または削除を適用する前に追加してください。そうしないと、これらの操作によって、影響を受けるローのトランザクションログにすべてのカラム値が記録されます。

この句は、SQL Anywhere の有効なクエリにしか使用できません。SAP Sybase IQ 拡張機能はサポートされません。

- **INTO LOCAL TEMPORARY TABLE** – ローカルテンポラリテーブルを作成し、そのテーブルにクエリの結果を格納します。この句を使用する場合、テンポラリテーブル名の先頭に # を付ける必要はありません。
- **FROM table-list** – *table-list* で指定されたローとビューを取得します。ジョイン演算子を使用して、ジョインを指定できます。詳細については、「FROM 句」を参照してください。FROM 句を指定しない **SELECT** 文を使用して、テーブルから抽出されなかった式の値を表示できます。次に例を示します。

```
SELECT @@version
```

グローバル変数 @@version の値を表示します。これは次と同じです。

```
SELECT @@version
FROM DUMMY
```

注意： FROM 句を省略した場合や、クエリ内のすべてのテーブルが SYSTEM DB 領域にある場合、クエリは SAP Sybase IQ ではなく SQL Anywhere によって処理されます。そのため、特に構文およびセマンティックの制限やオプションの設定の作用の違いによって、動作が変わる可能性があります。

FROM 句を必要としないクエリがある場合は、"FROM iq_dummy" 句を追加することによって、SAP Sybase IQ でのクエリの処理を強制することができます。iq_dummy は、ユーザが自分のデータベースに作成する 1 ロー、1 カラムのテーブルです。

- **WHERE search-condition** – FROM 句で指定したテーブルから選択するローを指定します。これを使用すると、複数のテーブル間のジョインも行えます。これを行うには、WHERE 句に条件を挿入します。この WHERE 句は、一方のテーブルのカラムまたはカラムグループを他方のテーブルのカラムまたはカラムグ

ループと関連付けます。両方のテーブルを FROM 句で指定する必要があります。

同じ **CASE** 文を、グループ化したクエリの SELECT 句と WHERE 句の両方で使用することはできません。

SAP Sybase IQ では、サブクエリ述部の分離もサポートされています。各サブクエリは、WHERE 句または HAVING 句内で他の述部とともに指定し、AND 演算子または OR 演算子を使用して結合できます。

- **GROUP BY** – カラム、エイリアス名、または関数をグループ化することができます。GROUP BY 式も select リストに入れる必要があります。クエリの結果には、指定したカラム、エイリアス、または関数の個々の値セットごとにローが 1 つずつ格納されます。テーブルリストのローの各グループに対する結果にはローが 1 つずつ存在するため、この結果ローは多くの場合グループと呼ばれます。GROUP BY 句があると、すべての NULL 値が同じものとして扱われます。これらのグループに集合関数を適用することで、意味のある結果を取得することができます。

GROUP BY には、定数を 2 つ以上含める必要があります。グループ化するクエリで定数を選択するために、GROUP BY 句に定数を追加する必要はありません。GROUP BY 式に定数が 1 つしか含まれていない場合は、エラーが返されてクエリが拒否されます。

GROUP BY を使用する場合、select リスト、HAVING 句、および ORDER BY 句で参照できるのは、GROUP BY 句で指定した識別子だけです。ただし、*select-list* と HAVING 句には、集合関数を含めることができます。

- **ROLLUP 演算子** – 詳細なレベルから総計までロールアップする GROUP BY 式の小計を計算します。

ROLLUP 演算子では、グループ化式が順に並べられたリストを引数として指定する必要があります。ROLLUP は、最初に GROUP BY で指定された標準の集合値を計算します。次に、グループ化を行うカラムのリストを右から左に移動し、より高いレベルの小計を連続して作成します。最後に総計が作成されます。グループ化するカラムの数が n 個の場合、ROLLUP は $n+1$ レベルの小計を作成します。

ROLLUP 演算子の制限：

- ROLLUP は、GROUP BY 句で使用可能なすべての集合関数をサポートしています。ただし、現時点では、COUNT DISTINCT および SUM DISTINCT はサポートしていません。
- ROLLUP は、**SELECT** 文のみで使用できます。SELECT サブクエリで ROLLUP を使用することはできません。

- ROLLUP、CUBE、および GROUP BY のカラムを同じ GROUP BY 句で組み合わせる複数グループ化の指定は、現時点ではサポートされていません。
- GROUP BY のキーとして定数式を指定することはできません。

ROLLUP 演算子を指定して GROUPING を使用すると、格納されていた NULL 値と ROLLUP によって作成されたクエリ結果の NULL 値が区別されます。

ROLLUP の構文：

```
SELECT ... [ GROUPING ( column-name ) ... ] ...
GROUP BY [ expression [, ... ]
| ROLLUP ( expression [, ... ] ) ]
```

GROUPING は、カラム名をパラメータとして受け取り、ブール値を返します。

表 9 : ROLLUP 演算子が指定された GROUPING によって返される値

結果の値	GROUPING の戻り値
ROLLUP 演算子によって作成された NULL	1 (真)
ローが小計であることを示す NULL	1 (真)
ROLLUP 演算子によって作成されていない NULL	0 (偽)
格納されていた NULL	0 (偽)

- **CUBE 演算子** – 複数の次元でデータをグループ化することによってデータを分析します。CUBE では、グループ化式 (次元) が順に並べられたリストを引数として指定する必要があります。これにより、**SELECT** 文は、組み合わせ可能なすべての次元のグループについて小計を計算することができます。CUBE 演算子は GROUP BY 句の一部です。

CUBE 演算子の制限：

- CUBE は、GROUP BY 句で使用可能なすべての集合関数をサポートしています。ただし、現時点では、COUNT DISTINCT および SUM DISTINCT はサポートしていません。
- CUBE は、逆分散統計関数である PERCENTILE_CONT と PERCENTILE_DISC を現時点ではサポートしていません。
- CUBE は、**SELECT** 文のみで使用できます。SELECT サブクエリで CUBE を使用することはできません。
- ROLLUP、CUBE、および GROUP BY のカラムを同じ GROUP BY 句で組み合わせる複数 GROUPING の指定は、現時点ではサポートされていません。
- GROUP BY のキーとして定数式を指定することはできません。

CUBE 演算子を指定して GROUPING を使用すると、格納されていた NULL 値と CUBE によって作成されたクエリ結果の NULL 値が区別されます。

CUBE の構文：

```
SELECT ... [ GROUPING ( column-name ) ... ] ...
GROUP BY [ expression [, ...]
| CUBE ( expression [, ...] ) ]
```

GROUPING は、カラム名をパラメータとして受け取り、ブール値を返します。

表 10 : CUBE 演算子が指定された GROUPING によって返される値

結果の値	GROUPING の戻り値
CUBE 演算子によって作成された NULL	1 (真)
ローが小計であることを示す NULL	1 (真)
CUBE 演算子によって作成されていない NULL	0 (偽)
格納されていた NULL	0 (偽)

クエリプランを生成するとき、SAP Sybase IQ オプティマイザは、GROUP BY CUBE ハッシュ操作で生成されるグループの合計数を見積もります。

MAX_CUBE_RESULTS データベースオプションでは、実行可能なハッシュアルゴリズムに対してオプティマイザが推測する、ローの予想数の上限を設定します。実際のロー数が **MAX_CUBE_RESULT** オプションの値を超える場合、オプティマイザはクエリの処理を中止し、「Estimate number: nnn exceed the DEFAULT_MAX_CUBE_RESULT of GROUP BY CUBE or ROLLUP」というエラーメッセージを返します。nnn は、オプティマイザが見積もった数字です。**MAX_CUBE_RESULT** オプションの設定については、「MAX_CUBE_RESULT オプション」を参照してください。

- **HAVING search-condition** – 個々のロー値ではなく、グループ値に基づきます。HAVING 句を使用できるのは、文に GROUP BY 句があるか、select リストが集合関数のみで構成されている場合だけです。HAVING 句の中で参照されるカラム名は、GROUP BY 句の中に入れるか、または HAVING 句の中の集合関数に対するパラメータとして使用する必要があります。
- **ORDER BY** – クエリの結果をソートします。ORDER BY リストの各項目には、昇順の場合は ASC、降順の場合は DESC のラベルを付けることができます。指定がない場合は、昇順であるとみなします。式が整数 n である場合、クエリ結果は select リストの n 番目の項目でソートされます。

Embedded SQL では、**SELECT** 文を使用してデータベースから結果を取得し、INTO 句でその値をホスト変数に格納します。**SELECT** 文から返されるローは 1 つのみである必要があります。複数のローを対象にクエリを実行する場合は、カーソルを使用する必要があります。

Java クラスを SELECT リストで指定することはできませんが、たとえば、Java クラスのラップとして機能する関数または変数を作成し、それを指定することはできます。

FOR JSON 句は、結果セットが JSON フォーマットで返されるように指定します。JSON フォーマットは、指定したモードによって異なります。この句は、FOR UPDATE 句または FOR READ ONLY 句と一緒に使用できません。FOR JSON で宣言したカーソルは暗黙的に READ ONLY になります。

RAW モードを指定した場合、結果セットの各ローは、フラットにされた JSON 表現として返されます。

AUTO モードでは、クエリのジョインに従って、クエリの結果をネストされた JSON オブジェクトとして返します。

EXPLICIT モードを指定すると、生成される JSON オブジェクトの形式を制御できます。EXPLICIT モードを使用すると、カラムおよびネストされた階層オブジェクトをより柔軟に指定して、同種または異種の配列を生成することができます。

- **row-limitation-option2** – WHERE 句を満たすローのサブセットを返します。row-limitation 句は、一度に 1 つのみ指定できます。この句を指定する場合は、ローの順序を意味のあるものにするために ORDER BY 句も指定する必要があります。row-limitation 句は、文の最上位のクエリブロックのみで有効です。

LIMIT 引数は、整数または整数の変数にする必要があります。OFFSET 引数は、0 以上の値が得られなければなりません。offset-expression の指定がない場合、デフォルトは 0 です。

row-limitation 句の LIMIT offset-expression, limit-expression は、LIMIT limit-expression OFFSET offset-expression と同じです。

LIMIT キーワードはデフォルトでは無効になっています。LIMIT キーワードを有効にするには、RESERVED_KEYWORDS オプションを使用します。

注意： 同じクエリ内で TOP と LIMIT を指定することはできません。

例

(先頭に戻る) (523 ページ)

- **例 1** – システムカタログのすべてのテーブルとビューをリストします。

```
SELECT tname
FROM SYS.SYSCATALOG
WHERE tname LIKE 'SYS%';
```

- **例 2** – すべての顧客とその顧客からの注文の総額をリストします。

SQL 文

```
SELECT CompanyName,  
       CAST( sum(SalesOrderItems.Quantity *  
               Products.UnitPrice) AS INTEGER) VALUE  
FROM Customers  
   LEFT OUTER JOIN SalesOrders  
   LEFT OUTER JOIN SalesOrderItems  
   LEFT OUTER JOIN Products  
GROUP BY CompanyName  
ORDER BY VALUE DESC
```

- **例 3** – 従業員の数をリストします。

```
SELECT count(*)  
FROM Employees;
```

- **例 4** – Embedded SQL の SELECT 文です。

```
SELECT count(*) INTO :size FROM Employees;
```

- **例 5** – 年、モデル、および色別に売り上げ合計をリストします。

```
SELECT year, model, color, sum(sales)  
FROM sales_tab  
GROUP BY ROLLUP (year, model, color);
```

- **例 6** – 値引きのあるすべての項目を抽出してテンポラリテーブルに格納します。

```
SELECT * INTO #TableTemp FROM lineitem  
WHERE l_discount < 0.5
```

- **例 7** – 従業員を姓でソートした場合に最初に表示される従業員に関する情報を返します。

```
SELECT FIRST *  
FROM Employees  
ORDER BY Surname;
```

- **例 8** – 名前を姓でソートした場合の最初の 5 人の従業員を返します。

```
SELECT TOP 5 *  
FROM Employees  
ORDER BY Surname;
```

```
SELECT *  
FROM Employees  
ORDER BY Surname  
LIMIT 5;
```

- **例 9** – 姓で降順にソートした場合の 5 番目と 6 番目の従業員をリストします。

```
SELECT *  
FROM Employees  
ORDER BY Surname DESC  
LIMIT 4,2;
```

使用法

(先頭に戻る) (523 ページ)

Interactive SQL で **SELECT** 文を使用して、データベース内のデータを参照したり、データベースから外部ファイルにデータをエクスポートしたりすることができます。

プロシージャまたは Embedded SQL で **SELECT** 文を使用することもできます。**SELECT** 文がローを 1 つだけ返す場合は、INTO 句を指定した **SELECT** 文を使用してデータベースから結果を取得します (SELECT INTO で作成されるテーブルは、IDENTITY/AUTOINCREMENT テーブルを継承しません)。複数のローを対象にクエリを実行する場合は、カーソルを使用する必要があります。複数のカラムを選択し、*#table* を使用しなければ、SELECT INTO によって永久的テーブルが作成されます。SELECT INTO *#table* を指定すると、カラムの数に関係なく、常にテンポラリテーブルが作成されます。カラムが 1 つしかないテーブルに対して SELECT INTO を実行すると、結果がホスト変数に格納されます。

注意： テンポラリテーブルに対して SELECT INTO を実行するスクリプトやストアードプロシージャを作成する場合は、ベースカラムではない select リスト項目を CAST 式にラップしてください。これにより、テンポラリテーブルのカラムのデータ型が目的のデータ型になることが保証されます。

名前が同じで所有者が異なるテーブルには、エイリアスを使用する必要があります。クエリでエイリアスを使用しないと、返される結果が不正確になります。

```
SELECT * FROM user1.t1
WHERE NOT EXISTS
  (SELECT *
   FROM user2.t1
   WHERE user2.t1.col1 = user1.t.col1);
```

正しい結果を得るには、次のように各テーブルにエイリアスを使用します。

```
SELECT * FROM user1.t1 U1
WHERE NOT EXISTS
  (SELECT *
   FROM user2.t1 U2
   WHERE U2.col1 = U1.col1);
```

variable-list がある INTO 句を使用できるのは、プロシージャ内のみです。

SELECT 文で、ベーステーブルまたはビューが許可されている場所にストアードプロシージャコールを記述することができます。ただし、CIS 機能補正のパフォーマンスに関する考慮事項が適用されます。たとえば、**SELECT** 文を使用して、プロシージャから結果セットを返すこともできます。

標準

(先頭に戻る) (523 ページ)

- SQL - ISO/ANSI SQL 準拠。

SQL 文

- SAP Sybase Database 製品 - SAP Sybase IQ でサポートされていますが、構文が多少異なります。

パーミッション

(先頭に戻る) (523 ページ)

指定したテーブルやビューに対する SELECT 権限が必要です。

参照：

- SELECT クエリブロック内のロー制限句 (5 ページ)
- CREATE VIEW 文 (292 ページ)
- DECLARE CURSOR 文 [ESQL] [SP] (299 ページ)
- FETCH 文 [ESQL] [SP] (357 ページ)
- FROM 句 (372 ページ)
- MAX_CUBE_RESULT オプション (712 ページ)
- OPEN 文 [ESQL] [SP] (461 ページ)
- UNION 演算 (564 ページ)
- RESERVED_KEYWORDS オプション (760 ページ)
- ROW_COUNT オプション (764 ページ)
- SUBQUERY_CACHING_PREFERENCE オプション (777 ページ)

SET 文 [ESQL]

SQL 変数に値を代入します。

クイックリンク：

「例」 (534 ページ)

「使用法」 (535 ページ)

「標準」 (536 ページ)

「パーミッション」 (536 ページ)

構文

```
SET identifier = expression
```

例

(先頭に戻る) (534 ページ)

- **例 1**— 次のコードフラグメントは、データベースに大きなテキスト値を挿入します。

```
EXEC SQL BEGIN DECLARE SECTION;
char buffer[5001];
EXEC SQL END DECLARE SECTION;

EXEC SQL CREATE VARIABLE hold_text VARCHAR;
EXEC SQL SET hold_text = '';
for(;;) {
    /* read some data into buffer ... */
    size = fread( buffer, 1, 5000, fp );
    if( size <= 0 ) break;

    /* buffer must be null-terminated */
    buffer[size] = '\0';
    /* add data to blob using concatenation */
    EXEC SQL SET hold_text = hold_text || :buffer;
}
EXEC SQL INSERT INTO some_table VALUES ( 1, hold_text );
EXEC SQL DROP VARIABLE hold_text;
```

- **例 2**— 次のコードフラグメントは、データベースに大きなバイナリ値を挿入します。

```
EXEC SQL BEGIN DECLARE SECTION;
DECL_BINARY( 5000 ) buffer;
EXEC SQL END DECLARE SECTION;
EXEC SQL CREATE VARIABLE hold_blob LONG BINARY;
EXEC SQL SET hold_blob = '';
for(;;) {
    /* read some data into buffer ... */
    size = fread( &(buffer.array), 1, 5000, fp );
    if( size <= 0 ) break;
    buffer.len = size;

    /* add data to blob using concatenation
       Note that concatenation works for
       binary data too! */
    EXEC SQL SET hold_blob = hold_blob || :buffer;
}
EXEC SQL INSERT INTO some_table VALUES ( 1, hold_blob );
EXEC SQL DROP VARIABLE hold_blob;
```

使用法

(先頭に戻る) (534 ページ)

SET 文は、**CREATE VARIABLE** 文を使用してそれまでに作成してある変数に新しい値を割り当てます。

SQL 文

変数は、カラム名を使用できる場所なら SQL 文のどこでも使用できます。識別子と一致するカラム名が存在しない場合、データベースサーバはその値と一致する変数がないかどうかをチェックし、その値を使います。

変数は、現在の接続にローカルなもので、データベースとの接続を切断したり、**DROP VARIABLE** を使用したりすると自動的に消えます。変数は **COMMIT** または **ROLLBACK** 文の影響を受けません。

変数は、Embedded SQL プログラムから **INSERT** 文または **UPDATE** 文の対象となるサイズの大きなテキストまたはバイナリオブジェクトを作成するために必要です。これは、Embedded SQL のホスト変数のサイズが 32,767 バイトに制限されているためです。

標準

(先頭に戻る) (534 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - サポートされていません。Adaptive Server では、変数は **SELECT** 文でテーブルを指定せずに割り当てます。これは SAP Sybase IQ でもサポートされている Transact-SQL 構文です。**SET** 文は、Adaptive Server でデータベースオプションを設定するのに使用されます。

パーミッション

(先頭に戻る) (534 ページ)

なし

参照：

- CREATE VARIABLE 文 (289 ページ)
- DROP VARIABLE 文 (347 ページ)

SET 文 [T-SQL]

Adaptive Server と互換性のある方法でデータベースオプションを設定します。

クイックリンク：

「パラメータ」 (537 ページ)

「使用法」 (538 ページ)

「標準」 (538 ページ)

「パーミッション」 (538 ページ)

構文

SET *option-name* **option-value**

option-value

```

ANSINULL [ ON | OFF ]
| ANSI_PERMISSIONS [ ON | OFF ]
| CLOSE_ON_ENDTRANS ON
| QUOTED_IDENTIFIER [ ON | OFF ]
| ROWCOUNT integer
| STRING_RTRUNCATION [ ON | OFF ]
| TRANSACTION ISOLATION LEVEL [ 0 | 1 | 2 | 3 ]

```

パラメータ

(先頭に戻る) (536 ページ)

- **ANSINULL** – SAP Sybase IQ と Adaptive Server では、値を NULL と比較するデフォルトの動作が異なります。ANSINULL を OFF に設定すると、Transact-SQL と互換性がある方法で NULL と比較されます。
- **ANSI_PERMISSIONS** – カラム参照を含む DELETE の実行に必要なパーミッションに関するデフォルトの動作は、SAP Sybase IQ と Adaptive Server で異なります。ANSI_PERMISSIONS を OFF に設定すると、DELETE のパーミッションに関して Transact-SQL との互換性が得られます。
- **CLOSE_ON_ENDTRANS** – ON (デフォルトであり、唯一の許容値) に設定すると、トランザクションの終了時にカーソルが閉じられます。オプションを ON に設定すると、CLOSE_ON_ENDTRANS は Transact-SQL と互換性のある動作になります。
- **QUOTED_IDENTIFIER** – 二重引用符内の文字列を識別子 (ON) とリテラル文字列 (OFF) のいずれとして解釈するかを制御します。
- **ROWCOUNT** – Transact-SQL で、カーソルでフェッチされるロー数の上限を整数で指定します。カーソルを再配置してフェッチされたローにも適用されます。フェッチがこの上限を超えると、警告が返されます。この設定は、**OPEN** 要求でカーソルに対するロー数の見積もりを返す際に評価されます。

注意： SAP Sybase IQ では `@@rowcount` グローバル変数がサポートされます。**SELECT** 文、**INSERT** 文、**DELETE** 文および **UPDATE** 文は、ROWCOUNT 句の値に影響します。ROWCOUNT 句はカーソル操作、**IF** 文、またはテーブルやプロシージャの作成や削除には影響しません。

SAP Sybase IQ では、ROWCOUNT が **dbisql** で表示可能なロー数より大きい場合、**dbisql** は、追加のフェッチを行ってカーソルを再配置できます。したがって、実際に表示されるロー数は、要求した数より少なくなることがあります。

また、トランケーションの警告が発生したためにローが再度フェッチされると、カウントが不正確になる場合があります。

値を 0 にすると、オプションをリセットし、すべてのローを取得します。

- **STRING_RTRUNCATION** – SQL 文字列データの割り当て時にスペース以外の文字をトランケートする際のデフォルトの動作は、SAP Sybase IQ と Adaptive Server で異なります。STRING_RTRUNCATION を ON に設定すると、16 進文字列 (バイナリデータ型) の比較を含めて Transact-SQL 互換の文字列比較が可能になります。
- **TRANSACTION ISOLATION LEVEL** – 現在の接続のロックの独立性レベルを設定します。Adaptive Server の有効オプションは 1 と 3 のみです。SAP Sybase IQ の有効オプションは 3 のみです。
- **SET PREFETCH** – SAP Sybase IQ では互換性保持のために使用が許可されますが、作用は特にありません。

使用法

(先頭に戻る) (536 ページ)

SAP Sybase IQ のデータベースオプションは、**SET OPTION** 文を使用して設定されます。ただし、SAP Sybase IQ は、互換性保持に特に有益な一連のオプションについては Adaptive Server の **SET** 文もサポートします。

標準

(先頭に戻る) (536 ページ)

- SQL – ISO/ANSI SQL 文法の Transact-SQL 拡張。
- SAP Sybase Database 製品 - SAP Sybase IQ は、Adaptive Server データベースオプションのサブセットをサポートしています。

パーミッション

(先頭に戻る) (536 ページ)

なし

参照：

- SET OPTION 文 (542 ページ)

SET CONNECTION 文 [ESQL] [Interactive SQL]

アクティブなデータベース接続を変更します。

クイックリンク：

「パラメータ」 (539 ページ)

「例」 (539 ページ)

「使用法」 (539 ページ)

「標準」 (540 ページ)

「パーミッション」 (540 ページ)

構文

```
SET CONNECTION [connection-name]
```

パラメータ

(先頭に戻る) (539 ページ)

- **connection-name** – 識別子、文字列、またはホスト変数

例

(先頭に戻る) (539 ページ)

- **例 1** – 次は、Embedded SQL の例です。

```
EXEC SQL SET CONNECTION :conn_name
```

- **例 2** – **dbisqli** から、現在の接続を接続名 "conn1" に設定します。

```
SET CONNECTION conn1
```

使用法

(先頭に戻る) (539 ページ)

現在の接続状態を保存し、再びアクティブな接続になるときにこれを再開します。*connection-name* を省略し、名前のない接続がある場合は、この接続がアクティブな接続になります。

注意：カーソルを Embedded SQL でオープンするとき、カーソルを現在の接続と関連付けます。接続が変更されると、カーソル名にはアクセスできません。カー

ソルはアクティブなまま配置され、関連付けられている接続が再びアクティブになると、アクセスできるようになります。

標準

(先頭に戻る) (539 ページ)

- SQL — **dbisql** の使用は ISO/ANSI SQL 文法のベンダ拡張です。Embedded SQL は上級レベル機能です。
- SAP Sybase Database 製品 - Open Client/Open Server でサポートされています。

パーミッション

(先頭に戻る) (539 ページ)

なし

参照：

- CONNECT 文 [ESQL] [Interactive SQL] (133 ページ)
- DISCONNECT 文 [Interactive SQL] (318 ページ)

SET DESCRIPTOR 文 [ESQL]

SQL 記述子領域の変数の記述や、記述子領域へのデータの格納を行います。

クイックリンク：

「パラメータ」 (541 ページ)

「例」 (541 ページ)

「標準」 (541 ページ)

「パーミッション」 (541 ページ)

構文

```
SET DESCRIPTOR descriptor-name
... { COUNT = { integer | hostvar }
    | VALUE n assignment [, ...] }
```

assignment

```
{ { TYPE
  | SCALE
  | PRECISION
  | LENGTH
  | INDICATOR } = { integer
```



```
| hostvar }
| DATA = hostvar }
```

パラメータ

(先頭に戻る) (540 ページ)

- **COUNT** – 記述子領域内にある記述済み変数の数を設定します。カウント対象の値は、記述子領域を割り付けたときに指定した変数の数を超えることはできません。
- **VALUE** – 値 n は、割り当ての対象となる記述子領域内の変数を指定します。
- **DATA** – DATA 句の使用時は、型チェックが実行され、記述子領域内の変数の型がホスト変数と同じ型であることが確認されます。エラーが発生すると、SQLCA にエラーコードが返されます。

例

(先頭に戻る) (540 ページ)

• 例 1

「ALLOCATE DESCRIPTOR 文 [ESQL]」を参照してください。

標準

(先頭に戻る) (540 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - Open Client/Open Server でサポートされています。

パーミッション

(先頭に戻る) (540 ページ)

なし

参照：

- ALLOCATE DESCRIPTOR 文 [ESQL] (7 ページ)
- DEALLOCATE DESCRIPTOR 文 [ESQL] (295 ページ)

SET OPTION 文

データベースの動作および Transact-SQL との互換性に影響を及ぼすオプションを変更します。オプションの値を設定すると、すべてのユーザまたは個別のユーザの動作を一時的または永続的なスコープで変更できます。

クイックリンク：

「パラメータ」 (542 ページ)

「例」 (543 ページ)

「使用法」 (543 ページ)

「標準」 (544 ページ)

「パーミッション」 (544 ページ)

構文

```
SET [ EXISTING ] [ TEMPORARY ] OPTION  
... [ userid. | PUBLIC. ] option-name = [ option-value ]
```

パラメータ

(先頭に戻る) (542 ページ)

- **option-value** – ホスト変数 (インジケータ使用可)、文字列、識別子、または数値。文字列に設定する場合、*option-value* の最大長は 127 バイトです。

option-value を省略すると、指定されたオプション設定がデータベースから削除されます。これがユーザ個人のオプション設定の場合は、値は PUBLIC 設定に戻ります。

注意： 整数値の指定が可能なデータベースオプションの場合、SAP Sybase IQ によって *option-value* の小数設定がすべて整数値にトランケートされます。たとえば、3.8 という値は 3 にトランケートされます。

- **EXISTING** – そのオプションに PUBLIC のユーザ ID 設定がすでに存在している場合を除き、個別のユーザ ID に対するオプション値は設定できません。
- **TEMPORARY** – 変更の有効期間を変更します。TEMPORARY 句がない場合、オプションの変更は永続的です。**SET OPTION** を使用して明示的に変更されるまで、変更されません。

個別のユーザ ID を使用して TEMPORARY 句が適用された場合、そのユーザがデータベースにログインしている間だけ、新しいオプション値が有効になります。

PUBLIC ユーザ ID とともに TEMPORARY 句が使用された場合、データベースの実行中はその変更が継続されます。データベースが停止されると、PUBLIC ユーザ ID の TEMPORARY オプションは永続値に戻ります。

TEMPORARY オプションが削除されると、オプション設定は永続的な設定に戻ります。

例

(先頭に戻る) (542 ページ)

- 例 1 – DATE_FORMAT オプションを設定します。

```
SET OPTION public.date_format = 'Mmm dd yyyy'
```

- 例 2 – WAIT_FOR_COMMIT オプションを ON に設定します。

```
SET OPTION wait_for_commit = 'on'
```

- 例 3 – embedded SQL の例を示します。

```
EXEC SQL SET OPTION :user.:option_name = :value;
EXEC SQL SET TEMPORARY OPTION Date_format = 'mm/dd/yyyy';
```

使用法

(先頭に戻る) (542 ページ)

オプションのクラスは次のとおりです。

- 一般的なデータベースオプション
- Transact-SQL 互換性データベースオプション

ユーザ ID または PUBLIC ユーザ ID を指定することで、個別ユーザ、*userid* で示されるロール、PUBLIC ユーザ ID (全ユーザがメンバーであるロール) のいずれに対してオプションが設定されるかが決まります。オプションがロール ID に適用される場合、ロールのメンバーによるオプション設定の継承は行われません。その変更はロール ID のみに適用されます。ロールが指定されない場合、そのオプション変更は現在ログオン中で **SET OPTION** 文を発行したユーザ ID に適用されます。たとえば、次の文では、オプション変更が PUBLIC ユーザ ID に適用されます。

```
SET OPTION Public.login_mode = standard
```

Embedded SQL では、データベースオプションが設定できるのは、一時的にすぎません。

PUBLIC ユーザ ID のオプション値を変更すると、独自の値を設定していないすべてのユーザにそのオプション値が設定されます。そのオプションに PUBLIC ユー

ザ ID 設定がすでに存在している場合を除き、個別のユーザ ID に対するオプション値は設定できません。

オプションの値を永続に設定するのではなく、PUBLIC ユーザ ID に対するオプションを一時的に設定すると、セキュリティが向上します。たとえば、**LOGIN_MODE** オプションが有効な場合、データベースは、そのデータベースを実行しているシステムのログインセキュリティに依存します。このオプションを一時的に有効にすると、Windows ドメインのセキュリティに依存しているデータベースは、データベースが停止し、ローカルマシンにコピーされた場合でも、セキュリティが損なわれることはありません。この場合、一時的に有効化された **LOGIN_MODE** は、永続値、つまり統合化ログインが許可されない Standard などに戻ります。

警告！ カーソルからローをフェッチしている際のオプション設定の変更は、予期しない動作を招く可能性があるためサポートされていません。たとえば、カーソルからのフェッチ中に **DATE_FORMAT** 設定を変更すると、結果セットで返されるローの日付フォーマットが統一されないことになります。ローをフェッチしている間にオプション設定を変更しないでください。

標準

(先頭に戻る) (542 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Adaptive Server ではサポートされていません。SAP Sybase IQ では、**SET** 文の使用時に一部の Adaptive Server オプションをサポートしています。

パーミッション

(先頭に戻る) (542 ページ)

各自のオプションを設定する際に特に要求されるシステム権限はありません。

別のユーザのデータベースオプションを設定するには、**SET ANY PUBLIC OPTION** システム権限が必要です。

PUBLIC ユーザ ID の **SYSTEM** オプションを設定するには、**SET ANY SYSTEM OPTION** システム権限が必要です。

PUBLIC ユーザ ID の **SECURITY** オプションを設定するには、**SET ANY SECURITY OPTION** システム権限が必要です。

参照：

- データベースオプション (585 ページ)

SET OPTION 文 [Interactive SQL]

Interactive SQL (**dbisql**) オプションを変更します。

クイックリンク：

「パラメータ」 (545 ページ)

「使用法」 (545 ページ)

「パーミッション」 (546 ページ)

構文

構文 1

```
SET [ TEMPORARY ] OPTION  
... [ userid. | PUBLIC.]option-name = [ option-value ]
```

構文 2

```
SET PERMANENT
```

構文 3

```
SET
```

パラメータ

(先頭に戻る) (545 ページ)

- **userid** – 識別子、文字列、またはホスト変数
- **option-name** – 識別子、文字列、またはホスト変数
- **option-value** – ホスト変数 (インジケータ使用可)、文字列、識別子、または数値

使用法

(先頭に戻る) (545 ページ)

SET PERMANENT (構文 2) は、現在の **dbisql** オプションをすべて **SYSOPTION** システムテーブルに格納します。**dbisql** が現在のユーザ ID で起動されるたびに、これらの値が自動的に設定されます。

構文 3 を使用すると、現在のオプション設定がすべて表示されます。**dbisql** またはデータベースサーバにテンポラリーオプションがある場合は、それが表示されます。それ以外の場合、永久オプション設定が表示されます。

オプション設定時にオプション名を誤って入力すると、誤った名前が **SYSOPTION** テーブルに保存されます。誤って入力した名前を **SYSOPTION** テーブルから削除

SQL 文

するには、オプション **PUBLIC** を使用します。この場合、オプション名の後に等号を付け、値は何も指定しません。

```
SET OPTION PUBLIC.a_mistyped_name=;
```

パーミッション

(先頭に戻る) (545 ページ)

各自のオプションを設定する際に特に要求されるシステム権限はありません。

別のユーザのデータベースオプションを設定するには、**SET ANY PUBLIC OPTION** システム権限が必要です。

PUBLIC ユーザ ID の **SYSTEM** オプションを設定するには、**SET ANY SYSTEM OPTION** システム権限が必要です。

PUBLIC ユーザ ID の **SECURITY** オプションを設定するには、**SET ANY SECURITY OPTION** システム権限が必要です。

参照：

- データベースオプション (585 ページ)

SET SQLCA 文 [ESQL]

デフォルトのグローバル *sqlca* 以外の SQLCA を使用するように、SQL プリプロセッサに通知します。

クイックリンク：

「パラメータ」 (546 ページ)

「例」 (547 ページ)

「使用法」 (547 ページ)

「標準」 (547 ページ)

「パーミッション」 (547 ページ)

構文

```
SET SQLCA sqlca
```

パラメータ

(先頭に戻る) (546 ページ)

- `sqlca` – 識別子または文字列

例

(先頭に戻る) (546 ページ)

- **例 1** – Windows DLL の次の関数を示します。DLL を使用するアプリケーションのそれぞれに独自の SQLCA があります。

```
an_sql_code FAR PASCAL ExecutesQL( an_application *app, char
*com )
{
    EXEC SQL BEGIN DECLARE SECTION;
    char *sqlcommand;
    EXEC SQL END DECLARE SECTION;
    EXEC SQL SET SQLCA "&app->.sqlca";
    sqlcommand = com;
    EXEC SQL WHENEVER SQLERROR CONTINUE;
    EXEC SQL EXECUTE IMMEDIATE :sqlcommand;
    return( SQLCODE );
}
```

使用法

(先頭に戻る) (546 ページ)

Embedded SQL 文ごとに、現在の SQLCA ポインタを暗黙のうちにデータベースインタフェースライブラリに渡します。C 言語ソースファイルにおいて、この文の後に記述されているすべての Embedded SQL 文は、新しい SQLCA を使います。この文を使用するのは、再入可能コードを記述するときだけです。`sqlca` は、ローカル変数を参照しなければなりません。グローバルまたはモジュール静的変数は別のスレッドによる修正を受けます。

標準

(先頭に戻る) (546 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Open Client/Open Server でサポートされていません。

パーミッション

(先頭に戻る) (546 ページ)

なし

SETUSER 文

ユーザは、タスクの開始の実行に必要な最小の権限をすでに持っている場合に、一時的に別のユーザのロールとシステム権限を使用して (同一化とも呼ばれる) 操作を実行できます。

注意： SET USER システム権限は 2 語で、SETUSER 文は 1 語です。

クイックリンク：

「パラメータ」 (548 ページ)

「使用法」 (548 ページ)

「標準」 (548 ページ)

「パーミッション」 (549 ページ)

構文

SETUSER *userID*

パラメータ

(先頭に戻る) (548 ページ)

- **UserID** – ログインパスワードを持つ既存のユーザまたはロールの名前になります。

使用法

(先頭に戻る) (548 ページ)

最小限の基準の検証は、SET USER システム権限が付与されるときではなく、SETUSER 文が実行されるときに行われます。

正常な同一化を終了するには、ユーザ ID を指定しないで SETUSER 文を発行します。

標準

(先頭に戻る) (548 ページ)

ANSI SQL – 準拠レベル： Transact-SQL 拡張。

パーミッション

(先頭に戻る) (548 ページ)

以下が必要です。

- 同一化実行者にターゲットユーザを同一化する権限が付与されている。
- 同一化実行者は、少なくともターゲットユーザに付与されているすべてのロールとシステム権限を持っている。
- 同一化実行者に、同等以上の管理権限付きで上述のロールとシステム権限が付与されている。

注意：管理権限の基準を満たすという目的においては、WITH ADMIN OPTION 句と WITH ADMIN ONLY OPTION 句は同様の管理権限を付与するものとみなされます。また、両者は、WITH NO ADMIN OPTION 句より上位の管理権限を付与するものとみなされます。たとえば、User1 には WITH ADMIN OPTION 句を指定して Role1 が付与され、User2 には WITH ADMIN ONLY 句を指定して Role1 が付与され、User3 には WITH NO ADMIN OPTION 句を指定して Role1 が付与されているとします。この場合、User1 と User2 には、同様の管理権限を持つ Role1 が付与されているとみなされます。また、User1 と User2 には、User3 より上位の管理権限を持つ Role1 が付与されているとみなされます。

- ターゲットユーザに拡張をサポートするシステム権限が付与されている場合、そのシステム権限を同一化実行者に付与する際に使用する句が、ターゲットユーザに付与する際に使用した句のスーパーセットである必要があります。拡張をサポートするのは SET USER システム権限と CHANGE PASSWORD システム権限のみです。
 - ANY 句は、*target_roles_list* 句と *target_users_list* 句のスーパーセットとみなされます。ターゲットユーザに ANY 句を使用して SET USER システム権限が付与されている場合、同一化実行者にも ANY 句を使用して付与されている必要があります。
 - ターゲットユーザに *target_roles_list* 句と *target_users_list* 句の両方を使用して SET USER システム権限が付与されている場合、同一化実行者にもその2つの句を使用してシステム権限が付与されている必要があり、さらにそれぞれの句のターゲットリストは、その句を使用してターゲットユーザに付与された内容と同等またはそのスーパーセットである必要があります。たとえば、同一化実行者とターゲットユーザのどちらも、ターゲットリストにそれぞれ User1 と User2、および Role1 と Role2 が含まれる場合、各句のターゲットリストで付与される内容は同等とみなされます。一方、同一化実行者に対してターゲットリストで付与される内容にそれぞれ User1 と User2 および Role1 と Role2 が含まれ、ターゲットユーザに対してターゲットリストで付与される内容に User1 と Role2 のみが含まれる場合、同

一化実行者に対してターゲットリストで付与される内容はターゲットユーザのスーパーセットであるとみなされます。

- ターゲットユーザに1つのターゲットリスト句を使用して SET USER システム権限が付与されている場合、同一化実行者のターゲットリストは、ターゲットユーザのリストと同等またはそのスーパーセットである必要があります。たとえば、同一化実行者とターゲットユーザの両方の *target_user_list* に User1 と User2 が含まれる場合 (同等) または同一化実行者のリストに User1 と User2、ターゲットユーザのリストに User2 がそれぞれ含まれる場合です。User1 と User2 (同一化実行者のリスト) は User2 (ターゲットユーザのリスト) のスーパーセットです。
- 定義により、ユーザは常に自分自身を同一化できます。したがって、ターゲットユーザに同一化実行者を同一化する権限が付与されている場合、これは同一化実行者の基準要件である同等またはスーパーセットであることに違反しません。たとえば、User3 が同一化実行者、User4 がターゲットユーザであり、User3 の *target_user_list* には、User4 と User5 が含まれています。User4 の *target_user_list* には、User3 と User5 が含まれています。このターゲットリストから同一化実行者を削除した場合、User3 のターゲットリストは基準要件を満たします。

SIGNAL 文

例外を発生させることができます。

クイックリンク：

「標準」 (550 ページ)

「パーミッション」 (550 ページ)

構文

SIGNAL *exception-name*

標準

(先頭に戻る) (550 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - **SIGNAL** は、Adaptive Server ではサポートされていません。

パーミッション

(先頭に戻る) (550 ページ)

なし

参照：

- BEGIN ... END 文 (108 ページ)
- RESIGNAL 文 (486 ページ)

START DATABASE 文 [Interactive SQL]

指定したデータベースサーバ上でデータベースを起動します。

クイックリンク：

「パラメータ」 (551 ページ)

「例」 (552 ページ)

「使用法」 (552 ページ)

「標準」 (552 ページ)

「パーミッション」 (552 ページ)

構文

```
START DATABASE database-file
... [ AS database-name ]
... [ ON engine-name ]
... [ AUTOSTOP { YES | NO } ]
... [ KEY key ]
```

パラメータ

(先頭に戻る) (551 ページ)

- **database-name** – 指定しないと、デフォルト名がデータベースに割り当てられます。このデフォルト名は、データベースファイルのルートです。たとえば、ファイル `c:\sybase\16_0\demo\iqdemo.db` のデータベースにはデフォルト名 `iqdemo` が付けられます。
- **engine-name** – 指定しないと、フォルトのデータベースサーバが使用されます。デフォルトのデータベースサーバは、現在作動中のサーバの中で最初に起動したサーバです。
- **AUTOSTOP** – YES (デフォルト) に設定すると、最後の接続が切断されたときに、データベースが自動的にアンロードされます。NO に設定すると、データベースはアンロードされません。

SQL 文

- **KEY** – 強い暗号で保護されたデータベースの **KEY** 値 (パスワード) を入力します。

注意： 指定した SAP Sybase IQ データベースサーバで 1 つのデータベースのみを起動します。

例

(先頭に戻る) (551 ページ)

- **例 1** – UNIX システムで、現在のサーバからデータベースファイル /s1/sybase/sample_2.db を起動します。

```
START DATABASE '/s1/sybase/sample_2.db'
```

- **例 2** – Windows システムで、サーバ eng1 にあるデータベースファイル c:\¥sybase¥sample_2.db を sam2 として起動します。

```
START DATABASE 'c:\¥sybase¥sample_2.db'  
AS sam2  
ON eng1
```

使用法

(先頭に戻る) (551 ページ)

データベースサーバが作動中である必要があります。ファイルが現在のディレクトリにない場合は、データベースファイルに対してフルパスを指定してください。

START DATABASE 文は、指定したデータベースに **dbisql** を接続しません。接続するには、**CONNECT** 文を発行する必要があります。

標準

(先頭に戻る) (551 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - なし。

パーミッション

(先頭に戻る) (551 ページ)

SERVER OPERATOR システム権限が必要です。

START ENGINE 文 [Interactive SQL]

データベースサーバを起動します。

クイックリンク：

「パラメータ」 (553 ページ)

「例」 (553 ページ)

「使用法」 (553 ページ)

「標準」 (554 ページ)

「パーミッション」 (554 ページ)

構文

```
START ENGINE AS engine-name [ STARTLINE command-string ]
```

パラメータ

(先頭に戻る) (553 ページ)

- **STARTLINE** – サーバのオプションセットを指定します。
- **command-string** – 有効なコマンド文字列は、データベースサーバのコマンドラインの説明に従った文字列です。『ユーティリティガイド』の「start_iq データベースサーバ起動ユーティリティ」を参照してください。

例

(先頭に戻る) (553 ページ)

- **例 1** – eng1 という名前のデータベースサーバを起動します。ただし、このサーバ上のデータベースはいずれも起動しません。

```
START ENGINE AS eng1
```

- **例 2** – キャッシュを 8096 KB として同じサーバを起動します。

```
START ENGINE AS eng1 STARTLINE 'start_iq -c 8096'
```

使用法

(先頭に戻る) (553 ページ)

SAP Sybase IQ を適切に動作させるには、サーバオプションをいくつか設定する必要があります。適切なオプションセットを使用するために、SAP Control Center を

SQL 文

使用してサーバを起動するか、または **start_iq** コマンドで設定ファイルを使用してサーバを起動してください。

標準

(先頭に戻る) (553 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - なし。

パーミッション

(先頭に戻る) (553 ページ)

なし

参照:

- STOP ENGINE 文 [Interactive SQL] (557 ページ)

START EXTERNAL ENVIRONMENT 文

外部環境を開始します。

構文

```
START EXTERNAL ENVIRONMENT environment-name
```

```
environment-name :  
JAVA  
| PERL  
| PHP  
| CLR  
| C_ESQL32  
| C_ESQL64  
| C_ODBC32  
| C_ODBC64
```

パラメータ

environment-name – 開始する外部環境の名前。

備考

権限
なし

関連する動作
なし

標準と互換性

- **SQL/2008** – ベンダー拡張。

例

Perl 外部環境を開始します。

```
START EXTERNAL ENVIRONMENT PERL;
```

START JAVA 文

Java VM を適宜ロードし、ユーザが Java の機能を使い始めるときに、Java VM ロード中の一時停止が発生しないようにします。

クイックリンク：

「例」 (555 ページ)

「標準」 (555 ページ)

「パーミッション」 (555 ページ)

構文

```
START EXTERNAL ENVIRONMENT JAVA
```

例

(先頭に戻る) (555 ページ)

- **例 1** – Java VM を起動します。

```
START EXTERNAL ENVIRONMENT JAVA
```

標準

(先頭に戻る) (555 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - なし。

パーミッション

(先頭に戻る) (555 ページ)

なし

参照：

- STOP JAVA 文 (559 ページ)

STOP DATABASE 文 [Interactive SQL]

指定したデータベースサーバ上のデータベースを停止します。

クイックリンク：

「パラメータ」 (556 ページ)

「例」 (556 ページ)

「標準」 (557 ページ)

「パーミッション」 (557 ページ)

構文

```
STOP DATABASE database-name  
... [ ON engine-name ]  
... [ UNCONDITIONALLY ]
```

パラメータ

(先頭に戻る) (556 ページ)

- **database-name** – データベース起動時の `-n` パラメータ、または DBN (DatabaseName) 接続パラメータで指定した名前。通常、この名前は、カタログストアを格納するデータベースファイルのファイル名から `.db` 拡張子を除いたものですが、ユーザ定義の名前も使用できます。
- **engine-name** – 指定しないと、作動中のすべてのエンジンで指定した名前のデータベースが検索されます。
- **UNCONDITIONALLY** – 指定すると、データベースはそのデータベースへの接続がある場合でも停止します。指定しないと、接続がある場合には、データベースは停止しません。

例

(先頭に戻る) (556 ページ)

- **例 1** – デフォルトサーバ上のデータベース `sample` を停止します。


```
STOP DATABASE sample
```

標準

(先頭に戻る) (556 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - なし。

パーミッション

(先頭に戻る) (556 ページ)

SERVER OPERATOR システム権限が必要です。

参照：

- DISCONNECT 文 [Interactive SQL] (318 ページ)
- START DATABASE 文 [Interactive SQL] (551 ページ)

STOP ENGINE 文 [Interactive SQL]

データベースサーバを停止します。

クイックリンク：

「パラメータ」 (557 ページ)

「例」 (558 ページ)

「標準」 (558 ページ)

「パーミッション」 (558 ページ)

構文

```
STOP ENGINE engine-name [ UNCONDITIONALLY ]
```

パラメータ

(先頭に戻る) (557 ページ)

- **UNCONDITIONALLY** - 指定すると、データベースサーバはそのサーバへの接続がある場合でも停止します。指定しないと、接続がある場合には、データベースサーバは停止しません。

SQL 文

例

(先頭に戻る) (557 ページ)

- **例 1** – sample という名前のデータベースを停止します。

```
STOP ENGINE sample
```

標準

(先頭に戻る) (557 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - なし。

パーミッション

(先頭に戻る) (557 ページ)

なし

参照:

- START ENGINE 文 [Interactive SQL] (553 ページ)

STOP EXTERNAL ENVIRONMENT 文

外部環境を停止します。

構文

```
STOP EXTERNAL ENVIRONMENT environment-name
```

```
environment-name :
```

```
JAVA  
| PERL  
| PHP  
| CLR  
| C_ESQL32  
| C_ESQL64  
| C_ODBC32  
| C_ODBC64
```

パラメータ

environment-name – 停止する外部環境の名前。

備考

権限
なし

関連する動作
なし

標準と互換性

- **SQL/2008** – ベンダー拡張。

例

この例では、Perl 外部環境を停止します。

```
STOP EXTERNAL ENVIRONMENT PERL;
```

STOP JAVA 文

Java VM に関連するリソースを解放して、システムリソースを経済的に使用します。

クイックリンク：

「標準」 (559 ページ)

「パーミッション」 (559 ページ)

構文

```
STOP EXTERNAL ENVIRONMENT JAVA
```

標準

(先頭に戻る) (559 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - なし。

パーミッション

(先頭に戻る) (559 ページ)

なし

参照：

- START JAVA 文 (555 ページ)

TRIGGER EVENT 文

指定されたイベントをトリガします。イベントには、イベントトリガ用に定義されたイベントと、スケジュールされたイベントがあります。

クイックリンク：

「使用法」 (560 ページ)

「パーミッション」 (560 ページ)

構文

```
TRIGGER EVENT event-name [ ( parm = value, ... ) ]
```

使用法

(先頭に戻る) (560 ページ)

トリガ条件またはスケジュールには、**CREATE EVENT** 文でアクションを関連付けます。**TRIGGER EVENT** を使用すると、スケジュールされた時間でなくても、トリガ条件が発生しなくても、イベントハンドラを強制的に実行できます。**TRIGGER EVENT** は、無効なイベントハンドラを実行しません。

トリガ条件によってイベントハンドラが実行されるときに、データベースサーバは `event_parameter` 関数を使用してコンテキスト情報をイベントハンドラに渡すことができます。**TRIGGER EVENT** を使用すると、これらのパラメータを明示的に指定して、イベントハンドラのコンテキストをシミュレートできます。

イベントをトリガするときは、イベント名を指定します。イベント名は、システムテーブル `SYSEVENT` をクエリすることでリストできます。次に例を示します。

```
SELECT event_id, event_name FROM SYS.SYSEVENT
```

パーミッション

(先頭に戻る) (560 ページ)

MANAGE ANY DBSPACE システム権限が必要です。

参照：

- ALTER EVENT 文 (20 ページ)
- CREATE EVENT 文 (157 ページ)

TRUNCATE 文

テーブル定義を消去することなく、すべてのローをテーブルまたはマテリアライズドビューから消去します。

クイックリンク：

「パラメータ」 (561 ページ)

「例」 (561 ページ)

「使用法」 (562 ページ)

「標準」 (562 ページ)

「パーミッション」 (562 ページ)

構文

構文 1

```
TRUNCATE
  TABLE [ owner.] table-name
  | MATERIALIZED VIEW owner.] materialized-view-name
```

構文 2

```
TRUNCATE TABLE [ owner .] table
  [ PARTITION partition-name
  | SUBPARTITION subpartition-name ]
```

パラメータ

(先頭に戻る) (561 ページ)

- **PARTITION** – トランケートするパーティションを指定します。他のパーティションのデータには影響しません。

注意： PARTITION 句で RLV 対応テーブルを指定するとエラーが発生します。

- **SUBPARTITION** – 複合分割スキームによって分割されるテーブルをトランケートします。

注意： SUBPARTITION 句で RLV 対応テーブルを指定するとエラーが発生します。

例

(先頭に戻る) (561 ページ)

- **例 1** – Sale テーブルから、すべてのローを削除します。

```
TRUNCATE TABLE Sale
```

使用法

(先頭に戻る) (561 ページ)

トランザクションログに個別のローの削除が記録されない点を除き、**TRUNCATE** は、WHERE 句なしの **DELETE** 文と同じです。**TRUNCATE TABLE** 文の後は、**DROP TABLE** 文を発行するまで、引き続きテーブル構造とすべてのインデックスが存在します。カラム定義と制約はそのまま残り、パーミッションは有効なままです。

TRUNCATE 文は、データ定義文と同様に単一の文としてトランザクションログに入力されます。それぞれの削除されたローは、トランザクションログには記録されません。

標準

(先頭に戻る) (561 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Adaptive Server でサポートされています。

パーミッション

(先頭に戻る) (561 ページ)

次のいずれかが必要です。

- **TRUNCATE ANY TABLE** システム権限
- **ALTER ANY TABLE** システム権限
- **ALTER ANY OBJECT** システム権限
- テーブルに対する **TRUNCATE** 権限
- オブジェクトを所有している

他のユーザがテーブルに対する読み込みアクセスを保持していても、テンポラリテーブルとベーステーブルの両方に対して **TRUNCATE TABLE** を実行することができます。この動作は、ベーステーブルのトランケートに排他的アクセスを必要とする SQL Anywhere とは異なります。他のユーザに読み込みアクセスがある間も **TRUNCATE TABLE** を実行できるのは、SAP Sybase IQ にテーブルのバージョン管理機能があるためです。ただし、これらのユーザに表示されるテーブルのバージョンは、読み書きのトランザクションのコミット時点によって異なります。

参照：

- **DELETE** 文 (309 ページ)

TRUNCATE TEXT INDEX 文

MANUAL または AUTO REFRESH テキストインデックスのデータを削除します。

クイックリンク：

「パラメータ」 (563 ページ)

「例」 (563 ページ)

「使用法」 (563 ページ)

「標準」 (564 ページ)

「パーミッション」 (564 ページ)

構文

```
TRUNCATE TEXT INDEX text-index-name  
ON [ owner. ] table-name
```

パラメータ

(先頭に戻る) (563 ページ)

- **ON** - テキストインデックスを構築するテーブルの名前。

例

(先頭に戻る) (563 ページ)

- **例 1** - 最初の文は、`txt_index_manual` テキストインデックスを作成します。2 番目の文では、テキストインデックスにデータを格納します。3 番目の文では、テキストインデックスデータをトランケートします。

```
CREATE TEXT INDEX txt_index_manual ON GROUPO.MarketingInformation  
( Description )  
    MANUAL REFRESH;  
REFRESH TEXT INDEX txt_index_manual ON  
GROUPO.MarketingInformation;  
TRUNCATE TEXT INDEX txt_index_manual ON  
GROUPO.MarketingInformation;
```

トランケートされたテキストインデックスは、次の更新時にデータが再格納されます。

使用法

(先頭に戻る) (563 ページ)

TRUNCATE TEXT INDEX 文は、テキストインデックス定義を削除しないで、手動テキストインデックスからデータを削除する場合に使用します。たとえば、テキストインデックスのテキスト設定オブジェクトを変更してストップリストを更新する場合、テキストインデックスをトランケートし、そのインデックスが参照しているテキスト設定オブジェクトを変更します。次にテキストインデックスを更新して、新しいデータをインデックスに格納します。

IMMEDIATE REFRESH (デフォルト) と定義されているテキストインデックスに対して TRUNCATE TEXT INDEX 文を実行することはできません。IMMEDIATE REFRESH テキストインデックスの場合は、インデックスを削除する必要があります。

TRUNCATE TEXT INDEX ではテーブルへの排他アクセスが必要です。トランケートされるテーブルを参照するカーソルが開いている場合は、それらのカーソルをすべて閉じ、COMMIT または ROLLBACK 文を実行してテーブルへの参照を解除します。

標準

(先頭に戻る) (563 ページ)

ANSI SQL – 準拠レベル： Transact-SQL 拡張。

パーミッション

(先頭に戻る) (563 ページ)

次のいずれかが必要です。

- ALTER ANY INDEX システム権限
- ALTER ANY OBJECT システム権限
- そのテーブルに対する REFERENCES 権限
- テーブルを所有していること

UNION 演算

2つ以上の SELECT 文の結果を結合します。

クイックリンク：

「パラメータ」 (565 ページ)

「例」 (565 ページ)

「使用法」 (565 ページ)

「標準」 (566 ページ)

「パーミッション」 (566 ページ)

構文

```
select-without-order-by
... UNION [ ALL ] select-without-order-by
... [ UNION [ ALL ] select-without-order-by ]...
... [ ORDER BY integer [ ASC | DESC ] [, ...] ]
```

パラメータ

(先頭に戻る) (564 ページ)

- **ALL** – UNION ALL の結果は、単にコンポーネントの **SELECT** 文の結果を結合したものです。UNION の結果は UNION ALL と同じですが、重複ローが削除されている点が異なります。重複ローを削除するには余分な処理が必要なため、可能であれば UNION の代わりに UNION ALL を使用してください。
- **ORDER BY** – 順序のリストでは整数のみを使用できます。これらの整数はソートされるカラムの位置を指定します。

例

(先頭に戻る) (564 ページ)

- **例 1** – 従業員と顧客のそれぞれの姓のすべてをリストします。

```
SELECT Surname
FROM Employees
UNION
SELECT Surname
FROM Customers
```

使用法

(先頭に戻る) (564 ページ)

複数の **SELECT** 文の結果は、UNION 句を使用して 1 つの大きな結果へと結合できます。コンポーネントの **SELECT** 文それぞれの select リストに同じ数の項目を指定します。各文に **ORDER BY** 句を含めることはできません。「FROM 句」を参照してください。

2 つの select リスト内の対応する項目のデータ型が異なる場合、SAP Sybase IQ は結果の中から対応するカラムのデータ型を選択し、各コンポーネント **SELECT** 文のカラムを必要に応じて自動的に変換します。

表示されるカラム名は、最初の **SELECT** 文に対して表示されるカラム名と同じです。

注意： **SELECT** 文に定数値と UNION ALL ビューが含まれているが、FROM 句が省略されている場合は、エラーを防ぐために `iq_dummy` を使用します。詳細については、「FROM 句」を参照してください。

標準

(先頭に戻る) (564 ページ)

- SQL - ISO/ANSI SQL 準拠。
- SAP Sybase Database 製品 - SAP Adaptive Server® Enterprise でサポートされています。この製品では、**COMPUTE** 句もサポートされています。

パーミッション

(先頭に戻る) (564 ページ)

SELECT 文の各コンポーネントに対する **SELECT** 権限が必要です。

参照：

- FROM 句 (372 ページ)
- SELECT 文 (523 ページ)

UPDATE 文

単一のテーブル、または単一のテーブルしか含まないビューの既存のローを修正します。

クイックリンク：

「パラメータ」へ移動 (567 ページ)

「例」 (568 ページ)

「使用法」 (568 ページ)

「標準」 (571 ページ)

「パーミッション」 (571 ページ)

構文

UPDATE *table-name*

```
... SET [column-name = expression, ...  
[ FROM table-expression ]  
... [ WHERE search-condition ]  
... [ ORDER BY expression [ ASC | DESC ] , ...]
```

```
table-name - (back to Syntax)
    [ owner. ] table-name [ [ AS ] correlation-name ]
    | [ owner. ] view-name [ [ AS ] correlation-name ]
```

```
table-expression - (back to Syntax)
    table-spec
    | table-expression join-type table-spec [ ON condition ]
    | table-expression, ...
```

パラメータ

(先頭に戻る) (566 ページ)

- **FROM 句** – ジョインに基づいてテーブルを更新できます。**FROM 句**が存在する場合、**table-name** では更新される単一のテーブルを指定します。この名前は **FROM 句**と同じように修飾されている必要があります。**FROM 句**で相関名が使用されている場合、同一の相関名を **table-name** として指定します。

次の文は、**FROM 句**を使用した **UPDATE 文**に相関名を使用したテーブル式が含まれており、この文のテーブル名に潜在的なあいまいさが存在することを示しています。

```
UPDATE table_1
SET column_1 = ...
FROM table_1 AS alias_1, table_2 AS alias_2
WHERE ...
```

FROM 句の `table_1` の各インスタンスには相関名があり、`table_1` のそれ自体へのセルフジョインを示しています。ただし、この **UPDATE 文**はセルフジョインを構成するローのうち、どのローを更新対象とするかの指定に失敗します。これは、次のように、**UPDATE 文**の相関名を指定することで修正できます。

```
UPDATE table_1
SET column_1 = ...
FROM table_1 AS alias_1, table_1 AS alias_2
WHERE ...
```

ローを更新しようとしているテーブルと同じテーブル名が **FROM 句**で使用されている場合、以下のいずれかに該当すると、同じテーブルを参照しているとみなされます。

- どちらのテーブル参照も、ユーザ ID を指定して修飾されていない
- 両方のテーブル参照が、ユーザ ID を指定して修飾されている
- 両方のテーブル参照が、相関名で指定されている

それらのテーブル参照が同じかどうかをサーバで判別できない場合、SQL エラーが表示されます。これにより、意図しないローの更新による意図しないセマンティックを防ぎます。

- **WHERE** 句 – 指定した場合、検索条件を満たすローのみが更新されます。WHERE 句を指定しない場合、すべてのローが更新されます。

例

(先頭に戻る) (566 ページ)

- **例 1** – 従業員 Philip Chin (従業員 129) を販売部からマーケティング部に異動します。

```
UPDATE Employees
SET DepartmentID = 400
WHERE EmployeeID = 129;
```

- **例 2** – マーケティング部 (400) はボーナスを各従業員の基本給の 4% から 6% に増加します。

```
UPDATE Employees
SET bonus = base * 6/100
WHERE DepartmentID = 400;
```

- **例 3** – 従業員全員の給与を、各部のボーナス分だけ昇給します。

```
UPDATE Employees
SET emp.Salary = emp.Salary + dept.bonus
FROM Employees emp, Departments dept
WHERE emp.DepartmentID = dept.DepartmentID;
```

- **例 4** – 別の方法を使用して、各従業員に部門のボーナス分の昇給を行います。

```
UPDATE Employees
SET emp.salary = emp.salary + dept.bonus
FROM Employees emp JOIN Departments dept
ON emp.DepartmentID = dept.DepartmentID;
```

使用法

(先頭に戻る) (566 ページ)

UPDATE 文で参照されるテーブルは、ベーステーブルまたはテンポラリテーブルです。

更新に関するデフォルトは、現在のユーザ、ユーザと現在のタイムスタンプ、およびタイムスタンプのみに適用されます。

それぞれ指定したカラムを、等号の右側の式の値に設定します。column-name も式の中で使用できます。この場合、古い値を使用します。

FROM 句にジョイン条件を使って複数のテーブルを指定し、指定されたすべてのテーブルのすべてのカラムを、ジョイン条件および **WHERE** 条件でフィルタして返すことができます。

FROM 句で間違ったジョイン条件を使用すると、予測できない結果が返されます。

FROM 句に 1 対多のジョインを指定し、SET 句でジョインの「多」の側のセルを参

照した場合、セルは最初に選択された値で更新されます。言い換えれば、ジョイン条件によって、テーブル内でロー ID ごとに複数のローが更新される場合、更新後の結果は最初に返されたローの値になります。次に例を示します。

```
UPDATE T1
SET T1.c2 = T2.c2
FROM T1 JOIN TO T2
ON T1.c1 = T2.c1
```

テーブル T2 に T2.c1 に対応するローが複数存在する場合、結果は次のようになります。

T2.c1	T2.c2	T2.c3
1	4	3
1	8	1
1	6	4
1	5	2

ORDER BY 句を指定しなければ、T1.c2 は 4、6、8、9 のいずれかになります。

- **ORDER BY T2.c3** を指定すると、T1.c2 は 8 に更新されます。
- **ORDER BY T2.c3 DESC** を指定すると、T1.c2 は 6 に更新されます。

SAP Sybase IQ は、更新対象のテーブルが外部ジョインの NULL 入力側にある場合は、**UPDATE** 文を拒否します。具体的には、次のようになります。

- 左外部ジョインでは、ジョインの左側のテーブルでジョイン対象カラムのローがすべて揃っている必要があります。
- 右外部ジョインでは、ジョインの右側のテーブルでジョイン対象カラムのローがすべて揃っている必要があります。
- フル外部ジョインでは、どのテーブルもジョイン対象のカラムのローがすべて揃っている必要があります。

たとえば、次の文では、テーブル T1 は左外部ジョインの左側にあるため、すべてのローが揃っていなければなりません。

```
UPDATE T1
SET T1.c2 = T2.c4
FROM T1 LEFT OUTER JOIN T2
ON T1.rowid = T2.rowid
```

通常、ローを更新する順序は重要ではありません。ただし、**NUMBER(*)** 関数と一緒に使って、ある指定された順序でローの中の数字を増加させる場合に、順序付けが役に立ちます。**NUMBER(*)** 関数を使用していない場合、**ORDER BY** 句は使用しないでください。**UPDATE** 文では **ORDER BY** 句がないほうがパフォーマンスが良いからです。

UPDATE 文で **NUMBER(*)** 関数を **SET** 句で使用し、**FROM** 句に 1 対多のジョインが指定されている場合、**NUMBER(*)** は増加する一意の数値を生成しますが、ローの削除が発生するため、数値の増加は連続的にはなりません。

ORDER BY 句を使用すると、**FROM** 句で複数のテーブルがジョインされている場合に **UPDATE** が返す結果を制御できます。

SAP Sybase IQ では検索された **UPDATE** 内の **ORDER BY** 句が無視され、構文が有効な ANSI 構文ではないことを示すメッセージが返されます。

SET 句の左側には、ベーステーブルのカラムを指定してください。

ビューを定義する **SELECT** 文の中に **GROUP BY** 句や集合関数がないか、または **UNION** 演算を伴わない場合は、ビューを更新できます。ビューに含めるテーブルは 1 つだけです。

テーブルに挿入される文字列は、データベースで大文字と小文字が区別されるかどうかに関係なく、常に入力された大文字と小文字がそのまま格納されます。このため、文字列 'Value' で更新された文字データ型カラムは、常に大文字の V と残りの文字が小文字でデータベースに格納されます。**SELECT** 文では、文字列 'Value' がされます。ただし、データベースで大文字と小文字が区別されない場合は、すべての比較で 'Value' は 'value'、'VALUE' などと同じとみなされます。IQ サーバは大文字と小文字を任意に組み合わせた結果を返すので、大文字と小文字を区別しない (**CASE IGNORE**) データベースで、大文字と小文字を区別する結果は期待できません。さらに、単一カラムのプライマリキーにエントリ 'Value' がある場合は、プライマリキーがユニークでなくなるため、**INSERT** 文による 'value' の挿入は拒否されます。

更新によってチェック制約に違反が起こる場合は、文全体がロールバックされます。

SAP Sybase IQ は、**SET** 句内でのスカラサブクエリをサポートします。次に例を示します。

```
UPDATE r
SET r.o= (SELECT MAX(t.o)
FROM t ... WHERE t.y = r.y),
r.s= (SELECT SUM(x.s)
FROM x ...
WHERE x.x = r.x)
WHERE r.a = 10
```

SAP Sybase IQ では、**UPDATE** 文でのカラムの **DEFAULT** 値がサポートされていません。カラムに **DEFAULT** 値がある場合、明示的にそのカラムの値を修正しないすべての **UPDATE** 文で、この **DEFAULT** 値がカラムの値として使用されます。

別の種類の **DEFAULT** カラムである **IDENTITY/AUTOINCREMENT** カラムの更新の詳細については、「**CREATE TABLE** 文」を参照してください。

標準

(先頭に戻る) (566 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 — IQ の **UPDATE** 文の構文は、通常、Adaptive Server の **UPDATE** 文の構文 1 と互換性があります。ただし、次の例外があります。SAP Sybase IQ では、**FROM** 句にジョイン条件を使用して、複数のテーブルを指定できます。
リモートテーブルの更新は CIS によってサポートされる SAP Sybase IQ の構文に限られます。

パーミッション

(先頭に戻る) (566 ページ)

更新するカラムに対する UPDATE 権限が必要です。

参照：

- SELECT クエリブロック内のロー制限句 (5 ページ)
- CREATE TABLE 文 (257 ページ)

UPDATE (位置付け) 文 [ESQL] [SP]

カーソルの現在の位置にあるデータを更新します。

クイックリンク：

「パラメータ」 (572 ページ)

「例」 (572 ページ)

「使用法」 (572 ページ)

「標準」 (573 ページ)

「パーミッション」 (573 ページ)

構文

```

UPDATE table-list
  SET set-item, ...
  WHERE CURRENT OF cursor-name

set-item
  column-name [.field-name...] = scalar-value

```

パラメータ

(先頭に戻る) (571 ページ)

- **cursor-name** – 識別子またはホスト変数
- **SET** – set-item で参照されるカラムは、更新対象のベーステーブルのカラムでなければなりません。エイリアスを参照したり、ほかのテーブルやビューのカラムを参照することはできません。更新対象のテーブルに対して、カーソル指定で相関名が付与されている場合は、SET 句でも相関名を使用する必要があります。SET 句の右側の式では、クエリの **SELECT** 句のカラム、定数、変数、式を参照することができます。
- **set-item** – 式に関数や式を含めることはできません。
- **WHERE CURRENT OF** – WHERE CURRENT OF での ORDER BY の使用は推奨されません。ORDER BY カラムも更新可能ですが、結果セットが再度順序付けされることはありません。そのため、結果が (間違った順番でフェッチされたように見えるため) 正しくないように見えます。

例

(先頭に戻る) (571 ページ)

- **例 1** – WHERE CURRENT OF カーソルを使用した **UPDATE** 文

```
UPDATE Employees SET surname = 'Jones'  
WHERE CURRENT OF emp_cursor
```

使用法

(先頭に戻る) (571 ページ)

この形式の **UPDATE** 文は、指定されたカーソルの現在のローを更新します。現在のローとは、直前にカーソルから正しくフェッチされたローのことです。カーソルに対する直前の処理が、位置付け **DELETE** 文であってはなりません。

指定されたクエリの現在のローの位置にあるローに対して、要求されたカラムに指定の値が設定されます。カラムは、指定の開いているカーソルの select リストに指定されていないことはありません。

位置付けされた **UPDATE** 文による変更は、カーソル結果セット内で確認できます。ただしクライアント側のキャッシュによってこれらの変更が表示されない場合を除きます。ローが更新され、開いているカーソルの WHERE 句の要件を満たさなくなった場合でも、そのローは表示されます。

SAP Sybase IQ では **CREATE VIEW... WITH CHECK OPTION** がサポートされないため、位置付けされた **UPDATE** でもこのオプションはサポートされません。WITH

CHECK OPTION 句では、ビューに表示できないローを作成する更新が許可されません。

位置付け **UPDATE** で、ロー ID カラムを更新することはできません。

SAP Sybase IQ では、結果セット内の同じローの反復的な更新がサポートされません。

標準

(先頭に戻る) (571 ページ)

- ANSI_UPDATE_CONSTRAINTS オプションが OFF に設定されている場合、更新可能なカーソルの範囲に ISO/ANSI SQL 文法のベンダ拡張が含まれる可能性があります。
- Open Client/Open Server では Embedded SQL の使用がサポートされ、SQL Anywhere ではプロシージャとトリガの使用がサポートされます。

パーミッション

(先頭に戻る) (571 ページ)

修正対象のカラムに対する UPDATE パーミッションが必要です。

参照：

- DECLARE CURSOR 文 [ESQL] [SP] (299 ページ)
- DELETE 文 (309 ページ)
- DELETE (位置付け) 文 [ESQL] [SP] (312 ページ)
- UPDATE 文 (566 ページ)

VALIDATE 文

現在のデータベース、単一のテーブル、あるいは、IQ カタログ(システム)ストア内のマテリアライズドビュー、またはインデックスを検証します。

警告！ テーブルまたはデータベース全体の検証は、データベースに変更を加えている接続がない場合に実行してください。そうしないと、実際に破損していても、何らかの形でデータベースが破損したことを示すエラーがレポートされません。

クイックリンク：

「パラメータ」 (574 ページ)

「標準」 (576 ページ)

「パーミッション」 (576 ページ)

構文

構文 1 - データベースの検証

```
VALIDATE { CHECKSUM | DATABASE }
```

構文 2 - テーブルとマテリアライズドビューの検証

```
VALIDATE {
  TABLE [ owner.]table-name
  | MATERIALIZED VIEW [ owner.]materialized-view-name }
  [ WITH EXPRESS CHECK ]
```

構文 3 - インデックスの検証

```
VALIDATE {
  INDEX index-name
  | [ INDEX ] FOREIGN KEY role-name
  | [ INDEX ] PRIMARY KEY }
  ON [ owner.]object-name

object-name
  table-name | materialized-view-name
```

構文 4 - テキストインデックスの検証

```
VALIDATE TEXT INDEX index-name
  ON [ owner.]table-name
```

パラメータ

(先頭に戻る) (573 ページ)

- CHECKSUM** - データベースの各ページのチェックサムを検証します。CHECKSUM 句によって、データベースページがディスク上で変更されていないことが確認されます。チェックサムを有効にしてデータベースを作成すると、各データベースページがディスクに書き込まれる前に、そのページのチェックサムが計算されます。CHECKSUM は、各データベースページをデータベースサーバのキャッシュ経由ではなく、ディスクから直接読み込んで、各ページのチェックサムを計算します。ページに対して計算されたチェックサムが、そのページについて格納されているチェックサムと一致しない場合は、エラーが発生し、無効なページに関する情報がデータベースサーバメッセージウィンドウに表示されます。

CHECKSUM 句は、チェックサムが無効になっているデータベースにはお勧めしません。これは、ディスクからデータベース全体が読み込まれるためです。

- DATABASE** - フリーマップでページが割り付け済みかフリーかが正しく識別され、孤立している BLOB がいないことを確認します。DATABASE 句は、チェッ

クサム検証も実行して、各データベースページが正しいオブジェクトに属していることを検証します。たとえば、テーブルページ上でテーブル ID が、定義にテーブルページのセットの最新のページが含まれる有効なテーブルを特定している必要があります。

DATABASE 句はページをデータベースサーバのキャッシュに順番に転送します。このため、検証時に、データベースサーバはキャッシュに転送されたページの内容とチェックサムを常に検証します。データベースクリーナーの実行中にデータベースの検証を開始すると、データベースクリーナーの実行が終了するまで検証は実行されません。

- **TABLE** – ベーステーブル内のすべてのローと値のセットが各インデックスに含まれるローと値のセットに一致するかどうかを確認することによって、指定されたテーブルとそのすべてのインデックスを検証します。また、TABLE 句では、すべてのテーブルの BLOB の移動、BLOB のアロケーションマップの検証、孤立した BLOB の検出も行われます。さらに、テーブルのインデックスページの物理構造を確認し、インデックスハッシュ値の順序とインデックスの一意性要件 (指定されている場合) を検証します。

外部キーインデックスの場合、WITH EXPRESS CHECK 句が指定されていない場合は、プライマリーテーブルの各値が検索されて、参照整合性制約がそのまま残っていることが確認されます。TABLE 句は、DATABASE 句と同様に、データベースサーバのキャッシュを使用するため、データベースサーバではテーブルとそのインデックスで使用されるすべてのページのチェックサムと基本的な妥当性も検証されます。

- **INDEX** – TABLE 句と同じ処理を実行しますが、指定されたインデックスだけを検証して、基になるテーブルやその他のインデックスを確認しない点が異なります。

外部キーインデックスの場合、WITH EXPRESS CHECK 句が指定されていない場合は、プライマリーテーブルの各値が検索されて、参照整合性制約がそのまま残っていることが確認されます。WITH EXPRESS CHECK 句を指定すると、参照整合性検査が無効になるため、パフォーマンスが大幅に向上する場合があります。指定されたインデックスが外部キーインデックスでない場合、WITH EXPRESS CHECK は機能しません。

- **TEXT INDEX** – インデックス内の単語に関する位置情報が残っていることを検証します。位置情報が維持されていない場合は、エラーが生成され、テキストインデックスの再作成が必要になります。テキストインデックスが自動または手動のいずれかの場合、**REFRESH TEXT INDEX** 文を実行して、テキストインデックスを再構築できます。生成されたエラーが即時テキストインデックスに関わる場合、即時インデックスを削除して、新しいインデックスを作成する必要があります。

標準

(先頭に戻る) (573 ページ)

ANSI SQL – 準拠レベル：Transact-SQL 拡張。

パーミッション

(先頭に戻る) (573 ページ)

次のいずれかが必要です。

- VALIDATE ANY OBJECT システム権限

VALIDATE LDAP SERVER 文

既存の LDAP サーバ設定オブジェクトの設定に対する変更を適用前に検証します。

クイックリンク：

「パラメータ」 (576 ページ)

「例」 (578 ページ)

「使用法」 (578 ページ)

「標準」 (579 ページ)

「パーミッション」 (579 ページ)

構文

```
VALIDATE LDAP SERVER [ ldapua-server-name | ldapua-server-attribs ]  
[ CHECK userid [ user-dn-string ] ]
```

ldapua-server-attribs

SEARCH DN

```
    URL { 'URL_string' | NULL }  
    | ACCESS ACCOUNT { 'DN_string' | NULL }  
    | IDENTIFIED BY ( 'password' | NULL )  
    | IDENTIFIED BY ENCRYPTED { encrypted-password | NULL }  
  
    | AUTHENTICATION URL { 'URL_string' | NULL }  
    | CONNECTION TIMEOUT timeout_value  
    | CONNECTION RETRIES retry_value  
    | TLS { ON | OFF }
```

パラメータ

(先頭に戻る) (576 ページ)

- **ldapua-server-name** – LDAP サーバ設定オブジェクトを指定します。
- **URL** – ホスト（名前または IP アドレスで指定）、ポート番号、指定のユーザ ID の DN ルックアップで実行される検索を指定します。この値は、LDAP URL 構文が正しいかどうかの検証後に、ISYSLDAPSERVER システムテーブルに格納されます。この文字列の最大サイズは 1024 バイトです。
- **ACCESS ACCOUNT** – SAP Sybase IQ 内のユーザではなく、SAP Sybase IQ が使用するために LDAP サーバで作成されたユーザ。このユーザの識別名 (DN) は、LDAP サーバへの接続に使用されます。このユーザは、SEARCH DN URL で指定された場所でユーザ ID によって DN を検索するためのパーミッションを、LDAP サーバ内に保持しています。この文字列の最大サイズは 1024 バイトです。
- **IDENTIFIED BY** – ACCESS ACCOUNT ユーザに関連付けられたパスワードを指定します。このパスワードは、対称暗号化を使用してディスクに保存されます。パスワードを解除して何も設定しない場合は、値 NULL を指定します。クリアテキストのパスワードの最大サイズは 255 バイトです。
- **IDENTIFIED BY ENCRYPTED** – ACCESS ACCOUNT 識別名に関連付けられるパスワードを暗号化形式で設定します。バイナリ値は暗号化されたパスワードであるため、ディスクにそのまま保存されます。パスワードを解除して何も設定しない場合は、値 NULL を指定します。バイナリの最大サイズは 289 バイトです。
- **AUTHENTICATION URL** – ユーザの認証に使用する LDAP サーバのホスト (名前または IP アドレスで指定) とポート番号を指定します。これは、<URL_string> として定義された値で、LDAP URL 構文が正しいかどうかの検証後に ISYSLDAPSERVER システムテーブルに格納されます。事前の DN 検索によって得られたユーザの DN とユーザパスワードによって、新しい接続が認証 URL にバインドされます。LDAP サーバへの正常な接続は、接続ユーザの ID の証明とみなされます。この文字列の最大サイズは 1024 バイトです。
- **CONNECTION TIMEOUT** – DN 検索と認証の両方に使用する SAP Sybase IQ から LDAP サーバへの接続のタイムアウトを指定します。この値はミリ秒で指定します。デフォルト値は 10 秒です。
- **CONNECTION RETRIES** – DN 検索と認証の両方に使用する SAP Sybase IQ から LDAP サーバへの接続の再試行回数を指定します。有効値の範囲は 1 ~ 60 で、デフォルト値は 3 です。
- **TLS** – DN 検索と認証の両方に使用する LDAP サーバへの接続に、TLS とセキュア LDAP プロトコルのいずれを使用するかを定義します。ON に設定すると、TLS プロトコルが使用され、URL は "ldap://" で始まります。OFF に設定すると

(または指定なしにすると)、セキュア LDAP プロトコルが使用され、URL は “ldaps://” で始まります。TLS プロトコルを使用する場合は、LDAP サーバで使用される証明書に署名した認証局 (CA) の証明書が含まれているファイル名を使用して、データベースセキュリティオプション TRUSTED_CERTIFICATES_FILE を指定します。

- **CHECK userID** – LDAP サーバで存在が検証される userID。
- **user-dn-string** – 検証を目的としてユーザの DN 値をユーザ ID と比較します。

例

(先頭に戻る) (576 ページ)

- **例 1** – apps_primary LDAP サーバ設定オブジェクトが次のように作成されているとします。

```
SET OPTION PUBLIC.login_mode = 'Standard,LDAPUA'
CREATE LDAP SERVER apps_primary
SEARCH DN
    URL 'ldap://my_LDAPserver:389/dc=MyCompany,dc=com??sub?cn=*'
    ACCESS ACCOUNT 'cn=aseadmin, cn=Users, dc=mycompany, dc=com'
    IDENTIFIED BY 'Secret99Password'
AUTHENTICATION URL 'ldap://my_LDAPserver:389/'
CONNECTION TIMEOUT 3000
WITH ACTIVATE
```

次の文は、userID の myusername が存在することを、オプションの CHECK 句を使用し、apps_primary LDAP サーバ設定オブジェクトで予測されるユーザの識別名 (引用符で囲まれている) と比較することによって検証します。

```
VALIDATE LDAP SERVER apps_primary
CHECK myusername 'cn=myusername, cn=Users, dc=mycompany, dc=com'
```

- **例 2** – 検索属性が含まれている場合、LDAP サーバ設定オブジェクトの名前が **VALIDATE LDAP SERVER** 文で定義されている必要はありません。

```
VALIDATE LDAP SERVER
SEARCH DN
    URL 'ldap://my_LDAPserver:389/dc=MyCompany,dc=com??sub?cn=*'
    ACCESS ACCOUNT 'cn=aseadmin, cn=Users, dc=mycompany, dc=com'
    IDENTIFIED BY 'Secret99Password'
AUTHENTICATION URL 'ldap://my_LDAPserver:389/'
CONNECTION TIMEOUT 3000
CHECK myusername 'cn=myusername, cn=Users, dc=mycompany, dc=com'
```

使用法

(先頭に戻る) (576 ページ)

この文は、管理者が LDAP ユーザ認証が使用されるように新しいサーバを設定する場合、および LDAP サーバ設定オブジェクトと外部 LDAP サーバ間の問題を診断する場合に便利です。**VALIDATE LDAP SERVER** 文による接続はすべて一時的で、この文の終了により切断されます。

名前で LDAP サーバ設定オブジェクトを検証する際は、以前の **CREATE LDAP SERVER** 文と **ALTER LDAP SERVER** 文の定義が使用されます。また、*ldapua-server-attributes* が LDAP サーバ設定オブジェクト名の代わりに指定された場合、指定の属性が検証されます。*ldapua-server-attributes* が指定された場合、構文エラーを識別するために URL が解析され、構文エラーが検出されると、文の処理が停止します。

LDAP サーバ設定オブジェクト名か、正常に解析された *ldapua-server-attributes* セットのいずれかを使用して、外部 LDAP サーバへの接続が試行されます。パラメータ **ACCESS ACCOUNT** とパスワードが指定されている場合、この値が **SEARCH DN URL** への接続確立に使用されます。構成は **SEARCH DN URL**、**ACCESS ACCOUNT**、および **ACCESS ACCOUNT** のパスワードになります。

オプションの **CHECK** 句を使用すると、外部 LDAP サーバ上のユーザの存在の検証に検索内の **userID** が使用されます。指定ユーザの予想される **DN** 値が既知の場合はこの値を指定でき、この値が検索結果と比較され、成功または失敗が判断されます。

標準

(先頭に戻る) (576 ページ)

ANSI SQL – 準拠レベル：Transact-SQL 拡張。

パーミッション

(先頭に戻る) (576 ページ)

MANAGE ANY LDAP SERVER システム権限が必要です。

WAITFOR 文

この文は、指定した時間が経過するか特定の時刻になるまで、現在の接続の処理を遅らせます。

クイックリンク：

「パラメータ」 (580 ページ)

「例」 (580 ページ)

「使用法」 (581 ページ)

「標準」 (581 ページ)

「パーミッション」 (581 ページ)

構文

```
WAITFOR {  
    DELAY time_value | TIME time_value }  
    [ CHECK EVERY integer ]  
    [ AFTER MESSAGE BREAK ]
```

パラメータ

(先頭に戻る) (579 ページ)

- **DELAY** – *time_value* で指定された期間だけ処理がサスペンドします。
- **TIME** – サーバの時間が指定された *time_value* に達するまで、処理がサスペンドされます。
- **time_value** – 文字列
- **CHECK EVERY** – **WAITFOR** 文が起動する頻度を制御します。デフォルトでは 5 秒ごとに **WAITFOR** が起動します。値はミリ秒単位であり、最小値は 250 ミリ秒です。
- **AFTER MESSAGE BREAK** – **WAITFOR** 文を使用すると、別の接続からのメッセージを待つことができます。ほとんどの場合、メッセージが受信されると、**WAITFOR** 文を実行したアプリケーションに転送され、**WAITFOR** 文は待機を継続します。AFTER MESSAGE BREAK 句が指定されている場合、別の接続からのメッセージが受信されると、**WAITFOR** 文が完了します。メッセージテキストはアプリケーションに転送されませんが、MessageReceived 接続プロパティの値を取得することによってアクセスすることができます。

例

(先頭に戻る) (579 ページ)

- **例 1** – 3 秒間待機します。

```
WAITFOR DELAY '00:00:03'
```

- **例 2** – 0.5 秒 (500 ミリ秒) 間待機します。

```
WAITFOR DELAY '00:00:00:500'
```

- **例 3** – 午後 8 時まで待機します。


```
WAITFOR TIME '20:00'
```

使用法

(先頭に戻る) (579 ページ)

WAITFOR 文は、定期的 (デフォルトでは 5 秒おき) に起動して、キャンセルされたかどうか、またはメッセージが受信されたかどうかをチェックします。いずれの処理も確認されなかった場合は、文は待機を継続します。

現在のサーバの時間が指定の時間を過ぎている場合は、次の日のその時間まで処理がサスペンドします。

WAITFOR は、次の文と同じ目的で使用できます。顧客がデータベースで Java を無効にすることを選択した場合に利用すると便利です。

```
call java.lang.Thread.sleep( <time_to_wait_in_millisecs> )
```

通常は、**WAITFOR TIME** を使用するよりも、スケジュールされたイベントを使用することをおすすめします。これは、スケジュールされたイベントは、専用接続で実行されるためです。

関連する動作：

- この文の実装では、待機中にワークスレッドが使用されます。これにより、-gn サーバコマンドラインオプションによって指定されたスレッドの 1 つが使用されます。

標準

(先頭に戻る) (579 ページ)

- SQL — ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - この文は Adaptive Server でも実装されています。

パーミッション

(先頭に戻る) (579 ページ)

なし

参照：

- CREATE EVENT 文 (157 ページ)

WHENEVER 文 [ESQL]

Embedded SQL プログラム内でのエラー処理を指定します。

クイックリンク：

「例」 (582 ページ)

「使用法」 (582 ページ)

「標準」 (583 ページ)

「パーミッション」 (583 ページ)

構文

WHENEVER

```
{ SQLERROR | SQLWARNING | NOTFOUND }  
... { GOTO label | STOP | CONTINUE | C code; }
```

例

(先頭に戻る) (582 ページ)

- **例 1** – この例は、NOTFOUND 句が満たされた場合に done を実行します。

```
EXEC SQL WHENEVER NOTFOUND GOTO done;
```

- **例 2** – この例は、SQLERROR 句を使用します。

```
EXEC SQL WHENEVER SQLERROR  
{  
    PrintError( &sqlca );  
    return( FALSE );  
};
```

使用法

(先頭に戻る) (582 ページ)

WHENEVER は Embedded SQL C プログラム内の任意の位置に入れることができます。この文はコードを生成しません。プリプロセッサは、それぞれの継続する SQL 文の後にコードを生成します。すべての Embedded SQL 文では、**WHENEVER** 文のソース行から同じエラー条件の次の **WHENEVER** 文、またはソースファイルの最後まで、エラーアクションは有効です。

デフォルトアクションは **CONTINUE** です。

WHENEVER は、単純なプログラムの中で便利に使用できます。ほとんどの場合、SQLCA (SQLCODE) の sqlcode フィールドを直接検査することがエラー状態の確

認の最も簡単な方法です。この場合、**WHENEVER** は使用されません。**WHENEVER** 文が実行されると、それぞれの文の後に、プリプロセッサによって *if(SQLCODE)* テストが生成されます。

注意： エラー条件は、文がいつ実行されたかではなく、C 言語ソースファイルの位置に基づいて効力を持ちます。

標準

(先頭に戻る) (582 ページ)

- SQL — ISO/ANSI SQL 文法のベンダ拡張。
- SAP Sybase Database 製品 - Open Client/Open Server でサポートされています。

パーミッション

(先頭に戻る) (582 ページ)

なし

WHILE 文 [T-SQL]

文または複合文を繰り返し実行します。

クイックリンク：

「例」 (583 ページ)

「使用法」 (584 ページ)

「標準」 (584 ページ)

「パーミッション」 (584 ページ)

構文

```
WHILE expression  
... statement
```

例

(先頭に戻る) (583 ページ)

- **例 1**– 最高額の製品の価格が 50 ドル未満の場合、**BREAK** 文によって **WHILE** ループがブレイクされ、それ以外の場合は、平均価格が 30 ドルを超えるまでループが続行します。

```
WHILE (SELECT AVG(unit_price) FROM Products) < 30  
BEGIN
```

SQL 文

```
DELETE FROM Products
WHERE UnitPrice = MAX(UnitPrice)
IF ( SELECT MAX(UnitPrice) FROM Products ) < 50
  BREAK
END
```

使用法

(先頭に戻る) (583 ページ)

WHILE 条件は、単一の SQL 文と、**BEGIN** と **END** のキーワードで囲まれた複合文にまとめられている SQL 文にのみ影響します。

複合文中での文の実行は、**BREAK** 文と **CONTINUE** 文で制御できます。**BREAK** 文はループを終了し、ループの最後を示す **END** キーワードの後から実行が再開されます。**CONTINUE** 文は、**CONTINUE** の後の文をすべて省略して **WHILE** ループを再開します。

標準

(先頭に戻る) (583 ページ)

- SQL – ISO/ANSI SQL 文法の Transact-SQL 拡張。
- SAP Sybase Database 製品 - Adaptive Server でサポートされています。

パーミッション

(先頭に戻る) (583 ページ)

なし

参照:

- **BEGIN ... END** 文 (108 ページ)

データベースオプション

データベースオプションと Interactive SQL オプションは、データベースの動作をカスタマイズしたり、変更したりします。SAP Sybase IQ データベースオプションは、一般的なオプション、Transact-SQL 互換性オプション、Interactive SQL オプションの3つのクラスに分類されます。

データベースオプションの概要

データベースオプションは、互換性、エラー処理、同時実行性などさまざまな面からデータベースの動作を制御します。

たとえば、次の目的でデータベースオプションを使用できます。

- 互換性 – 使用する SAP Sybase IQ データベースがどの程度 Adaptive Server と同様の動作をするか、および SQL92 に準拠していない SQL でエラーを生成するかどうかを制御できます。
- エラー処理 – 0 で割ろうとしたとき、またはオーバフローエラーなどのエラーが発生したときの処理を制御できます。
- 同時実行性とトランザクション – 同時実行性の程度と COMMIT 動作の詳細をオプションを使用して制御できます。

SET OPTION 文を使用してオプションを設定します。一般的な構文は次のとおりです。

```
SET [ EXISTING ] [ TEMPORARY ] OPTION
... [ userid. | PUBLIC. ] option-name = [ option-value ]
```

ユーザ ID またはロール名を指定し、そのユーザまたはロールのみに対するオプションを設定します。PUBLIC ロールにはすべてのユーザが属します。ユーザ ID またはロールを指定しない場合、オプション変更は **SET OPTION** 文を発行した、現在ログオン中のユーザ ID に適用されます。

たとえば、次の文は、すべてのユーザが属するロールである PUBLIC ユーザ ID に変更を適用します。

```
SET OPTION Public.login_mode = standard
```

注意： ユーザまたはロールを指定せずにオプションを TEMPORARY に設定した場合、新しいオプション値は、この文を発行した現在ログオン中のユーザ ID のみに、その接続の存続期間だけ適用されます。PUBLIC ロールに対してオプションを TEMPORARY に設定すると、その変更はデータベースが実行されている限り存

続きます。データベースが停止すると、PUBLIC ロールに対する TEMPORARY オプションは永続値に戻ります。

TEMPORARY キーワードを発行せずにオプションを設定した場合、新しいオプション値は、文を発行したユーザまたはロールの永続値となります。

テンポラリオプション値と永久オプション値の詳細については、「データベースオプションの範囲と継続期間」、「temporary オプション」、「SET OPTION 文」を参照してください。

option-value の最大長は、文字列を設定する場合は 127 バイトです。

注意： 整数値で指定されるデータベースオプションは、SAP Sybase IQ によって *option-value* の小数値設定がすべて整数値にトランケートされます。たとえば、3.8 という値は 3 にトランケートされます。

警告！ ローをフェッチしている間にオプション設定を変更しないでください。

参照：

- データベースオプションの範囲と継続期間 (587 ページ)
- temporary オプション (588 ページ)
- SET OPTION 文 (542 ページ)

現在のオプション設定

オプション設定のリスト、または個別のオプションの値は、**sp_iqcheckoptions**、**sa_conn_properties**、SET 文、SAP Control Center、および SYSOPTIONS システムビューを使用して取得できます。

- 接続しているユーザは、**sp_iqcheckoptions** ストアドプロシージャを使用して、デフォルトから変更されているデータベースオプションの現在値とデフォルト値のリストを表示できます。**sp_iqcheckoptions** では SAP Sybase IQ と SQL Anywhere のすべてのデータベースオプションが考慮されます。SAP Sybase IQ は一部の SQL Anywhere オプションのデフォルトを変更して、その変更値が新しいデフォルト値になります。新しい SAP Sybase IQ のデフォルト値が再変更されない限り、**sp_iqcheckoptions** はそのオプションをリストに含めません。

sp_iqcheckoptions は、デフォルト値から変更されたサーバ起動オプションも一覧表示します。

DBA が **sp_iqcheckoptions** を実行すると、DBA にはすべてのロールとユーザに永続的に設定されているすべてのオプションと、DBA に設定されているテンポラリオプションが表示されます。DBA 以外のユーザには、そのユーザ自身のテンポラリオプションが表示されます。デフォルトでないサーバ起動オプションはすべてのユーザに表示されます。

sp_iqcheckoptions ストアドプロシージャには、パラメータを指定する必要はありません。Interactive SQL で、次のコマンドを実行します。

```
sp_iqcheckoptions
```

システムテーブル DBA.SYSOPTIONDEFAULTS には、SAP Sybase IQ と SQL Anywhere のオプションすべての名前とデフォルト値が格納されます。このテーブルにクエリを実行すると、すべてのオプションのデフォルト値を参照できます。

- 現在の接続のオプション設定は、「接続プロパティ」から取得できます。**sa_conn_properties** システムプロシージャを使用すると、すべての接続プロパティをリストできます。

```
call sa_conn_properties
```

- Interactive SQL では、引数なしで **SET** 文を使用すると、現在のオプション設定がリストされます。

```
SET
```

- SAP Control Center では、データベースを右クリックしてサブメニューから [オプション] を選択します。
- SYSOPTIONS システムビューに対して次のようにクエリを実行します。

```
SELECT *
FROM SYSOPTIONS
```

これによって明示的に設定されているすべての PUBLIC 値、および USER 値が表示されます。

- 個々の設定は **connection_property** システム関数を使用して確認できます。たとえば、次の文で Ansinull オプションの値がわかります。

```
SELECT connection_property ('Ansinull')
```

データベースオプションの範囲と継続期間

オプションは、パブリック、ユーザ、テンポラリの3つの範囲レベルで設定できます。

temporary オプションは、user と public より優先度が高くなっています。ユーザレベルのオプションは public より優先度が高くなっています。ユーザレベルのオプションを現在のユーザに対して設定すると、対応する temporary オプションも同じように設定されます。

COMMIT の動作などの一部のオプションは、データベース全体を範囲とします。これらのオプションを設定するには、DBA パーミッションが必要です。ISOLATION_LEVEL などの他のオプションは、現在の接続のみに適用でき、特別なパーミッションは必要ありません。

オプション設定への変更が有効になる時点は、オプションによって異なります。RECOVERY_TIME などのグローバルオプションの変更は、サーバの次回起動時に

データベースオプション

有効になります。次のリストは、サーバの再起動時に有効化されるオプションの一部を示しています。

サーバの再起動を必要とするデータベースオプション
CACHE_PARTITIONS
CHECKPOINT_TIME
OS_FILE_CACHE_BUFFERING
OS_FILE_CACHE_BUFFERING_TEMPDB
PREFETCH_BUFFER_LIMIT
PREFETCH_BUFFER_PERCENT
RECOVERY_TIME
SWEEPER_THREADS_PERCENT
WASH_AREA_BUFFERS_PERCENT

一般に、現在の接続にのみ影響を与えるオプションはすぐに有効になります。たとえば、トランザクションの途中でもオプション設定を変更できます。

警告！ カーソルが開いている間にオプションを変更すると、結果の信頼性が損なわれる場合があります。たとえば、カーソルが開いているときに DATE_FORMAT を変更しても、次のローのフォーマットは変わりません。カーソルの取得方法によっては、ユーザが指定した変更が有効になるまでにいくつかのローが処理されてしまう場合があります。

temporary オプション

TEMPORARY キーワードを SET OPTION 文に追加すると、変更の存続期間を変更できます。

通常、オプションの変更は永続的です。**SET OPTION** 文を使用して明示的に変更されるまで変わりません。

SET TEMPORARY OPTION 文で指定された新しいオプション値は、現在の接続のみに適用され、現在の接続の間だけ持続します。

SET TEMPORARY OPTION を使用して PUBLIC オプションを設定すると、データベースが実行中である間だけ、その変更が継続します。データベースが停止すると、PUBLIC ユーザ ID に対する TEMPORARY オプションは元の永続値に戻ります。

PUBLIC ユーザ ID に対するオプションを設定すると、セキュリティに一時的な利点をもたらします。たとえば、LOGIN_MODE オプションが有効な場合、データベースは、そのデータベースを実行しているシステムのログインセキュリティに

依存します。LOGIN_MODEを一時的に有効にすると、Windows ドメインのセキュリティに依存しているデータベースは、データベースが停止し、ローカルマシンにコピーされるといった場合でも、セキュリティが損なわれることはありません。この場合、LOGIN_MODE オプションは、永続値に戻ります。永続値は Standard で、統合ログインが許可されないモードです。

PUBLIC オプション

PUBLIC オプションは、ユーザ、ユーザ拡張ロール、または PUBLIC ロールに設定できます。自分で設定することも、別のユーザまたはロールに対して設定することもできます。

PUBLIC ロールに対して PUBLIC オプションを設定すると、ユーザレベルで PUBLIC オプションを設定していないすべてのユーザにこの値が設定されます。ユーザまたはユーザ拡張ロールに対して PUBLIC オプションを設定すると、PUBLIC ロールレベルで定義されたすべての値が上書きされます。

自分に対して PUBLIC オプションを設定する場合に特に必要なシステム権限はありません。しかし、別のユーザ、ユーザ拡張ロール、または PUBLIC ロールに対してこのオプションを設定する場合は、SET ANY PUBLIC OPTION システム権限が必要です。PUBLIC オプションをユーザ定義ロールに対して設定することはできません。PUBLIC データベースオプションはすぐに有効になります。変更を有効にするために、データベースサーバを停止してから再起動する必要はありません。

SECURITY オプション

SECURITY オプションはデータベースのセキュリティに関する特殊なカテゴリです。このオプションはユーザレベルまたは PUBLIC レベル (オプションによって異なる) で設定できます。

SECURITY データベースオプションに対する変更はすぐに反映されます。SECURITY データベースオプションを設定するには、SET ANY SECURITY OPTION システム権限が必要です。

変更を有効にするために、データベースサーバを停止してから再起動する必要はありません。

SYSTEM オプション

SYSTEM オプションはデータベースのセキュリティに関する特殊なカテゴリです。このオプションはユーザレベルまたは PUBLIC レベルで設定できます。

SYSTEM オプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。すぐに有効になります。

オプション設定の削除

option-value を省略すると、指定したオプション設定がデータベースから削除されます。

option-value が各ユーザ固有のオプション設定の場合は、値は PUBLIC 設定に戻ります。TEMPORARY オプションを削除すると、その設定値は永久値に戻ります。

たとえば、次の文を実行すると ANSINULL オプションはデフォルト値に戻ります。

```
SET OPTION ANSINULL =
```

オプション設定時にオプション名を誤って入力すると、誤った名前が SYSOPTION テーブルに保存されます。誤って入力した名前を SYSOPTION テーブルから削除するには、オプション PUBLIC でオプション名の後に等号を付け、値を指定しません。

```
SET OPTION PUBLIC.a_mistyped_name=;
```

たとえば、あるオプションを設定するときに名前を誤って入力した場合、SYSOPTIONS ビューで選択することでそのオプションが保存されたことを確認できます。

```
SET OPTION PUBLIC.a_mistyped_name='ON';
SELECT * FROM SYSOPTIONS ORDER BY 2;
```

user_name	オプション	設定
PUBLIC	a_mistyped_name	ON
PUBLIC	Abort_On_Error_File	
PUBLIC	Abort_On_Error_Line	0
PUBLIC	Abort_On_Error_Number	0
...		

誤って入力したオプションは、値を設定しないことで削除し、その後、オプションが削除されたことを確認します。

```
SET OPTION PUBLIC.a_mistyped_name=;
SELECT * FROM SYSOPTIONS ORDER BY 2;
```

user_name	オプション	設定
PUBLIC	Abort_On_Error_File	
PUBLIC	Abort_On_Error_Line	0

user_name	オプション	設定
PUBLIC	Abort_On_Error_Number	0
...		

PUBLIC オプションを削除した後に、USER オプションを追加しようとすると、次のエラーメッセージが表示されます。

```
Couldn't execute the statement.
Invalid option 'chained' -- no PUBLIC setting exists
SQLCODE=-200?ODBC 3 State="42000"
Line 1,Column 29
```

PUBLIC オプションをそのデフォルト値にリセットするには、デフォルト値を明示的に設定します。

```
SET OPTION PUBLIC.chained ='ON';
```

オプションの初期設定

ストアードプロシージャを使用して、ユーザのデータベースオプションを初期設定できます。

SAP Sybase IQ へは、TDS (Tabular Data Stream) プロトコル (Open Client and jConnect™ for JDBC™ 接続)、または SAP Sybase IQ プロトコル (ODBC、Embedded SQL) を介して接続できます。

ユーザが TDS プロトコルと SAP Sybase IQ 固有のプロトコルの両方を使用できる場合は、ストアードプロシージャを使用して初期設定を設定することができます。SAP Sybase IQ の出荷時設定では、Adaptive Server のデフォルトの動作を反映させるため、この方法を使用して Open Client 接続と jConnect 接続を設定しています。

初期設定は、LOGIN_PROCEDURE オプションを使用して制御されます。これは接続の有効性を検証するすべてのチェックの実行後に呼び出されます。

LOGIN_PROCEDURE オプションでは、ユーザの接続時に実行されるストアードプロシージャが指定されます。デフォルト設定では、**sp_login_environment** システムストアードプロシージャが使用されます。異なるストアードプロシージャを指定できます。

sp_login_environment は、接続が TDS によって行われたかどうかを確認します。TDS によって行われている場合は、**sp_tsql_environment** プロシージャを呼び出して、いくつかのオプションに現在の接続に合った新しいデフォルト値を設定します。

参照：

- LOGIN_PROCEDURE オプション (708 ページ)

廃止予定のデータベースオプション

このリリースで廃止予定のデータベースオプションの詳細については、『新機能の概要 SAP Sybase IQ 16.0』を参照してください。

一般的なデータベースオプション

一般的なデータベースオプションは、Transact-SQL 互換性オプションと Interactive SQL オプションを除くすべてのオプションで構成されるクラスです。

参照：

- Transact-SQL 互換性オプション (598 ページ)
- Interactive SQL オプション (600 ページ)
- AFFINITY_AUTOEXCLUDE_TIMEOUT オプション (601 ページ)
- AGGREGATION_PREFERENCE オプション (602 ページ)
- ALLOW_SNAPSHOT_VERSIONING オプション (604 ページ)
- ANSI_UPDATE_CONSTRAINTS オプション (608 ページ)
- ALLOW_READ_CLIENT_FILE オプション (610 ページ)
- ASE_BINARY_DISPLAY オプション (610 ページ)
- ASE_FUNCTION_BEHAVIOR オプション (611 ページ)
- AUDITING オプション [データベース] (612 ページ)
- BASE_TABLES_IN_RLV_STORE オプション (613 ページ)
- BIT_VECTOR_PINNABLE_CACHE_PERCENT オプション (614 ページ)
- BLOCKING オプション (615 ページ)
- BLOCKING_TIMEOUT オプション (616 ページ)
- BT_PREFETCH_MAX_MISS オプション (616 ページ)
- BT_PREFETCH_SIZE オプション (617 ページ)
- BTREE_PAGE_SPLIT_PAD_PERCENT オプション (618 ページ)
- CACHE_AFFINITY_PERCENT オプション (619 ページ)
- CACHE_PARTITIONS オプション (620 ページ)
- CHECKPOINT_TIME オプション (622 ページ)
- CIS_ROWSET_SIZE オプション (623 ページ)
- CONVERSION_MODE オプション (626 ページ)
- CONVERT_VARCHAR_TO_1242 オプション (632 ページ)
- COOPERATIVE_COMMIT_TIMEOUT オプション (633 ページ)
- COOPERATIVE_COMMITS オプション (634 ページ)

- CREATE_HG_WITH_EXACT_DISTINCTS (635 ページ)
- CURSOR_WINDOW_ROWS オプション (637 ページ)
- DATE_FIRST_DAY_OF_WEEK オプション (638 ページ)
- DATE_FORMAT オプション (639 ページ)
- DATE_ORDER オプション (641 ページ)
- DBCC_LOG_PROGRESS オプション (642 ページ)
- DBCC_PINNABLE_CACHE_PERCENT オプション (643 ページ)
- DEBUG_MESSAGES オプション (643 ページ)
- DEDICATED_TASK オプション (644 ページ)
- DEFAULT_DBSPACE オプション (645 ページ)
- DEFAULT_DISK_STRIPING オプション (647 ページ)
- DEFAULT_HAVING_SELECTIVITY_PPM オプション (647 ページ)
- DEFAULT_KB_PER_STRIPE オプション (649 ページ)
- DEFAULT_LIKE_MATCH_SELECTIVITY_PPM オプション (650 ページ)
- DEFAULT_LIKE_RANGE_SELECTIVITY_PPM オプション (651 ページ)
- DEFAULT_PROXY_TABLE_ROW_COUNT オプション (652 ページ)
- DEFAULT_TABLE_UDF_ROW_COUNT オプション (652 ページ)
- DELAYED_COMMIT_TIMEOUT オプション (653 ページ)
- DELAYED_COMMITS オプション (654 ページ)
- DISABLE_RI_CHECK オプション (654 ページ)
- DQP_ENABLED オプション (656 ページ)
- DQP_ENABLED_OVER_NETWORK オプション (657 ページ)
- EARLY_PREDICATE_EXECUTION オプション (658 ページ)
- ENABLE_ASYNC_IO オプション (659 ページ)
- ENABLE_LOB_VARIABLES オプション (660 ページ)
- EXTENDED_JOIN_SYNTAX オプション (661 ページ)
- FLOATING_POINT_ACCUMULATOR オプション (661 ページ)
- FORCE_DROP オプション (662 ページ)
- FORCE_NO_SCROLL_CURSORS オプション (663 ページ)
- FORCE_UPDATABLE_CURSORS オプション (664 ページ)
- FP_LOOKUP_SIZE オプション (665 ページ)
- FP_LOOKUP_SIZE_PPM オプション (666 ページ)
- FP_NBIT_AUTOSIZE_LIMIT オプション (667 ページ)
- FP_NBIT_ENFORCE_LIMITS オプション (668 ページ)
- FP_NBIT_IQ15_COMPATIBILITY オプション (670 ページ)

- FP_NBIT_LOOKUP_MB オプション (672 ページ)
- FP_NBIT_ROLLOVER_MAX_MB オプション (673 ページ)
- FP_PREDICATE_WORKUNIT_PAGES オプション (675 ページ)
- FPL_EXPRESSION_MEMORY_KB オプション (675 ページ)
- GARRAY_FILL_FACTOR_PERCENT オプション (676 ページ)
- GARRAY_INSERT_PREFETCH_SIZE オプション (677 ページ)
- GARRAY_PAGE_SPLIT_PAD_PERCENT オプション (678 ページ)
- GARRAY_RO_PREFETCH_SIZE オプション (679 ページ)
- HASH_PINNABLE_CACHE_PERCENT オプション (679 ページ)
- HASH_THRASHING_PERCENT オプション (680 ページ)
- HG_DELETE_METHOD オプション (681 ページ)
- HG_SEARCH_RANGE オプション (682 ページ)
- HTTP_SESSION_TIMEOUT オプション (683 ページ)
- IDENTITY_ENFORCE_UNIQUENESS オプション (684 ページ)
- IDENTITY_INSERT オプション (684 ページ)
- IN_SUBQUERY_PREFERENCE オプション (686 ページ)
- INDEX_ADVISOR オプション (687 ページ)
- INDEX_ADVISOR_MAX_ROWS オプション (690 ページ)
- INDEX_PREFERENCE オプション (691 ページ)
- INFER_SUBQUERY_PREDICATES オプション (692 ページ)
- IQGOVERN_MAX_PRIORITY オプション (693 ページ)
- IQGOVERN_PRIORITY オプション (694 ページ)
- IQGOVERN_PRIORITY_TIME オプション (694 ページ)
- ISOLATION_LEVEL オプション (695 ページ)
- JAVA_LOCATION オプション (696 ページ)
- JAVA_VM_OPTIONS オプション (696 ページ)
- JOIN_EXPANSION_FACTOR オプション (697 ページ)
- JOIN_OPTIMIZATION オプション (698 ページ)
- JOIN_PREFERENCE オプション (700 ページ)
- JOIN_SIMPLIFICATION_THRESHOLD オプション (702 ページ)
- LF_BITMAP_CACHE_KB オプション (703 ページ)
- LOAD_ZEROLENGTH_ASNULL オプション (704 ページ)
- LOG_CONNECT オプション (705 ページ)
- LOG_CURSOR_OPERATIONS オプション (705 ページ)
- LOG_DEADLOCKS オプション (706 ページ)

- LOGIN_MODE オプション (707 ページ)
- LOGIN_PROCEDURE オプション (708 ページ)
- MAIN_RESERVED_DBSPACE_MB オプション (709 ページ)
- MAX_CARTESIAN_RESULT オプション (709 ページ)
- MAX_CLIENT_NUMERIC_PRECISION オプション (710 ページ)
- MAX_CLIENT_NUMERIC_SCALE オプション (711 ページ)
- MAX_CUBE_RESULT オプション (712 ページ)
- MAX_CURSOR_COUNT オプション (713 ページ)
- MAX_HASH_ROWS オプション (714 ページ)
- MAX_IQ_THREADS_PER_CONNECTION オプション (715 ページ)
- MAX_IQ_THREADS_PER_TEAM オプション (715 ページ)
- MAX_JOIN_ENUMERATION オプション (716 ページ)
- MAX_PARTITIONED_HASH_MB オプション (717 ページ)
- MAX_PREFIX_PER_CONTAINS_PHRASE オプション (718 ページ)
- MAX_QUERY_PARALLELISM オプション (719 ページ)
- MAX_QUERY_TIME オプション (719 ページ)
- MAX_STATEMENT_COUNT オプション (720 ページ)
- MAX_TEMP_SPACE_PER_CONNECTION オプション (721 ページ)
- MINIMIZE_STORAGE オプション (722 ページ)
- MIN_PASSWORD_LENGTH オプション (724 ページ)
- MIN_ROLE_ADMINS オプション (724 ページ)
- MONITOR_OUTPUT_DIRECTORY オプション (725 ページ)
- MPX_AUTOEXCLUDE_TIMEOUT オプション (726 ページ)
- MPX_HEARTBEAT_FREQUENCY オプション (727 ページ)
- MPX_IDLE_CONNECTION_TIMEOUT オプション (727 ページ)
- MPX_LIVENESS_TIMEOUT オプション (727 ページ)
- MPX_MAX_CONNECTION_POOL_SIZE オプション (728 ページ)
- MPX_MAX_UNUSED_POOL_SIZE オプション (729 ページ)
- MPX_WORK_UNIT_TIMEOUT オプション (730 ページ)
- NOEXEC オプション (731 ページ)
- NON_ANSI_NULL_VARCHAR オプション (732 ページ)
- NOTIFY_MODULUS オプション (734 ページ)
- ODBC_DISTINGUISH_CHAR_AND_VARCHAR オプション (734 ページ)
- ON_CHARSET_CONVERSION_FAILURE オプション (735 ページ)
- OS_FILE_CACHE_BUFFERING オプション (738 ページ)

- OS_FILE_CACHE_BUFFERING_TEMPDB オプション (739 ページ)
- POST_LOGIN_PROCEDURE オプション (741 ページ)
- PRECISION オプション (742 ページ)
- PREFETCH オプション (743 ページ)
- PREFETCH_BUFFER_LIMIT オプション (744 ページ)
- PREFETCH_BUFFER_PERCENT オプション (744 ページ)
- PREFETCH_GARRAY_PERCENT オプション (745 ページ)
- PREFETCH_SORT_PERCENT オプション (745 ページ)
- PRESERVE_SOURCE_FORMAT オプション [database] (746 ページ)
- QUERY_DETAIL オプション (747 ページ)
- QUERY_NAME オプション (748 ページ)
- QUERY_PLAN オプション (748 ページ)
- QUERY_PLAN_AFTER_RUN オプション (749 ページ)
- QUERY_PLAN_AS_HTML オプション (750 ページ)
- QUERY_PLAN_AS_HTML_DIRECTORY オプション (752 ページ)
- QUERY_PLAN_MIN_TIME オプション (754 ページ)
- QUERY_PLAN_TEXT_ACCESS オプション (754 ページ)
- QUERY_PLAN_TEXT_CACHING オプション (755 ページ)
- QUERY_ROWS_RETURNED_LIMIT オプション (756 ページ)
- QUERY_TEMP_SPACE_LIMIT オプション (757 ページ)
- QUERY_TIMING オプション (758 ページ)
- RECOVERY_TIME オプション (760 ページ)
- RESERVED_KEYWORDS オプション (760 ページ)
- RETURN_DATE_TIME_AS_STRING オプション (761 ページ)
- REVERT_TO_V15_OPTIMIZER オプション (762 ページ)
- ROUND_TO_EVEN オプション (763 ページ)
- ROW_COUNT オプション (764 ページ)
- RV_AUTO_MERGE_EVAL_INTERVAL オプション (765 ページ)
- RV_MERGE_NODE_MEMSIZE オプション (766 ページ)
- RV_MERGE_TABLE_MEMPERCENT オプション (767 ページ)
- RV_MERGE_TABLE_NUMROWS オプション (768 ページ)
- RV_RESERVED_DBSPACE_MB オプション (768 ページ)
- SCALE オプション (769 ページ)
- SNAPSHOT_VERSIONING オプション (770 ページ)
- SIGNIFICANTDIGITSFORDOUBLEEQUALITY オプション (771 ページ)

- SORT_COLLATION オプション (772 ページ)
- SORT_PINNABLE_CACHE_PERCENT オプション (773 ページ)
- SUBQUERY_CACHING_PREFERENCE オプション (777 ページ)
- SUBQUERY_FLATTENING_PERCENT オプション (778 ページ)
- SUBQUERY_FLATTENING_PREFERENCE オプション (779 ページ)
- SUBQUERY_PLACEMENT_PREFERENCE オプション (780 ページ)
- SUPPRESS_TDS_DEBUGGING オプション (781 ページ)
- SWEEPER_THREADS_PERCENT オプション (782 ページ)
- TDS_EMPTY_STRING_IS_NULL オプション [database] (783 ページ)
- TEMP_EXTRACT_APPEND オプション (783 ページ)
- TEMP_EXTRACT_BINARY オプション (784 ページ)
- TEMP_EXTRACT_COLUMN_DELIMITER オプション (785 ページ)
- TEMP_EXTRACT_DIRECTORY オプション (786 ページ)
- TEMP_EXTRACT_ESCAPE_QUOTES オプション (787 ページ)
- TEMP_EXTRACT_NAME_n オプション (788 ページ)
- TEMP_EXTRACT_NULL_AS_EMPTY オプション (790 ページ)
- TEMP_EXTRACT_NULL_AS_ZERO オプション (791 ページ)
- TEMP_EXTRACT_QUOTE オプション (792 ページ)
- TEMP_EXTRACT_QUOTES オプション (793 ページ)
- TEMP_EXTRACT_QUOTES_ALL オプション (794 ページ)
- TEMP_EXTRACT_ROW_DELIMITER オプション (795 ページ)
- TEMP_EXTRACT_SIZE_n オプション (796 ページ)
- TEMP_EXTRACT_SWAP オプション (797 ページ)
- TEMP_RESERVED_DBSPACE_MB オプション (798 ページ)
- TEMP_SPACE_LIMIT_CHECK オプション (799 ページ)
- TEXT_DELETE_METHOD オプション (800 ページ)
- TIME_FORMAT オプション (801 ページ)
- TIMESTAMP_FORMAT オプション (802 ページ)
- TOP_NSORT_CUTOFF_PAGES オプション (803 ページ)
- TRIM_PARTIAL_MBC オプション (804 ページ)
- TRUSTED_CERTIFICATES_FILE オプション (805 ページ)
- USER_RESOURCE_RESERVATION オプション (806 ページ)
- VERIFY_PASSWORD_FUNCTION オプション (807 ページ)
- WASH_AREA_BUFFERS_PERCENT オプション (809 ページ)
- WAIT_FOR_COMMIT オプション (810 ページ)

- `WD_DELETE_METHOD` オプション (811 ページ)

データ抽出オプション

データ抽出機能を使用すると、標準インタフェースから 1 つ以上のディスクファイルまたは指定パイプに **SELECT** 文の出力をリダイレクトすることによってデータベースからデータを抽出できます。

データ抽出機能の制御には、`TEMP_EXTRACT_...` データベースオプションが使用されます。

Transact-SQL 互換性オプション

Transact-SQL® 互換性オプションを使用することにより、SAP Sybase IQ の動作に Adaptive Server との互換性を持たせることができます。また旧版の動作と ISO SQL92 動作を両方可能にすることもできます。

Adaptive Server との互換性を強化するには、SAP Sybase IQ の **SET OPTION** 文ではなく、Transact-SQL の **SET** 文を使用することで、現在の接続の存続期間にわたってこれらのオプションの一部を設定できます。

参照：

- 一般的なデータベースオプション (592 ページ)
- Interactive SQL オプション (600 ページ)
- アルファベット順のオプションリスト (601 ページ)
- SET 文 [T-SQL] (536 ページ)
- `ALLOW_NULLS_BY_DEFAULT` オプション [TSQL] (603 ページ)
- `ANSI_CLOSE_CURSORS_ON_ROLLBACK` オプション [TSQL] (604 ページ)
- `ANSI_PERMISSIONS` オプション [TSQL] (605 ページ)
- `ANSINULL` オプション [TSQL] (606 ページ)
- `ANSI_SUBSTRING` オプション [TSQL] (607 ページ)
- `CHAINED` オプション [TSQL] (621 ページ)
- `CLOSE_ON_ENDTRANS` オプション [TSQL] (623 ページ)
- `CONTINUE_AFTER_RAISERROR` オプション [TSQL] (624 ページ)
- `CONVERSION_ERROR` オプション [TSQL] (625 ページ)
- `DIVIDE_BY_ZERO_ERROR` オプション [TSQL] (655 ページ)
- `NEAREST_CENTURY` オプション [TSQL] (730 ページ)
- `NON_KEYWORDS` オプション [TSQL] (733 ページ)
- `ON_ERROR` オプション [Interactive SQL] (736 ページ)

- ON_TSQL_ERROR オプション [TSQL] (737 ページ)
- QUOTED_IDENTIFIER オプション [TSQL] (759 ページ)
- SQL_FLAGGER_ERROR_LEVEL オプション [TSQL] (774 ページ)
- SQL_FLAGGER_WARNING_LEVEL オプション [TSQL] (775 ページ)
- STRING_RTRUNCATION オプション [TSQL] (776 ページ)
- TSQL_VARIABLES オプション [TSQL] (805 ページ)

Adaptive Server との互換性確保のための Transact-SQL オプションの設定

一部のオプションのデフォルト設定は、Adaptive Server のデフォルト設定と異なります。動作の互換性を保証したい場合は、各オプションを明示的に設定してください。

Open Client または JDBC インタフェースを使用して接続された場合、Adaptive Server との互換性の保持のために、現在の接続に対していくつかのオプションが明示的に設定されます。

表 11 : ASE 互換性のために明示的に設定される Transact-SQL オプション

オプション	ASE 互換の設定
ALLOW_NULLS_BY_DEFAULT	OFF
ANSINULL	OFF
CHAINED	OFF
CONTINUE_AFTER_RAISERROR	ON
DATE_FORMAT	YYYY-MM-DD
DATE_ORDER	MDY
ESCAPE_CHARACTER	OFF
ISOLATION_LEVEL	1
ON_TSQL_ERROR	CONDITIONAL
QUOTED_IDENTIFIER	OFF
TIME_FORMAT	HH:NN:SS.SSS
TIMESTAMP_FORMAT	YYYY-MM-DD HH:NN:SS.SSS
TSQL_VARIABLES	OFF

Interactive SQL オプション

Interactive SQL オプションは、データベースに対する Interactive SQL の対応を変更します。

構文 1

```
SET [ TEMPORARY ] OPTION
... [ userid. | PUBLIC. ] option-name = [ option-value ]
```

構文 2

```
SET PERMANENT
```

構文 3

```
SET
```

パラメータ

```
userid:
identifier, string or host-variable
```

```
option-name:
identifier, string or host-variable
```

```
option-value:
host-variable (indicator allowed), string, identifier,
or number
```

説明

TEMPORARY キーワードが含まれた構文 1 は、複合文の **BEGIN** キーワードと **END** キーワードの間で使用できません。

構文 2 の **SET PERMANENT** は、現在の Interactive SQL オプションのすべてを **SYSOPTIONS** システムテーブルに格納します。Interactive SQL が現在のユーザ ID で起動されるたびに、これらの値が自動的に設定されます。

構文 3 を使用すると、現在のオプション設定がすべて表示されます。Interactive SQL またはデータベースサーバに テンポラリオプションがある場合は、それが表示されます。それ以外の場合、永久オプション設定が表示されます。。

参照：

- 一般的なデータベースオプション (592 ページ)
- Transact-SQL 互換性オプション (598 ページ)
- アルファベット順のオプションリスト (601 ページ)
- **DEFAULT_ISQL_ENCODING** オプション [Interactive SQL] (648 ページ)

- ON_ERROR オプション [Interactive SQL] (736 ページ)

アルファベット順のオプションリスト

一般的なオプション、Transact-SQL 互換性オプション、Interactive SQL データベースオプションの説明。オプションのクラスを示す、角カッコで囲まれたインジケータがオプション名の後に付くこともあります。

データベースオプションのクラスインジケータは次のとおりです。

- [Interactive SQL] – データベースに対する Interactive SQL の対応を変更するオプション。
- [TSQL] – SAP Sybase IQ の動作に Adaptive Server との互換性を持たせたり、旧版の動作と ISO SQL92 の動作の両方を使用可能にするオプション。

参照：

- データベースオプションの概要 (585 ページ)
- 一般的なデータベースオプション (592 ページ)
- Transact-SQL 互換性オプション (598 ページ)
- Interactive SQL オプション (600 ページ)

AFFINITY_AUTOEXCLUDE_TIMEOUT オプション

SAP Sybase IQ がアフィニティマップから停止ノードを削除し、そのパーティションを他のノードに再割り当てするまでの時間。

指定可能な値

0 ~ 10080 分 (1 週間)

デフォルト

10 分

スコープ

オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。すぐに有効になります。

備考

SAP Sybase IQ がアフィニティマップから停止ノードを削除し、そのパーティションを他のノードに再割り当てするまでの時間。

AGGREGATION_PREFERENCE オプション

集合関数を処理する際に使用するアルゴリズムを選択します。

指定できる値

-6 ~ 6

デフォルト値

0

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

クエリ内の集合 (GROUP BY 関数、DISTINCT 関数、SET 関数) に関して、SAP Sybase IQ オプティマイザは、集合を処理するアルゴリズムをいくつかから選択することができます。AGGREGATION_PREFERENCE を使用すると、オプティマイザが選択する高コストのアルゴリズムの決定を上書きできます。このオプションは、クエリエンジン内でアルゴリズムが妥当であるかどうかを判定する内部ルールの上書きは行いません。

このオプションは通常、内部のテスト、および、オプティマイザがうまく処理できないクエリの手動チューニングに利用されます。経験豊富な DBA のみ使用してください。AGGREGATION_PREFERENCE オプションを設定すると、オプティマイザに対する変更が適切であると意味することになるため、設定が必要な場合は、SAP Sybase テクニカルサポートにご連絡ください。

値	アクション
0	オプティマイザの選択に従う。
1	ソートを使用する集合関数を優先する。
2	IQ のインデックスを使用する集合関数を優先する。
3	ハッシュを使用する集合関数を優先する。

値	アクション
4	個別／グループ化ソートを使用する集合を優先する。
5	グループ化カラムにハッシュ分割テーブルのすべての分割キーが含まれる場合、ソートを使用する集合を優先する。
6	グループ化カラムにハッシュ分割されたテーブルのすべての分割キーが含まれる場合、ハッシュを使用する集合を優先する。
-1	ソートを使用する集合関数を避ける。
-2	IQ のインデックスを使用する集合関数を避ける。
-3	ハッシュを使用する集合関数を避ける。
-4n	個別／グループ化ソートを使用する集合を避ける。
-5	グループ化カラムにハッシュ分割されたテーブルのすべての分割キーが含まれる場合、ソートを使用する集合を避ける。
-6	グループ化カラムにハッシュ分割されたテーブルのすべての分割キーが含まれる場合、ハッシュを使用する集合を避ける。

ALLOW_NULLS_BY_DEFAULT オプション [TSQL]

NULL か NOT NULL かを指定しないで作成された新規カラムに NULL 値を許可するかどうかを制御します。

指定できる値

ON、OFF

デフォルト値

ON

Open Client および JDBC 接続の場合は OFF

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

ALLOW_NULLS_BY_DEFAULT オプションは、Transact-SQL との互換性を維持するために組み込まれています。

ALLOW_SNAPSHOT_VERSIONING オプション

データベース内のすべてのベーステーブルに適用されます (RLV 対応テーブル専用とは対照的です)。すべてのベーステーブルのテーブルのバージョン管理を、テーブルレベルまたはローレベルのスナップショットバージョン管理のいずれかに限定します。このオプションは、IQ カタログストアには適用されません。

指定可能な値

any、table-level、row-level

- **any** – スナップショットバージョン管理に対する制限なし
- **row-level** – ローレベルのスナップショットバージョン管理のみを許可。テーブルレベルのバージョン管理を使用してテーブルの変更を試行するトランザクションはすべて、不正なスナップショットアイソレーションエラーで失敗する
- **table-level** – テ이블レベルのスナップショットバージョン管理のみを許可。ローレベルのバージョン管理を使用してテーブルの変更を試行するトランザクションはすべて、不正なスナップショットアイソレーションエラーで失敗する

デフォルト

any

スコープ

オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。すぐに有効になります。

ANSI_CLOSE_CURSORS_ON_ROLLBACK オプション [TSQL]

WITH HOLD 句で開いたカーソルを、**ROLLBACK** を実行するときに閉じるかどうかを制御します。

指定できる値

ON

デフォルト値

ON

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

ANSI の SQL/3 標準では、トランザクションをロールバックするときにすべてのカーソルが閉じている必要があります。このオプションを使用すると、強制的にこの動作に従うようになります。変更はできません。このオプションより、CLOSE_ON_ENDTRANS オプションが優先されます。

ANSI_PERMISSIONS オプション [TSQL]

DELETE 文と UPDATE 文のパーミッションのチェックを制御します。

指定できる値

ON、OFF

デフォルト値

ON

スコープ

オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。すぐに有効になります。

備考

ANSI_PERMISSIONS を ON にすると、DELETE 文と UPDATE 文に対する SQL92 パーミッションの要件がチェックされます。Adaptive Server のデフォルト値は OFF です。次の表にその違いを示します。

表 12 : ANSI_PERMISSIONS オプションの効果

SQL 文	ANSI_PERMISSIONS OFF 時に必要なパーミッション	ANSI_PERMISSIONS ON 時に必要なパーミッション
UPDATE	値が設定されているカラムでの UPDATE パーミッション	値が設定されているカラムでの UPDATE パーミッション WHERE 句に指定されたすべてのカラムでの SELECT パーミッション SET 句の右側に指定されたすべてのカラムでの SELECT パーミッション
DELETE	テーブルに対する DELETE パーミッション	テーブルに対する DELETE パーミッション WHERE 句で指定されるすべてのカラムに対する SELECT パーミッション

ANSINULL オプション [TSQL]

NULL 値のある = と != の解釈を制御します。

指定できる値

ON、OFF

デフォルト値

ON

Open Client および JDBC 接続の場合は OFF

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

ANSINULL ON の場合、'= ' または '!=' を使用した NULL との比較結果は不定になります。CASE などの他の処理による比較結果も対象に含まれます。

ANSINULL ON を OFF に設定すると、Adaptive Server との互換性保持のため、不定ではない結果を返す NULL との比較を使用できます。

注意：SQL Anywhere とは異なり、SAP Sybase IQ では、NULL 値が格納されたカラムに対する集合関数に対して警告「集合関数では、NULL 値は無視されません。」(SQLSTATE=01003) は生成されません。

ANSI_SUBSTRING オプション [TSQL]

start または length パラメータに負の値が指定されている場合に、SUBSTRING (SUBSTR) 関数の動作を制御します。

指定できる値
ON、OFF

デフォルト値
ON

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

ANSI_SUBSTRING オプションを ON に設定した場合、**SUBSTRING** 関数は ANSI/ISO SQL/2003 と同じ動作をします。開始オフセットが負またはゼロの場合は、文字列の左側に非文字が埋め込まれているものと解釈され、負の長さが指定された場合はエラーが発生します。

このオプションを OFF に設定した場合、**SUBSTRING** 関数の動作は以前のバージョンの SAP Sybase IQ と同じになります。負の開始オフセットは、文字列の末尾からのオフセットになり、負の長さは、開始オフセットから左に向かって指定の部分文字列の終端までの長さになります。開始オフセット 0 を使用した場合、開始オフセット 1 と同じ結果となります。

SUBSTRING 関数では、正以外の開始オフセットおよび負の長さを使用しないでください。可能な限り、**SUBSTRING** 関数の代わりに **LEFT** 関数または **RIGHT** 関数を使用してください。

例

次の例は、**ANSI_SUBSTRING** オプションの設定に応じて **SUBSTRING** 関数によって返される値の違いを示しています。

```
SUBSTRING( 'abcdefgh',-2,4 );
ansi_substring = Off ==> 'gh'
// substring starts at second-last character
ansi_substring = On ==> 'gh'
// takes the first 4 characters of
// ???abcdefgh and discards all ?
```

```
SUBSTRING( 'abcdefgh',4,-2 );
ansi_substring = Off ==> 'cd'
ansi_substring = On ==> value -2 out of range
for destination
```

```
SUBSTRING( 'abcdefgh',0,4 );
ansi_substring = Off ==> 'abcd'
ansi_substring = On ==> 'abcd'
```

ANSI_UPDATE_CONSTRAINTS オプション

許可する更新の範囲を制御します。

指定できる値

OFF、CURSORS、STRICT

デフォルト値

CURSORS

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

SAP Sybase IQ には、ANSI SQL 標準では許可されない更新を可能にするいくつかの拡張機能があります。これらの拡張機能を利用して、強力かつ効率的に更新を行うことができます。しかし、場合によっては、通常と異なる動作が起きることもあります。ユーザ側のアプリケーションがこれらの拡張機能の動作を想定していない場合、このような動作によって、更新が失われるなどの異常が発生することがあります。

ANSI_UPDATE_CONSTRAINTS は、SQL92 標準で許容される更新に限定するかどうかを制御します。

このオプションを **STRICT** に設定すると、次の更新が禁止されます。

- JOINS を含むカーソルの更新
- **ORDER BY** 句に指定されるカラムの更新
- **UPDATE** 文での **FROM** 句の使用

このオプションを **CURSORS** に設定した場合は、カーソルにだけ同じ制限が加えられます。カーソルが **FOR UPDATE** または **FOR READ ONLY** で開かれている場合を除き、データベースサーバは SQL92 標準に基づいて更新が許可されるかどうかを判断します。

ANSI_UPDATE_CONSTRAINTS が **CURSORS** または **STRICT** に設定されている場合、**ORDER BY** 句を含むカーソルのデフォルトは **FOR READ ONLY** になります。これ以外の場合は引き続き **FOR UPDATE** がデフォルトになります。

例

次のコードの結果は、**ANSI_UPDATE_CONSTRAINTS** の設定によって変わります。

```
CREATE TABLE mmg (a CHAR(3));
CREATE TABLE mmg1 (b CHAR(3));

INSERT INTO mmg VALUES ('001');
INSERT INTO mmg VALUES ('002');
INSERT INTO mmg VALUES ('003');
INSERT INTO mmg1 VALUES ('003');
SELECT * FROM mmg;
SELECT * FROM mmg1;
```

オプション 1: **ANSI_UPDATE_CONSTRAINTS** を **STRICT** に設定

```
SET OPTION public.Ansi_update_constraints = 'strict';
DELETE MMG FROM MMG1 WHERE A=B;
```

この更新操作は許可されていないというエラーが表示されます。

オプション 2: **ANSI_UPDATE_CONSTRAINTS** を **CURSORS** または **OFF** に設定

```
SET OPTION public.Ansi_update_constraints = 'CURSORS'; // or 'OFF'
DELETE mmg FROM mmg1 WHERE A=B;
```

データベースオプション

エラーが発生することなく削除が完了します。

参照：

- UPDATE 文 (566 ページ)

ALLOW_READ_CLIENT_FILE オプション

クライアント側データの転送を有効にします。

指定できる値

On、Off

デフォルト

Off

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY SECURITY OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

READ_CLIENT_FILE 関数を使用する場合など、クライアントコンピュータからファイルを読み込む場合は、このオプションを有効にしてください。

ASE_BINARY_DISPLAY オプション

SAP Sybase IQ バイナリカラムの表示を Adaptive Server バイナリカラムの表示と一致させるよう指定します。

指定できる値

ON、OFF

デフォルト値

OFF

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値にな

りますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

ASE_BINARY_DISPLAY は、**SELECT** 文の出力に影響します。

このオプションは IQ ストアのカラムのみに影響します。変数、カタログストアのカラム、または SQL Anywhere カラムには影響しません。このオプションが ON の場合、SAP Sybase IQ では、0x1234567890abcdef のように判読可能な ASCII 形式でカラムが表示されます。このオプションが OFF の場合、SAP Sybase IQ では、カラムが ASCII ではなくバイナリ出力として表示されます。

バイナリデータ型でバルクコピー操作をサポートするには、**ASE_BINARY_DISPLAY** を OFF に設定します。SAP Sybase IQ では、**LOAD TABLE USING CLIENT FILE** 文によるリモートデータのバルクロードがサポートされます。

参照：

- **LOAD TABLE** 文 (430 ページ)

ASE_FUNCTION_BEHAVIOR オプション

INTTOHEX や **HEXTOINT** などの SAP Sybase IQ の関数の出力が、Adaptive Server の関数の出力と一致するように指定します。

指定できる値

ON、OFF

デフォルト値

OFF

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

ASE_BEHAVIOR_FUNCTION が ON の場合、**HEXTOINT** や **INTTOHEX** などの SAP Sybase IQ データ型変換機能の一部が Adaptive Server の関数の出力と一致する出力を返すように指定します。ASE と SAP Sybase IQ で出力のフォーマットと長さに違いが存在するのは、ASE がデフォルトとして主に符号付き 32 ビットを使用し、SAP Sybase IQ がデフォルトとして主に符号なしの 64 ビットを使用するためです。

SAP Sybase IQ では 64 ビットの整数をサポートしていません。ASE には 64 ビットの整数データ型がありません。

例

この例では、**HEXTOINT** 関数は、**ASE_BEHAVIOR_FUNCTION** が ON と OFF のいずれであるかによって異なる値を返します。

HEXTOINT 関数は、**ASE_BEHAVIOR_FUNCTION** が OFF の場合 4294967287 を返します。

```
select hextoint('ffffffff7') from iq_dummy
```

HEXTOINT 関数は、**ASE_FUNCTION_BEHAVIOR** が ON の場合 -9 を返します。

```
select hextoint('ffffffff7') from iq_dummy
```

参照：

- **CONVERSION_ERROR** オプション [TSQL] (625 ページ)

AUDITING オプション [データベース]

データベース内の監査を有効または無効にします。

指定できる値

ON、OFF

デフォルト値

OFF

スコープ

オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SECURITY OPTION システム権限が必要です。すぐに有効になります。

備考

監査とは、データベース内の多くのイベントに関する詳細をトランザクションログに記録することです。監査は、パフォーマンスの多少の低下と引き換えに、いくつかのセキュリティ機能を実現します。データベースの監査をオンにすると、トランザクションログの使用は停止できません。トランザクションログをオフにする前に、監査をオフにする必要があります。監査がオンの場合、データベースは読み取り専用モードで起動できません。

AUDITING オプションを機能させるには、監査オプションを ON に設定し、**sa_enable_auditing_type** システムプロシージャを使用して、監査対象の情報の種類を指定する必要があります。これにはパーミッションチェックの組み合わせ、接続試行回数、DDL 文、パブリックオプション、トリガなどが含まれます。次に該当する場合は監査が行われません。

- **AUDITING** オプションが OFF に設定されている場合
- 監査オプションが無効の場合

AUDITING オプションを ON に設定し、監査オプションを指定しない場合、すべての種類の監査情報が記録されます。

BASE TABLES IN RLV STORE オプション

新しく作成されたテーブルを RLV ストアに登録し、ローレベルのバージョン管理を有効にします。RLV 対応のテーブルは、複数のライタの同時アクセスが可能になります。この設定は、**CREATE_TABLE** 文を使用してテーブルレベルで無効にできます。

指定可能な値

ON、OFF

デフォルト

OFF

スコープ

オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。すぐに有効になります。

備考

ON に設定されていると、新しく登録されたテーブルは RLV ストアに登録されます。RLV 対応のテーブルは、リアルタイム更新の対象として最適化されます。

CREATE_TABLE 文の { ENABLE | DISABLE } RLV STORE 句は、常に **BASE_TABLES_IN_RLV_STORE** オプションより優先されます。

Base_Tables_in_RLV_STORE option が有効になると、新しく作成された IQ ベース テーブルは自動的に RLV 対応になります。このオプションを有効にしても、既存の IQ ベーステーブルには影響しません。

BIT_VECTOR_PINNABLE_CACHE_PERCENT オプション

永続的ビットベクトルオブジェクトが維持できる、ユーザの一時メモリの最大パーセンテージを制御します。

指定できる値
0 - 100

デフォルト値
40

スコープ
オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

BIT_VECTOR_PINNABLE_CACHE_PERCENT は、任意の永続的ビットベクトルオブジェクトをメモリ内に留めることができる、ユーザの一時メモリ割り付けのパーセンテージを制御します。デフォルトは 40% です。通常は変更しないでください。

このオプションは、主に テクニカルサポート で使用されます。

BIT_VECTOR_PINNABLE_CACHE_PERCENT の値を変更する場合は、細心の注意が必要です。事前にさまざまなクエリへの影響を分析してください。

参照：

- HASH_PINNABLE_CACHE_PERCENT オプション (679 ページ)
- SORT_PINNABLE_CACHE_PERCENT オプション (773 ページ)

BLOCKING オプション

ロック競合に対する動作を制御します。**BLOCKING** は、マルチプレックスのセカンダリノードではサポートされません。

指定できる値

ON、OFF

デフォルト値

OFF

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

BLOCKING が OFF の場合、トランザクションが書き込み処理を試みて、それが別のトランザクションの読み込みロックによってブロックされた場合に、トランザクションにエラーが返されます。

BLOCKING が ON の場合、別のトランザクションが保持する既存のロックと競合するロックを取得しようとした場合、競合するすべてのロックが解放されるか、blocking_timeout に達するまで、トランザクションは待機します。blocking_timeout ミリ秒以内にロックが解放されない場合は、待機しているトランザクションにエラーが返されます。

参照：

- **BLOCKING_TIMEOUT** オプション (616 ページ)

BLOCKING_TIMEOUT オプション

トランザクションがロックを獲得するまでの待機時間を制御します。
BLOCKING_TIMEOUT は、マルチプレックスのセカンダリノードではサポートされません。

指定可能な値
整数値 (ミリ秒単位)

デフォルト
0

スコープ
オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考
ブロッキングオプションを ON に設定すると、トランザクションが取得しようとしたロックが既存のロックと競合している場合は、設定された時間 (ミリ秒単位) が経過するまで、競合するロックが解放されるのを待ちます。blocking_timeout ミリ秒以内にロックが解放されない場合は、待機しているトランザクションにエラーが返されます。

このオプションを 0 に設定すると、ロックを獲得しようとしているトランザクションはすべて、競合するトランザクションがロックを解放するまで待機します。

参照：

- BLOCKING オプション (615 ページ)

BT_PREFETCH_MAX_MISS オプション

SAP Sybase IQ が特定のクエリの B ツリーページのプリフェッチを継続するかどうかを判断する方法を制御します。

指定できる値
0 - 1000

デフォルト値

2

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

SAP Sybase テクニカルサポートから指示があった場合にのみ使用してください。HG (High_Group) インデックスを使用するクエリでは、SAP Sybase IQ は、プリフェッチする必要がもうないと判断するまで B ツリーページを順にプリフェッチします。一部のクエリでは、プリフェッチを早く終了することがあります。**BT_PREFETCH_MAX_MISS** の値を大きくすると、SAP Sybase IQ はより長くプリフェッチを継続しますが、I/O が不必要に増大する可能性があります。

HG を使用するクエリの実行速度が予想より遅い場合は、**BT_PREFETCH_MAX_MISS** の値を徐々に増やしてみます。

最高のパフォーマンスを得るためには、いくつかの設定を試してみる必要があります。大部分のクエリには、1 から 10 までの範囲で値を指定します。

参照：

- BT_PREFETCH_SIZE オプション (617 ページ)
- PREFETCH_BUFFER_LIMIT オプション (744 ページ)

BT_PREFETCH_SIZE オプション

High_Group B ツリーの先読みバッファのサイズを制限します。

指定できる値

0 - 100.0 に設定すると、B ツリーのプリフェッチが無効になります。

デフォルト値

10

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

B ツリーのプリフェッチは、INSERT、大規模な DELETE、範囲述部、DBCC (データベース一貫性チェッカ) コマンドなどの High_Group インデックスへのすべてのシーケンシャルアクセスにおいてデフォルトで有効になります。

BT_PREFETCH_SIZE は B ツリーページの先読みバッファのサイズを制限します。プリフェッチのサイズを小さくしていくとバッファの空きが増えますが、ある時点でパフォーマンスが低下します。プリフェッチのサイズを大きくしていくと、これ以上増やしても効果が上がらない値がわかる場合があります。このオプションは、ユニークでない High_Group インデックスに対して

PREFETCH_GARRAY_PERCENT、**GARRAY_INSERT_PREFETCH_SIZE**、および **GARRAY_RO_PREFETCH_SIZE** オプションとともに使用します。

参照：

- **GARRAY_INSERT_PREFETCH_SIZE** オプション (677 ページ)
- **GARRAY_RO_PREFETCH_SIZE** オプション (679 ページ)
- **PREFETCH_GARRAY_PERCENT** オプション (745 ページ)

BTREE_PAGE_SPLIT_PAD_PERCENT オプション

1 ページあたりのフィルファクタを B ツリー構造のページ分割時に決定します。

指定できる値

0–90

デフォルト値

50

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値にな

りますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

B ツリー構造は、HG、LF、DT、TIME、および DTTM インデックスによって使用されます。B ツリーページを分割すると、新しいキーがインデックスに挿入される際の分割を回避するため、指定された割合が空白になります。

インデックスは、ページレベルで記憶領域を予約します。この記憶領域は、追加データとして新しいキーが挿入されたときに割り当てることができます。領域を確保するには追加のディスク領域が必要となりますが、インクリメンタルな挿入のパフォーマンスが向上します。インクリメンタルな挿入を行う予定があり、それによってできる新しいローの値が以前のインデックスにまだ存在しない場合は、GARRAY_PAGE_SPLIT_PAD_PERCENT に 0 以外の値を設定すると、インクリメンタルな挿入処理のパフォーマンスが向上する可能性があります。

インデックスをインクリメンタルに更新する予定がない場合は、このオプションの値を小さくして、ディスク領域を節約できます。

参照：

- GARRAY_FILL_FACTOR_PERCENT オプション (676 ページ)
- GARRAY_PAGE_SPLIT_PAD_PERCENT オプション (678 ページ)

CACHE_AFFINITY_PERCENT オプション

アフィニティに使用するメインバッファキャッシュの最大割合。存在するアフィニティデータが不十分である場合は、非アフィニティデータがこの領域を使用できます。

指定可能な値

0 - 100 %

デフォルト

70

スコープ

オプションは、データベース (PUBLIC) レベルでのみ設定できます。

データベースオプション

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。変更を有効にするには、データベースサーバを停止してから再起動する必要があります。

備考

このオプションは、アフィニティ化データバッファに使用されるバッファキャッシュの割合を定義します。SAP Sybase IQ バッファキャッシュは長い MRU/LRU チェーンとして構成されます。非アフィニティ化データバッファは、この割合が 0 以外の場合にアフィニティ化バッファの後ろのチェーンに入れられるため、そのアフィニティ化データは非アフィニティ化データよりも長くキャッシュ内に留まります。アフィニティ化データバッファではこの割合全部を埋めることができない場合は、非アフィニティ化データで残りを埋めることができます。

注意： このオプションを変更する前に、**WASH_AREA_BUFFERS_PERCENT** オプションの値を確認してください。**WASH_AREA_BUFFERS_PERCENT** はバッファキャッシュの LRU 側に影響し、**CACHE_AFFINITY_PERCENT** は MRU 側に影響します。これら 2 つの値の合計が 100% を超えないようにする必要があります。

CACHE PARTITIONS オプション

メインおよびテンポラリバッファキャッシュに使用するパーティションの数を設定します。

指定できる値

0、1、2、4、8、16、32、64

- **0** - (デフォルト) SAP Sybase IQ は、パーティション数を `number_of_cpus/8` として自動的に計算し、結果を 2 の累乗に丸めます。最大数は 64 です。
- **1** - パーティションは 1 つのみ。この値では分割が無効化されます。
- **2 ~ 64** - パーティションの数。2 の累乗の値を指定する必要があります。

デフォルト

0

スコープ

オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。次回、データベースサーバを起動したときに現在のデータベースで有効になります。

備考

バッファキャッシュの分割は、複数の CPU を搭載したシステムでロック競合を減らし、パフォーマンスを向上させることがあります。通常は、システム上の CPU

の数を基に SAP Sybase IQ が自動計算した値を使用します。ただし、マルチ CPU 環境でクエリまたはロードのパフォーマンスが予想より悪い場合は、**CACHE_PARTITIONS** の値を変更するとパフォーマンスが向上することがあります。

理想的なパーティションの数は、CPU の数とプラットフォームの両方で決まります。環境に最も適した設定を判断するには、いくつかの値で試してみる必要があります。

CACHE_PARTITIONS に設定した値は、メインとテンポラリの両方のバッファ キャッシュに適用されます。各バッファキャッシュにおいて、パーティションの絶対最大数は 64 です。

-iqpartition start_iq サーバオプションでは、サーバレベルでパーティションの制限を設定します。サーバの起動時に **-iqpartition** を指定すると、この設定が **CACHE_PARTITIONS** の設定よりも優先されます。

パーティションの数は、その他のバッファキャッシュの設定には影響しません。また、IQ モニタが収集する統計情報にも影響しません。すべてのパーティションの統計情報は、合計され、1つの値として報告されます。

例

100 の CPU が搭載されたシステムで、**CACHE_PARTITIONS** を設定しなかった場合、SAP Sybase IQ はパーティションの数を自動的に 16 に設定します。

$100 \text{ cpus} / 8 = 12$ なので、丸めると 16 になります。

この設定では、メインバッファキャッシュのパーティション数とテンポラリー キャッシュのパーティション数がそれぞれ 16 になります。

100 の CPU を搭載した同じシステムで、パーティション数を明示的に 8 に設定するには、次のように指定します。

```
SET OPTION "PUBLIC".CACHE_PARTITIONS=8
```

CHAINED オプション [TSQL]

BEGIN TRANSACTION 文のないときにトランザクションモードを制御します。

指定できる値

ON、OFF

デフォルト値

ON

Open Client および JDBC 接続の場合は OFF

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

Transact-SQL トランザクションモードを制御します。非連鎖モード (**CHAINED = OFF**) では、明示的な **BEGIN TRANSACTION** 文でトランザクションが開始された場合を除いて、各文が個別にコミットされます。連鎖モード (**CHAINED = ON**) では、データ取得文または変更文の前にトランザクションが暗黙的に開始されます。Adaptive Server では、デフォルトの設定は OFF です。

CHECKPOINT_TIME オプション

チェックポイントを使用しないでデータベースサーバを実行する最大時間を分単位で設定します。

指定できる値
整数値

デフォルト値
60

スコープ

オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。変更を有効にするには、データベースサーバを停止し、再起動します。

説明

このオプションを **RECOVERY_TIME** とともに使用して、チェックポイントと実行タイミングを決定します。

参照：

- RECOVERY_TIME オプション (760 ページ)

CIS_ROWSET_SIZE オプション

各フェッチに対してリモートサーバから返されるローの数を設定します。

指定できる値
整数

デフォルト値
50

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

このオプションでは、ODBC を使用してリモートデータベースサーバに接続しているときの ODBC FetchArraySize 値を設定します。

CLOSE_ON_ENDTRANS オプション [TSQL]

トランザクションの終了時にカーソルを閉じるかどうかを制御します。

指定できる値
ON

デフォルト値
ON

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

データベースオプション

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

CLOSE_ON_ENDTRANS が ON (デフォルトであり、許可される唯一の値) の場合、カーソルはトランザクションが終了すると閉じられます。これは Transact-SQL と互換性のある動作です。

CONTINUE_AFTER_RAISERROR オプション [TSQL]

RAISERROR 文に応じて動作を制御します。

指定できる値

ON、OFF

デフォルト値

ON

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

RAISERROR 文は、プロシージャ内でエラーを発生させるために使用します。

CONTINUE_AFTER_RAISERROR が OFF の場合は、**RAISERROR** 文が検出されると、プロシージャの実行が停止されます。

CONTINUE_AFTER_RAISERROR が ON の場合は、**RAISERROR** 文は実行終了エラーを通知しなくなります。代わりに、**RAISERROR** ステータスコードとメッセージが格納され、プロシージャが完了すると直前の **RAISERROR** が返されます。

RAISERROR を発生させたプロシージャが別のプロシージャに呼び出されたものである場合、最も外側のプロシージャが終了するまで **RAISERROR** は返されません。

中間 **RAISERROR** のステータスとコードは、プロシージャが終了すると失われます。リターン時に **RAISERROR** 以外のエラーが発生した場合は、発生したエラー情報が返され、**RAISERROR** 情報は失われます。アプリケーションでは、別の実行ポイントでグローバル変数 @@error を検査して、中間 **RAISERROR** ステータスを問い合わせることができます。

CONTINUE_AFTER_RAISERROR の設定によって **RAISERROR** 文の後の動作が制御されるのは、**ON_TSQL_ERROR** オプションが **CONDITIONAL** (デフォルト) に設定されている場合に限られます。**ON_TSQL_ERROR** オプションが **STOP** または **CONTINUE** に設定されている場合、**ON_TSQL_ERROR** の設定が **CONTINUE_AFTER_RAISERROR** の設定よりも優先されます。

参照：

- **ON_TSQL_ERROR** オプション [TSQL] (737 ページ)

CONVERSION_ERROR オプション [TSQL]

データベースからデータをフェッチするときの、データ型変換障害のレポートを制御します。

指定できる値

ON、OFF

デフォルト値

ON

スコープ

オプションは、データベース (**PUBLIC**) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの **PUBLIC** 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、**SET ANY PUBLIC OPTION** システム権限が必要です。個々の接続または **PUBLIC** ロールに一時的に設定できます。すぐに有効になります。

備考

このオプションでは、データベースからのデータのフェッチ時またはデータベースへのデータの挿入時のデータ型変換障害を、エラー (**CONVERSION_ERROR** が ON) と警告 (**CONVERSION_ERROR** が OFF) のどちらとしてデータベースに通知させるかを制御します。

CONVERSION_ERROR を ON に設定すると、`SQL_CONVERT_ERROR` エラーが生成されます。

オプションを OFF に設定すると、警告 `SQL_CANT_CONVERT` が生成されます。**LOAD** 文でデータ変換を実行する各スレッドは、最大で 1 件の警告メッセージを `.iqmsg` ファイルに書き込むだけです。

変換エラーが警告のみでレポートされた場合、変換できなかった値の代わりに NULL 値が使用されます。Embedded SQL では、インジケータ変数はエラーを出したカラムに -2 を設定します。

CONVERSION_MODE オプション

さまざまな操作で、バイナリデータ型 (BINARY、VARBINARY、および LONG BINARY) と他の非バイナリデータ型 (BIT、TINYINT、SMALLINT、INT、UNSIGNED INT、BIGINT、UNSIGNED BIGINT、CHAR、VARCHAR および LONG VARCHAR) 間の暗黙的な変換を制限します。

指定できる値

0, 1

デフォルト値

0

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、`SET ANY PUBLIC OPTION` システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

デフォルト値 0 は、12.7 より前のバージョンの暗黙の変換動作を保持します。

CONVERSION_MODE を 1 に設定すると、**INSERT**、**UPDATE**、およびクエリ内でのバイナリデータ型から他の非バイナリデータ型への暗黙的な変換が制限されます。バイナリ変換制限モードは、**LOAD TABLE** のデフォルト値と **CHECK** 制約にも適用されます。**CONVERSION_MODE 1** を設定することで、実質的に意味のない操作となる暗号化データの暗黙のデータ型変換を防止できます。

SAP Sybase IQ Advanced Security オプションの暗号化カラム機能を使用するには、そのためのライセンスを取得している必要があります。

暗黙的な変換の制限

CONVERSION_MODE オプションのバイナリ変換制限モード値が 1 (**CONVERSION_MODE** = 1) の場合、次の操作の暗黙的な変換が制限されます。

- CHECK 制約またはデフォルト値を指定した **LOAD TABLE**
- **INSERT...SELECT**、**INSERT...VALUE**、**INSERT...LOCATION**
- 特定の種類の **UPDATE**
- 更新可能カーソルを介する、特定の種類の **INSERT** と **UPDATE**
- 通常、クエリのすべての側面

LOAD TABLE に対する暗黙的なバイナリ変換制限モード

暗黙的なバイナリ変換制限モード (**CONVERSION_MODE** を 1 に設定) は、**CHECK** 制約またはデフォルト値を指定した **LOAD TABLE** に適用されます。

例

```
CREATE TABLE t3 (c1 INT,
  csi SMALLINT,
  cvb VARBINARY(2),
  CHECK (csi<cvb));
SET TEMPORARY OPTION CONVERSION_MODE = 1;
```

次の要求は、

```
LOAD TABLE t3(c1 ',', csi ',', cvb ',')
FROM '/s1/mydata/t3.inp'
QUOTES OFF ESCAPES OFF
ROW DELIMITED BY '¥n'
```

次のメッセージで失敗します。

```
"Invalid data type comparison in predicate
(t3.csi < t3.cvb), [-1001013] ['QFA13']"
```

INSERT に対する暗黙的なバイナリ変換制限モード

暗黙的なバイナリ変換制限モード (**CONVERSION_MODE** を 1 に設定) は、**INSERT...SELECT**、**INSERT...VALUE**、および **INSERT...LOCATION** に適用されます。

例

```
CREATE TABLE t1 (c1 INT PRIMARY KEY,
  cbt BIT NULL,
  cti TINYINT,
  csi SMALLINT,
  cin INTEGER,
  cui UNSIGNED INTEGER,
  cbi BIGINT,
```

```
cub UNSIGNED BIGINT,  
cch CHAR(10),  
cvc VARCHAR(10),  
cbn BINARY(8),  
cvb VARBINARY(8),  
clb LONG BINARY,  
clc LONG VARCHAR);  
  
CREATE TABLE t2 (c1 INT PRIMARY KEY,  
cbt BIT NULL,  
cti TINYINT,  
csi SMALLINT,  
cin INTEGER,  
cui UNSIGNED INTEGER,  
cbi BIGINT,  
cub UNSIGNED BIGINT,  
cch CHAR(10),  
cvc VARCHAR(10),  
cbn BINARY(8),  
cvb VARBINARY(8),  
clb LONG BINARY,  
clc LONG VARCHAR);  
  
CREATE TABLE t4 (c1 INT, cin INT DEFAULT 0x31);  
  
SET TEMPORARY OPTION CONVERSION_MODE = 1;
```

次の要求は、

```
INSERT INTO t1(c1, cvb) SELECT 99, cin FROM T2  
WHERE c1=1
```

次のメッセージで失敗します。

```
"Unable to convert column 'cvb' to the requested  
datatype (varbinary) from datatype (integer).  
[-1013043] ['QCA43']"
```

UPDATE に対する暗黙的なバイナリ変換制限モード

暗黙的なバイナリ変換制限モード (CONVERSION_MODE を 1 に設定) は、特定の種類の **UPDATE** に適用されます。

暗黙的なバイナリ変換制限モードが次に適用されます。

- **UPDATE SET VALUE FROM** *expression* (定数の場合も含まれます)
- **UPDATE SET VALUE FROM** *other column*
- **UPDATE SET VALUE FROM** *host variable*
- **JOIN UPDATE SET VALUE FROM** *column of other table*

例

次の要求は、

```
UPDATE t1 SET cbi=cbn WHERE c1=1
```


次のメッセージで失敗します。

```
"Unable to implicitly convert column 'cbi' to datatype
(bigint) from datatype (binary). [-1000187] ['QCB87']"
```

更新可能カーソルによる位置付け INSERT と位置付け UPDATE に対する暗黙的なバイナリ変換制限モード

暗黙的なバイナリ変換制限モード (CONVERSION_MODE を 1 に設定) は、更新可能カーソルによる特定の種類の INSERT と UPDATE に適用されます。

暗黙的なバイナリ変換制限モードが次に適用されます。

- **PUT cursor-name USING ... host-variable**
- 別のカラムからの位置付け **UPDATE**
- 定数からの位置付け **UPDATE**
- ホスト変数からの位置付け **UPDATE**

クエリに対する暗黙的なバイナリ変換制限モード

暗黙的なバイナリ変換制限モード (CONVERSION_MODE を 1 に設定) は、通常、クエリのあらゆる面に適用されます。

比較演算子

CONVERSION_MODE = の場合、次の演算子に制限が適用されます。

- =, !=, <, <=, >=, <>, !>, !<
- BETWEEN ... AND
- IN

ただし、制限が適用されるのは、これらの演算子が次の句の検索条件で使用されているときです。

- WHERE 句
- HAVING 句
- CHECK 句
- ジョインでの ON フレーズ
- IF CASE 式

例

次のクエリは、

```
SELECT COUNT(*) FROM T1
WHERE cvb IN (SELECT csi FROM T2)
```

次のメッセージで失敗します。

```
"Invalid data type comparison in predicate  
(t1.cvb IN (SELECT t1.csi ...)), [-1001013]  
['QFA13']"
```

文字列関数

CONVERSION_MODE = 1 の場合、次の文字列関数に制限が適用されます。

- **CHAR**
- **CHAR_LENGTH**
- **DIFFERENCE**
- **LCASE**
- **LEFT**
- **LOWER**
- **LTRIM**
- **PATINDEX**
- **RIGHT**
- **RTRIM**
- **SIMILAR**
- **SORTKEY**
- **SOUNDEX**
- **SPACE**
- **STR**
- **TRIM**
- **UCASE**
- **UPPER**

例

次のクエリは、

```
SELECT ASCII(cvb) FROM t1 WHERE c1=1
```

次のメッセージで失敗します。

```
"Data exception - data type conversion is not  
possible. Argument to ASCII must be string,  
[-1009145] ['QFA2E']"
```

次の関数には、文字列引数またはバイナリ引数のいずれかを指定できます。
CONVERSION_MODE = 1 の場合、複数のデータ型が混在する引数 (1 つが文字列で、
もう 1 つがバイナリ) に制限が適用されます。

- **INSERTSTR**
- **LOCATE**
- **REPLACE**
- **STRING**

- **STUFF**

例

次のクエリは、

```
SELECT STRING(cvb, cvc) FROM t1 WHERE c1=1
```

(ここで、カラム cvb は VARBINARY として、カラム cvc は VARCHAR として定義されています)

次のメッセージで失敗します。

```
"Data exception - data type conversion is not possible. Arguments to STRING must be all binary or all string, [-1009145] ['QFA2E']"
```

次の文字列関数には制限は適用されません。

- **BIT_LENGTH**
- **BYTE_LENGTH**
- **CHARINDEX**
- **LENGTH**
- **OCTET_LENGTH**
- **REPEAT**
- **REPLICATE**
- **SUBSTRING**

算術演算と関数

CONVERSION_MODE=1 の場合、算術演算で使用されている次の演算子に制限が適用されます。

+, -, *, /

制限はビット処理式で次のビット処理演算子に適用されます。

& (AND)、| (OR)、^ (XOR)

次の関数の整数引数にも制限が適用されます。

- **ROUND**
- **"TRUNCATE"**
- **TRUNCNUM**

例

次のクエリは、

```
SELECT ROUND(4.4, cvb) FROM t1 WHERE C1=1
```

次のメッセージで失敗します。

データベースオプション

```
"Data exception - data type conversion is not possible. Second Argument to ROUND cannot be converted into an integer, [-1009145] ['QFA2E']"
```

さまざまな関数に対する整数引数

CONVERSION_MODE = 1 の場合、次の関数の整数引数に制限が適用されます。

- **ARGN**
- **SUBSTRING**
- **DATEADD**
- **YMD**

例

次のクエリは、

```
SELECT ARGN(cvb, csi, cti) FROM t1 WHERE c1=1
```

次のメッセージで失敗します。

```
"Data exception - data type conversion is not possible. First Argument to ARGN cannot be converted to an integer, [-1009145] ['QFA2E']"
```

統計関数、集合関数、数値関数

CONVERSION_MODE = 1 の場合、引数として数値式を要求する統計関数、集合関数、数値関数には、それ以上制限は適用されません。

CONVERT_VARCHAR_TO_1242 オプション

バージョン 12.4.2 より前の VARCHAR データを圧縮フォーマットに変換します。

指定できる値

ON、OFF

デフォルト値

OFF

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。sp_iqcheckdb を (あらゆるモードで) 実行したときに有効になります。

備考

データベースのデータ圧縮率を高めます。特に可変長文字列が多いデータベースに対して有効です。

このオプションを設定後、バージョン 12.4.2 より前に作成された VARCHAR カラムに対して一度だけ **sp_iqcheckdb** を実行してください。

COOPERATIVE_COMMIT_TIMEOUT オプション

トランザクションログ内の **COMMIT** エントリをディスクに書き込むタイミングを制御します。

指定できる値

整数値 (ミリ秒単位)

デフォルト値

250

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、**SET ANY PUBLIC OPTION** システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

このオプションは、**COOPERATIVE_COMMITS** が ON に設定されているときのみ有効です。データベースサーバは、他の接続がログページに書き込むのを、指定したミリ秒数だけ待ってからディスクへの書き込みを実行します。デフォルト設定は 250 ミリ秒です。

参照：

- COOPERATIVE_COMMITS オプション (634 ページ)

COOPERATIVE_COMMITS オプション

コミットされた情報がディスクに書き込まれるタイミングを制御します。

指定できる値

ON、OFF

デフォルト値

ON

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

COOPERATIVE_COMMITS が OFF に設定されている場合、データベースサーバは **COMMIT** された情報を受信するとすぐにその情報をディスクに書き込みます。その後、アプリケーションは続行を許可されます。

CREATE_HG_AND_FORCE_PHYSICAL_DELETE が ON (デフォルト) の場合、データベースサーバは **COMMIT** された情報をディスクにすぐには書き込みません。アプリケーションは、**COOPERATIVE_COMMIT_TIMEOUT** オプションで設定された最大時間が経過するまで、他の処理によるページへの入力を待つ必要があります。この時間の経過後にコミット内容がディスクに書き込まれます。

CREATE_HG_AND_FORCE_PHYSICAL_DELETE を ON に設定し、**COOPERATIVE_COMMIT_TIMEOUT** の設定値を大きくすると、ディスク I/O の数が減少するため、データベースサーバの全体的なスループットが向上しますが、その代わりに、個々の接続のターンアラウンドタイムは長くなります。

参照：

- **COOPERATIVE_COMMIT_TIMEOUT** オプション (633 ページ)

CREATE_HG_WITH_EXACT_DISTINCTS

データベースエンジンで多層 HG インデックスと単層 HG インデックスのどちらを作成するかを決定します。

指定可能な値

ON、OFF

デフォルト

OFF

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

CREATE_HG_WITH_EXACT_DISTINCTS は、データベースエンジンで HG インデックスを単層 HG と多層 HG のどちらとして作成するかを決定します。

- **CREATE_HG_WITH_EXACT_DISTINCTS='ON'** の場合、後続の HG インデックス (**CREATE INDEX** コマンドを使用して明示的に作成されるものと、PRIMARY KEY または FOREIGN KEY 宣言に基づくテーブルの暗黙的な作成または変更によって作成されるもの) は、すべて単層 HG インデックスになります。
- **CREATE_HG_WITH_EXACT_DISTINCTS='OFF'** の場合、後続の HG インデックス (**CREATE INDEX** コマンドを使用して明示的に作成されるものと、PRIMARY KEY または FOREIGN KEY 宣言に基づくテーブルの暗黙的な作成または変更によって作成されるもの) は、すべて多層 HG インデックスになります。

このオプションは、新規に作成されたすべての 16.0 データベースと、SAP Sybase IQ 15.x からアップグレードされたすべての 16.0 データベースで、デフォルトで ON に設定されます。新しい多層構造を利用するには、このオプションを OFF に設定します。単層 HG インデックスから多層 HG への変換 (およびこの逆の変換) には、**sp_iqrebuildindex** を使用します。

CREATE_HG_AND_FORCE_PHYSICAL_DELETE

16.0 の多層 HG インデックス削除動作を制御します。

指定可能な値

ON/OFF

デフォルト

ON

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

CREATE_HG_AND_FORCE_PHYSICAL_DELETE は、削除操作での HG からのローの物理的な削除を直ちに実行するか、それともロードでの後の時点まで遅らせて実行するかを決定します。

- **CREATE_HG_AND_FORCE_PHYSICAL_DELETE='ON'** (デフォルト) に設定すると、SAP Sybase IQ は物理的な削除を実行します。この設定では、一部のクエリ (.. のリンクリストや順序付きの射影など) のパフォーマンスが向上しますが、多層 HG インデックスを持つテーブルに対する削除クエリの処理が遅くなる場合があります。
- **CREATE_HG_AND_FORCE_PHYSICAL_DELETE='OFF'** に設定すると、SAP Sybase IQ は仮想または遅延削除を実行します。この設定は、多層 HG インデックスを持つテーブルに対するクエリに影響することがあります。

CREATE_HG_AND_FORCE_PHYSICAL_DELETE は、多層 HG カラムインデックスの作成前に設定します。既存の HG インデックスには作用しません。

sp_iqrebuildindex へは影響しません。このオプションは多層 HG インデックスの存続期間中永続し、変更はできません。変更するには、そのインデックスを削除し、このオプションの設定を切り替えてから再作成する必要があります (**sp_iqrebuildindex** でインデックスのステータスを変更することはできません)。

注意： `sp_iqrebuildindex` の出力には、このオプションのステータスを示す `Force Physical Delete` カラムが含まれます。

CURSOR_WINDOW_ROWS オプション

バッファするカーソルローの数を定義します。

指定できる値
20 - 100000

デフォルト値
200

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、`SET ANY PUBLIC OPTION` システム権限が必要です。個々の接続または PUBLIC ロールに対してのみ設定できます。変更を有効にするには、データベースサーバを停止し、再起動します。

備考

アプリケーションがカーソルを開くと、SAP Sybase IQ は、クエリによって生成されたデータを保持する FIFO (先入れ先出し) バッファを作成します。

CURSOR_WINDOW_ROWS では、バッファに保持できるロー数を定義します。カーソルが `NO SCROLL` 以外のモードで開かれた場合、SAP Sybase IQ では、バッファに保持できるローの総数まで後方スクロールを行えます。この数を超すローまで後方スクロールするには、クエリを再開する必要があります。`NO SCROLL` カーソルの場合は後方スクロールが許可されていないため、上記は該当しません。

たとえば、このオプションのデフォルト値のままの場合、最初はクエリ結果セットのロー 1 から 200 までがバッファに保持されます。先頭の 300 ローをフェッチした場合、バッファにはロー 101 から 300 までが保持されます。このバッファ内では、ごくわずかなオーバーヘッドコストで前方または後方へのスクロールができます。ロー 101 より前にスクロールすると、SAP Sybase IQ はそのクエリを再開し、必要なローがバッファに含まれるようにします。これは高コストの処理なので、アプリケーションではできるだけ避ける必要があります。

CURSOR_WINDOW_ROWS の値を大きくしてスクロールエリアを広げるという方法

データベースオプション

もありますが、ほとんどのアプリケーションではデフォルト設定の 200 で十分です。

DATE_FIRST_DAY_OF_WEEK オプション

1 週間の始まりを何曜日にするかを指定します。

指定できる値
0-6

デフォルト値
0 (日曜日)

スコープ

PUBLIC または他のユーザ／ロールに対してこのオプションを設定するには SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに対して設定できます。

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに対して設定できます。変更を有効にするには、データベースサーバを停止し、再起動します。

備考

1 週間の始まりの曜日をこのオプションを使用して指定します。デフォルトでは、日曜日が第 1 日、月曜日が第 2 日、火曜日が第 3 日、のようになります。

表 13 : DATE_FIRST_DAY_OF_WEEK の有効な値

値	第 1 日
0	日曜日
1	月曜日
2	火曜日
3	水曜日
4	木曜日

値	第 1 日
5	金曜日
6	土曜日

たとえば、**DATE_FIRST_DAY_OF_WEEK** の値を 3 に変更すると、水曜日が第 1 日、木曜日が第 2 日... というようになります。このオプションの影響を受けるのは、**DOW** 関数と **DATEPART** 関数のみです。

SQL Anywhere のオプション **FIRST_DAY_OF_WEEK** ではこれと同じ機能が実行されますが、0 から 6 ではなく 1 から 7 の値が割り当てられます。1 は月曜日、7 は日曜日 (デフォルト) を表します。

DATE_FORMAT オプション

データベースから取得した日付のフォーマットを設定します。

指定できる値
文字列

デフォルト値
'YYYY-MM-DD'。これは ISO 日付フォーマット仕様準拠です。

スコープ
オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考
フォーマットは次の記号を組み合わせた文字列です。

表 14 : DATE_FORMAT 文字列で使用される記号

記号	説明
yy	西暦年 2 桁。

記号	説明
yyyy	西暦年 4 桁。
mm	2 桁の月、またはコロンの後の場合 (hh:mm など) は 2 桁の分。
mmm	3 文字の月の名前。
mmmm[m...]	月を示す長い文字列。指定された m の数が月名の文字数を超えるまで、m と同じ数の文字数が使用される。
d	曜日を示す 1 桁の数字 (0 = 日曜、6 = 土曜)。
dd	2 桁の日。
ddd	3 文字の曜日名。
dddd[d...]	曜日を示す長い文字列。指定された d の数が曜日名の文字数を超えるまで d と同じ数の文字数が使用される。
jjj	通し日数 (1 ~ 366)。

注意： 日付フォーマット文字列の中でマルチバイト文字は使用できません。データベースの照合順が 932JPN などのマルチバイト照合順であっても、使用できるのは 1 バイト文字だけです。日付フォーマットの文字列でマルチバイト文字を使用するには、連結演算子を使用します。たとえば、'?' がマルチバイト文字の場合、次のように連結演算子を使用してマルチバイト文字を日付フォーマット文字列の外に移動します。

```
SELECT DATEFORMAT (StartDate, 'yy') + '?'
FROM Employees;
```

各記号は、フォーマットされた日付の対応するデータで置き換えられます。数字の出力ではなく文字を表すフォーマット記号は大文字で入力でき、その場合、置き換えられる文字も大文字になります。数字の場合、フォーマット文字列に大文字と小文字を混在させると、出力に先行ゼロが付きません。

記号の大文字、小文字を変更することで、数字の埋め込みを制御できます。大文字か小文字でそろえると (MM、mm、DD、dd) 空いたフィールドに 0 が埋め込まれます。大文字と小文字を混ぜると (Mm、mM、Dd、dD)、0 は埋め込まれません。各表示は必要なスペースだけを使用します。次に例を示します。

```
SELECT dateformat ( cast ('2011/01/01' as date ), 'yyyy/Mm/Dd' )
```

この例では、次の値が返されます。

```
2011/1/1
```

例

DATE_FORMAT の設定と、その設定で 2011 年 5 月 21 日 (土) に次の文を実行した場合の結果を表に示します。

```
SELECT CURRENT DATE
```

表 15 : **DATE_FORMAT** の設定

DATE FORMAT	SELECT CURRENT DATE
yyyy/mm/dd/ddd	2011/05/21/sat
jjj	141
mmm yyyy	may 2011
mm-yyyy	05-2011

参照：

- RETURN_DATE_TIME_AS_STRING オプション (761 ページ)
- TIME_FORMAT オプション (801 ページ)

DATE_ORDER オプション

日付フォーマットの解釈を制御します。

指定できる値

'MDY'、'YMD'、または 'DMY'

デフォルト値

'YMD'これは ISO 日付フォーマット仕様準拠です。

Open Client および DJBC 接続では 'MDY'

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

DATE_ORDER は、10/11/12 が 1912 年 10 月 11 日なのか、1910 年 11 月 12 日なのか、それとも 1912 年 11 月 10 日なのかを指定するために使用します。このオプションには、値として 'MDY'、'YMD'、または 'DMY' を指定できます。

DBCC_LOG_PROGRESS オプション

sp_iqcheckdb システムストアドプロシージャの進捗をレポートします。

指定できる値

ON、OFF

デフォルト値

OFF

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、**SET ANY PUBLIC OPTION** システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

DBCC_LOG_PROGRESS が ON の場合、**sp_iqcheckdb** システムストアドプロシージャは進行メッセージを IQ メッセージファイルに送信します。このメッセージによって、ユーザは **sp_iqcheckdb** 処理の進行状況を追跡できます。

ストアドプロシージャについては、『リファレンス：ビルディングブロック、テーブル、およびプロシージャ』を参照してください。

例

次は、**sp_iqcheckdb 'check database'** コマンドの進捗ログの出力例です。

```
IQ Utility Check Database Start CHECK STATISTICS table: tloansf Start
CHECK STATISTICS for field: aqsn_dt Start CHECK STATISTICS processing
index: IQ_IDX_T444_C1_FP Start CHECK STATISTICS processing index:
tloansf_aqsn_dt_HNG Done CHECK STATISTICS field: aqsn_dt
```

次は、`sp_iqcheckdb 'allocation table nation'` コマンドの進捗ログの出力例です。

```
Start ALLOCATION table: nation
Start ALLOCATION processing index: nationhg1
Done ALLOCATION table: nation
Done ALLOCATION processing index: nationhg1
```

DBCC_PINNABLE_CACHE_PERCENT オプション

sp_iqcheckdb システムストアードプロシージャが使用するキャッシュの割合を制御します。

指定できる値
0 - 100

デフォルト値
50

スコープ
オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。次回 **sp_iqcheckdb** を実行したときに有効になります。

備考

sp_iqcheckdb システムストアードプロシージャは、このオプションで指定された固定数のバッファを使用して動作します。**sp_iqcheckdb** のパフォーマンスを最大限に向上させるため、デフォルトでキャッシュの大部分は予約されています。

ストアードプロシージャについては、『リファレンス：ビルディングブロック、テーブル、およびプロシージャ』を参照してください。

DEBUG_MESSAGES オプション

DEBUG ONLY 句を含む MESSAGE 文を実行するかどうかを制御します。

指定できる値
ON、OFF

デフォルト値
OFF

スコープ
オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考
このオプションでは、DEBUG ONLY 句が指定された **MESSAGE** 文を含むストアードプロシージャでのデバッグメッセージの動作を制御できます。デフォルトではこのオプションは OFF で、**MESSAGE** 文の実行時にデバッグメッセージは表示されません。**DEBUG_MESSAGES** を ON に設定すると、すべてのストアードプロシージャでデバッグメッセージが有効になります。

注意： DEBUG ONLY メッセージは、**DEBUG_MESSAGES** オプションが OFF になっていれば低コストなため、通常、実稼働システムのストアードプロシージャ内に残されます。ただし、頻繁に実行される場所ではパフォーマンスがやや低下する可能性があるため、慎重に使用してください。

参照：

- MESSAGE 文 (457 ページ)

DEDICATED_TASK オプション

要求処理のタスクが、1つの接続からの要求だけを処理するようにします。

指定できる値
ON、OFF

デフォルト値
OFF

スコープ
オプションは、現在の接続の存続期間において、個々の接続または PUBLIC ロールにテンポラリオプションとしてのみ設定できます。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。すぐに有効になります。

備考

DEDICATED_TASK 接続オプションを ON に設定すると、要求処理タスクは、その接続の要求処理専用となります。このオプションを有効にして接続を事前に確立しておくことで、データベースサーバが応答しなくなった場合にデータベースサーバの状態に関する情報を収集できます。

DEFAULT_DBSPACE オプション

テーブルを作成するデフォルトの DB 領域を変更します。

指定できる値

DB 領域名を含む文字列

デフォルト値

"(空の文字列)"

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

DEFAULT_DBSPACE を使用することで、管理者は任意のロールまたはユーザのデフォルトの DB 領域を設定でき、ユーザは自分のデフォルトの DB 領域を設定できます。

グローバルテンポラリテーブルには、IQ_SYSTEM_TEMP が使用されます。ただし、SYSTEM を指定するテーブル IN 句が使用されている場合は、SA グローバルテンポラリテーブルが作成されます。

データベースの作成時に、**PUBLIC.DEFAULT_DBSPACE** オプションの設定が空か、または明示的に IQ_SYSTEM_MAIN に設定されている場合は、システム DB 領域 IQ_SYSTEM_MAIN が作成されて暗黙的に定義されます。データベースの作成後すぐに、メイン DB 領域をもう 1 つ作成し、DB 領域 IQ_SYSTEM_MAIN における

CREATE 権限を PUBLIC から取り消します。そして、新しい DB 領域で、DB 領域における CREATE 権限を特定のユーザまたは PUBLIC に付与し、**PUBLIC.DEFAULT_DBSPACE** を新しいメイン DB 領域に設定します。次に例を示します。

```
CREATE DBSPACE user_main USING FILE user_main
'user_main1' SIZE 10000;
GRANT CREATE ON user_main TO PUBLIC;
REVOKE CREATE ON IQ_SYSTEM_MAIN FROM PUBLIC;
SET OPTION PUBLIC.DEFAULT_DBSPACE = 'user_main';
```

例

次の例では、すべての DB 領域での CONNECT 権限と RESOURCE 権限がユーザ usrA と usrB に付与され、これらのユーザのそれぞれに特定の DB 領域での CREATE 権限が付与されます。

```
GRANT CONNECT, RESOURCE TO usrA, usrB
IDENTIFIED BY pwdA, pwdB;
GRANT CREATE ON dbbsp1 TO usrA;
GRANT CREATE ON dbbsp3 TO usrB;
SET OPTION "usrA".default_dbSPACE = 'dbbsp1';
SET OPTION "usrB".default_dbSPACE = 'dbbsp3';
SET OPTION "PUBLIC".default_dbSPACE = dbbsp2;

CREATE TABLE "DBA".t1(c1 int, c2 int);
INSERT INTO t1 VALUES (1, 1);
INSERT INTO t1 VALUES (2, 2);
COMMIT;
```

UsrA が以下を実行します。

```
CREATE TABLE "UsrA".t1(c1 int, c2 int);
INSERT INTO t1 VALUES (1, 1);
INSERT INTO t1 VALUES (2, 2);
COMMIT;
```

UsrB が以下を実行します。

```
CREATE TABLE "UsrB".t1(c1 int, c2 int);
INSERT INTO t1 VALUES (1, 1);
INSERT INTO t1 VALUES (2, 2);
COMMIT;
```

DBA が以下を実行します。

```
SELECT Object, DbspaceName, ObjSize
FROM sp_iqindexinfo();
```

sp_iqindexinfo の結果は次のようになります。

DBA.t1	dbbsp2	200k
DBA.t1.ASIQ_IDX_T730_C1_FP	dbbsp2	288k
DBA.t1.ASIQ_IDX_T730_C2_FP	dbbsp2	288k

usrA.t1	dbbsp1	200k
usrA.t1.ASIQ_IDX_T731_C1_FP	dbbsp1	288k
usrA.t1.ASIQ_IDX_T731_C2_FP	dbbsp1	288k
usrB.t1	dbbsp3	200k
usrB.t1.ASIQ_IDX_T732_C1_FP	dbbsp3	288k
usrB.t1.ASIQ_IDX_T732_C2_FP	dbbsp3	288k

DEFAULT_DISK_STRIPING オプション

すべての DB 領域にディスクストライピングの値を設定します。

指定できる値

ON、OFF

デフォルト値

ON

スコープ

オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。すぐに有効になります。

備考

デフォルトでは、IQ メインストアにあるすべての DB 領域のディスクストライピングは ON になります。CREATE DBSPACE がストライピングを指定しない場合、このオプションは、CREATE DBSPACE によってのみ使用され、デフォルトのストライピング値を定義します。

参照：

- CREATE DBSPACE 文 (150 ページ)

DEFAULT_HAVING_SELECTIVITY_PPM オプション

ほとんどの HAVING 句のデフォルトの選択性的の見積もりをオプティマイザに提供します (100 万分の 1 単位)。

指定できる値

0 - 1000000

デフォルト値

0

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できません。データベースレベルで設定した場合、値は新しいユーザのデフォルト値にな

りますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

DEFAULT_HAVING_SELECTIVITY_PPM は HAVING 句の選択性を設定し、オプティマイザの見積もりを上書きします。HAVING 句は GROUP BY 句、または集合関数のみで構成されている select リストを指定したクエリの結果をフィルタします。

DEFAULT_HAVING_SELECTIVITY_PPM がデフォルトの 0 に設定されている場合、オプティマイザは HAVING 句によってフィルタされるローの数を推定します。IQ オプティマイザに十分な情報がなく、正確な選択性が決められない場合、一般的な見積もり値の 40% が採用されます。**DEFAULT_HAVING_SELECTIVITY_PPM** を使用して、クエリ内のすべての HAVING 述部に対するオプティマイザの見積もりを上書きできます。

ユーザは、クエリ内の個々の HAVING 句の選択性を指定することもできます。詳細については、『リファレンス：ビルディングブロック、テーブル、およびプロシージャ』を参照してください。

DEFAULT_ISQL_ENCODING オプション [Interactive SQL]

READ 文と OUTPUT 文が使用するコードページを指定します。

指定できる値

identifier または *string*

デフォルト値

システムコードページを使用 (空の文字列)

スコープ

現在の接続の継続中に、テンポラリオプションとしてのみ設定できます。

備考

DEFAULT_ISQL_ENCODING は、ファイルの読み込み／書き込み時に使用するコードページの指定に使用されます。これを永続的に設定することはできません。デフォルトのコードページは、実行中のプラットフォームのデフォルトのコードページです。英語版 Windows のマシンでは、デフォルトのコードページは 1252 です。

Interactive SQL は、特定の **OUTPUT** 文または **READ** 文で使用するコードページを次のリストの順で決定します (先に出てきたコードページが優先されます)。

- **OUTPUT** 文または **READ** 文の **ENCODING** 句で指定されているコードページ
- `DEFAULT_ISQL_ENCODING` オプションで指定されたコードページ (このオプションが設定されている場合)
- Interactive SQL を実行しているコンピュータのデフォルトのコードページ

例

コード化を UTF-16 (Unicode ファイルの読み込み用) に設定します。

```
SET TEMPORARY OPTION DEFAULT_ISQL_ENCODING = 'UTF-16'
```

参照：

- `OUTPUT` 文 [Interactive SQL] (463 ページ)
- `READ` 文 [Interactive SQL] (478 ページ)

DEFAULT_KB_PER_STRIPE オプション

書き込み操作が次のストライプに移動する前の、ストライプへのデータの書き込み量の上限を KB 単位で設定します。

この設定は、IQ メインストアのすべての DB 領域のデフォルトサイズです。

指定できる値

1 以上の整数

デフォルト値

1

スコープ

オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、`SET ANY SYSTEM OPTION` システム権限が必要です。すぐに有効になります。

備考

デフォルト値の 1KB は、1 ページが圧縮され、圧縮されたページが 1 回の操作でディスクに書き込まれることを意味します。指定したページサイズに関係なく、次の操作は、DB 領域の次の DB ファイルに書き込みます。

次のストライプに移動する前に複数のページを同じストライプに書き込むには、**DEFAULT_KB_PER_STRIPE** の設定を変更します。たとえば、ページサイズが 128KB である場合、**DEFAULT_KB_PER_STRIPE** を 512KB に設定すると、SAP

Sybase IQ はページの書き込みをキューに保留し、圧縮ページのサイズが512KBを超えると、ディスクに書き込みます。

CREATE DBSPACE でストライプサイズが指定されていない場合、このオプションは、**CREATE DBSPACE** によってのみ使用され、IQ メインストアの DB 領域のデフォルトのストライピングサイズを定義します。

参照：

- CREATE DBSPACE 文 (150 ページ)

DEFAULT LIKE MATCH SELECTIVITY PPM オプション

ほとんどの **LIKE** 述部のデフォルトの選択性を見積もりをオプティマイザに提供します (100 万分の 1 単位)。

指定できる値

0 ~ 1000000

デフォルト値

150000

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

DEFAULT LIKE MATCH SELECTIVITY PPM は、たとえば `LIKE 'string%string'` (% はワイルドカード文字) のような一般的な LIKE 述部のデフォルトの選択性を設定します。

他に選択性に関する情報が提供されておらず、一致文字列が、一連の定数文字と 1 つのワイルドカードで始まっていない場合、オプティマイザはこのオプションを参照します。

カラムに、LF インデックスまたは 1 バイト、2 バイト、3 バイト FP インデックスのいずれかがある場合、オプティマイザは正しい値を得ることができるため、この値を使用する必要はありません。

ユーザが、クエリ内で選択性を指定することもできます。ユーザ指定の条件ヒントについては、『リファレンス：ビルディングブロック、テーブル、およびプロシージャ』を参照してください。

参照：

- `DEFAULT_LIKE_RANGE_SELECTIVITY_PPM` オプション (651 ページ)
- `FP_LOOKUP_SIZE` オプション (665 ページ)

DEFAULT_LIKE_RANGE_SELECTIVITY_PPM オプション

オプティマイザに、先頭の定数 `LIKE` 述部のデフォルトの選択性の見積もりを提供します (100 万分の 1 単位)。

指定できる値

1000000 ~ 1

デフォルト値

150000

スコープ

オプションは、データベース (`PUBLIC`) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの `PUBLIC` 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、`SET ANY PUBLIC OPTION` システム権限が必要です。個々の接続または `PUBLIC` ロールに一時的に設定できます。すぐに有効になります。

備考

`DEFAULT_LIKE_RANGE_SELECTIVITY_PPM` は、`LIKE 'string%'` (一致文字列は一連の定数文字の後ろにワイルドカード文字 (%) が 1 つ付いたもの) という形の `LIKE` 述部のデフォルトの選択性を設定します。選択性に関する他の情報が提供されていない場合、オプティマイザはこのオプションを参照します。

カラムに、`LF` インデックスまたは 1 バイト、2 バイト、3 バイト `FP` インデックスのいずれかがある場合、オプティマイザは正しい値を得ることができるため、この値を使用する必要はありません。

ユーザが、クエリ内で選択性を指定することもできます。ユーザ指定の条件ヒントについては、『リファレンス：ビルディングブロック、テーブル、およびプロシージャ』を参照してください。

参照：

- DEFAULT_LIKE_MATCH_SELECTIVITY_PPM オプション (650 ページ)
- FP_LOOKUP_SIZE オプション (665 ページ)

DEFAULT_PROXY_TABLE_ROW_COUNT オプション

プロキシテーブルから返されるローの数のデフォルト見積もりを上書きできます。

指定できる値

0 ~ 4294967295

デフォルト値

200000

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

DEFAULT_TABLE_UDF_ROW_COUNT オプション

テーブル UDF (C、C++、または Java のいずれかのテーブル UDF) が返すローの推定数のデフォルト値を上書きできます。

指定できる値

0 ~ 4294967295

デフォルト値

200000

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

テーブル UDF では、**DEFAULT_TABLE_UDF_ROW_COUNT** オプションを使用して、テーブル UDF が返すローの推定数をクエリプロセッサに伝達できます。これは、Java のテーブル UDF でこの情報を伝達する際の唯一の方法です。一方、C または C++ のテーブル UDF では、UDF の開発者は、

EXTFNAPIV4_DESCRIBE_PARM_TABLE_NUM_ROWS describe パラメータを使用して、describe フェーズでこの情報を公開することにより、返すロー数の予想値を伝達することを考慮する必要があります。

EXTFNAPIV4_DESCRIBE_PARM_TABLE_NUM_ROWS の値は、**DEFAULT_PROXY_TABLE_UDF_ROW_COUNT** オプションの値を常に上書きします。

DELAYED_COMMIT_TIMEOUT オプション

COMMIT の後でサーバが制御をアプリケーションにいつ戻すかを決定します。

指定できる値

整数値 (ミリ秒単位)

デフォルト値

500

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

DELAYED_COMMITS は OFF 以外の値をとらないため、SAP Sybase IQ はこのオプションを無視します。

DELAYED_COMMIT オプション

COMMIT の後でサーバが制御をアプリケーションにいつ戻すかを決定します。

指定できる値
OFF

デフォルト値
OFF。これが ISO の COMMIT 動作に準拠しています。

スコープ
オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考
OFF に設定されている場合 (SAP Sybase IQ では OFF のみ指定可能)、アプリケーションは **COMMIT** 内容がディスクに書き込まれるまで待たなければなりません。ANSI/ISO の COMMIT の動作を得るには、このオプションを OFF に設定する必要があります。

DISABLE_RI_CHECK オプション

ロード、挿入、更新、削除の操作で参照整合性のチェックを省略することを許可し、パフォーマンスを向上させます。

指定できる値
ON、OFF

デフォルト値
OFF

スコープ
オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、

そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

DISABLE_RI_CHECK が ON に設定されている場合、ユーザの責任において、要求中に参照整合性の違反が起きないようにする必要があります。

DIVIDE_BY_ZERO_ERROR オプション [TSQL]

ゼロ除算のレポートを制御します。

指定できる値

ON、OFF

デフォルト値

ON

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

このオプションは、ゼロ除算をエラーとしてレポートするかどうかを指示します。オプションを ON に設定すると、ゼロ除算は SQLSTATE 22012 のエラーになります。

このオプションを OFF に設定すると、ゼロ除算はエラーになりません。NULL が返されます。

DQP_ENABLED オプション

テンポラリデータベースオプション **DQP_ENABLED** を使用すると、接続レベルで分散クエリ処理を有効または無効にできます。

指定できる値

ON、OFF

デフォルト値

ON

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

テンポラリデータベースオプション **DQP_ENABLED** を OFF に設定することによって、現在の接続の DQP を無効にすることができます。このオプションを ON (デフォルト値) に設定すると、現在の接続における DQP を有効にすることができますが、それが可能なのは現在の接続の論理サーバに対するユーザのログインポリシーによって、そのユーザに対して DQP が有効になっている場合のみです。

ユーザのログインポリシーに従って DQP が無効にされている場合、

DQP_ENABLED を ON に設定すると、次のエラーが発生します。

```
Invalid setting for option 'DQP_ENABLED'
```

注意： ユーザのログインポリシーオプションへのすべての変更は、新しい接続のみに影響します。既存の接続のログインポリシーオプション設定は、接続が最初に確立した時点の設定に基づきます。

DQP_ENABLED_OVER_NETWORK オプション

テンポラリデータベースオプション **DQP_ENABLED_OVER_NETWORK** を使用すると、接続レベルでネットワーク経由の分散クエリ処理を有効または無効にできます。

指定可能な値

ON、OFF

デフォルト

OFF

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

PUBLIC または他のユーザ／ロールに対してこのオプションを設定するには SET ANY PUBLIC OPTION システム権限が必要です。個々またはパブリックに一時的に設定できます。

備考

テンポラリデータベースオプション **DQP_ENABLED_OVER_NETWORK** を ON に設定すると、現在の接続におけるネットワーク経由の DQP を有効にすることができます。OFF (デフォルト) に設定しても効果はありません。また、DQP_ENABLED 論理サーバポリシーオプションの設定により、現在の接続におけるクエリで DQP がネットワーク経由で使用されるかどうかが決まります。

LS ポリシーオプション設定	データベースオプション設定	DQP Query の動作
DQP_ENABLED 1	DQP_ENABLED_OVER_NETWORK ON	現在の接続上のクエリがネットワーク経由で実行される。他のクエリは、共有テンポラリストアを使用する。
DQP_ENABLED 2	DQP_ENABLED_OVER_NETWORK ON	すべてのクエリがネットワーク経由で実行される。
DQP_ENABLED 2	DQP_ENABLED OFF	すべてのクエリがシンプレックスモードで実行される。

注意：論理サーバポリシーオプションに対する変更はすべて、新しい接続のみに影響します。既存の接続の論理サーバポリシーオプションは、接続が最初に確立した時点の設定に従います。

EARLY_PREDICATE_EXECUTION オプション

クエリの最適化の前に簡単なローカル述部を実行するかどうかを制御します。

指定できる値

ON、OFF

デフォルト値

ON

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

このオプションが ON (デフォルト値) の場合、オプティマイザは、クエリプラン内の "Estimated Result Rows" の値をより正確なものにするために、ジョインの順序付け、ジョインアルゴリズムの選択、グループ化アルゴリズムの選択などを行ってクエリを最適化する前に、ローカルカラムと制約のみを含む述部を検索、準備、実行します。このオプションが OFF の場合、オプティマイザは単純な述部を検索および準備しますが、クエリの最適化の前にこれらを実行することはありません。述部を実行しないと、"Estimated Result Rows" の最終的な値の正確さは劣ります。

通常、**EARLY_PREDICATE_EXECUTION** は常に ON にしておきます。それにより、多くのクエリのクエリプランが向上します。

EARLY_PREDICATE_EXECUTION が ON に設定されている場合、NOEXEC オプションが ON でも、SAP Sybase IQ はクエリプランの生成に先立ってすべてのクエリのローカル述部を実行します。生成されるクエリプランは実行時プランと同一です。ルートノードに関する次の情報がクエリプランに含まれます。

- ローカルの不変の述部を実行するのに使われるスレッド。1 より大きな数の場合、ローカルの不変の述部を並列実行します。
- `Early_Predicate_Execution`：オプションが OFF かどうかを示します。
- `Time of Cursor Creation`：カーソルが作成された時間。

このオプションが実行を制御する単純な述部は、クエリプラン内では不変の述部として扱われます。リーフノード上にローカルの不変の述部がある場合、そのリーフノードに関する次の情報がクエリプランに含まれます。

- `Generated Post Invariant Predicate Rows`：ローカルの不変の述部を実行した後の実際の結果。
- `Estimated Post Invariant Predicate Rows`：ローカルの不変の述部の推定選択性によって算出されます。
- `Time of Condition Start`：ローカルの不変の述部の実行を開始した時間。
- `Time of Condition Done`：ローカルの不変の述部の実行が完了した時間。
- `Elapsed Condition Time`：ローカルの不変の述部を実行する際の経過時間。

ENABLE_ASYNC_IO オプション

RLV 永続ログにより使用される非同期 IO を、DBA が有効化または無効化できるようにします。

指定可能な値

TRUE、FALSE

値を変更した場合、データベースを閉じてから再度開くか、サーバを再起動する必要があります。

デフォルト

TRUE

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、`SET ANY PUBLIC OPTION` システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。許可される場合、任意の他のユーザまたはロール、あるいはロールを介してすべてのユーザに設定できます。すぐに有効になります。

ENABLE_LOB_VARIABLES オプション

ラージオブジェクト変数のデータ型変換を制御します。

指定できる値

ON、OFF

デフォルト値

OFF

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

ENABLE_LOB_VARIABLES は、ラージオブジェクト変数のデータ型変換を制御します。

ENABLE_LOB_VARIABLES が OFF の場合、32K 未満のラージオブジェクト変数は暗黙的に変換されます。ラージオブジェクト変数が 32K 以上の場合はエラーがレポートされます。LONG VARCHAR 変数は VARCHAR データ型に暗黙的に変換され、32K を超えた部分がトランケートされます。LONG BINARY 変数は VARBINARY データ型に暗黙的に変換され、32K を超えた部分がトランケートされます。

ENABLE_LOB_VARIABLES が ON の場合、どのサイズのラージオブジェクト変数も元のデータ型とサイズが保持されます。

例

32K を超えるラージオブジェクト変数のデータ型とサイズを保持するには、次のように入力します。

```
SET TEMPORARY OPTION ENABLE_LOB_VARIABLES = ON
```


EXTENDED_JOIN_SYNTAX オプション

マルチテーブルジョインに対してあいまいな構文を持つクエリを許可するか、またはエラーとして報告するかを制御します。

指定できる値

ON、OFF

デフォルト値

ON

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

このオプションを使用して、NULL 入力テーブルに重複する相関名があるために構文があいまいになっている外部ジョインを含むクエリに対して、構文エラーを報告します。

報告されるのは、次のようなジョイン句があるクエリです。ここで C1 は条件です。

```
( R left outer join T , T join S on ( C1 ) )
```

EXTENDED_JOIN_SYNTAX を ON に設定すると、このクエリは次のように解釈されます。ここで、C1 と C2 は条件です。

```
( R left outer join T on ( C1 ) ) join S on ( C2 )
```

FLOATING_POINT_ACCUMULATOR オプション

浮動小数点数の SUM または AVG に使用するアキュムレータの選択を制御します。

指定可能な値

1, 2, 3

デフォルト

2

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

設定 1 (高速アキュムレータ) は設定 2 よりも高速で、float 値と double 値に使用されるスペースが削減されます。この設定では、double 数と float 数の加算に単一の倍精度変数が使用され、このようなアプローチの既知の精度限界が適用されます。

設定 2 (デフォルト) (ミディアムアキュムレータ) では、複数の倍精度変数を使用して float 値と double 値が集計されます。1e-20 ~ 1e20 の範囲の加数に対して非常に精度が高くなります。この範囲以外では若干精度が下がりますが、それでもほとんどのアプリケーションに対して十分な精度が得られます。設定 2 を使用すれば、オプティマイザは設定 3 よりも簡単により高性能なハッシュを選択できます。

設定 3 (ラージアキュムレータ) は、すべての float 値と double 値に対して非常に精度が高くなりますが、そのサイズが原因でハッシュ最適化が使用できないことが多いため、ほとんどのアプリケーションの性能限界になります。

FORCE_DROP オプション

SAP Sybase IQ が DROP コマンドの実行中にデータベースのディスク領域を再利用するのではなくリークすることを指定します。

指定できる値

ON、OFF

デフォルト値

OFF

スコープ

オプションは、現在の接続の存続期間において、個々の接続または PUBLIC ロールにテンポラリーオプションとしてのみ設定できます。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。すぐに有効になります。

備考

破損インデックス、カラム、またはテーブルを削除する必要がある場合は、**FORCE_DROP** オプションを ON に設定してください。これによって、削除対象のオブジェクト内に誤りまたは誤りの可能性があるファイル領域割り付け情報をもとに、フリーリストが不適切に更新されないようにします。破損オブジェクトを削除した後はサーバスイッチの **-iqfrec** と **-iqdroplks** を使用して、ファイル領域を再利用できます。

オブジェクトを強制削除するときは、DBA 以外にデータベースに接続しているユーザがないことを確認してください。強制削除の後、サーバをすぐに再起動する必要があります。

オブジェクトの強制削除は、SAP Sybase 製品の保守契約を結んでいるサポートセンタから指示があった場合を除いて行わないでください。

システムリカバリとデータベース修復のための **FORCE_DROP** プロシージャについては、『管理：バックアップ、リストア、およびデータリカバリ』を参照してください。

FORCE_NO_SCROLL_CURSORS オプション

すべてのカーソルをスクロールしないよう強制的に設定します。

指定できる値

ON、OFF

デフォルト値

OFF

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

データベースオプション

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

デフォルトでは、すべてのカーソルはスクロール可能です。スクロールカーソルでホスト変数が宣言されていない場合、SAP Sybase IQ は結果を一時的に保持するためのバッファを生成します。結果セットの各ローがこのバッファに保持され、後方スクロールを可能になります。

FORCE_NO_SCROLL_CURSORS を ON に設定すると、一時的に使用されるメモリが削減されます。このオプションは、大量 (数百万) のローを取得する場合に役立ちます。ただし、フロントエンドアプリケーションで後方スクロールカーソルオペレーションが頻繁に使用される場合は、このオプションを OFF に設定した方が、クエリの応答が速くなります。

フロントエンドアプリケーションで後方スクロールがほとんど行われない場合は、**FORCE_NO_SCROLL_CURSORS** = 'ON' を永続的な PUBLIC オプションに設定します。メモリの節約になるため、クエリのパフォーマンスが向上します。

FORCE_UPDATABLE_CURSORS オプション

更新可能として宣言されていないカーソルを更新可能にするかどうかを制御します。

指定できる値

ON、OFF

デフォルト値

OFF

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

FORCE_UPDATABLE_CURSORS が ON の場合は、更新可能として宣言されていないカーソルを更新できます。このオプションによって、**DECLARE CURSOR** 文の FOR UPDATE 句を指定しなくても、フロントエンドアプリケーションで更新可能カーソルを使えるようになります。

SAP Sybase IQ では、どうしても必要な場合を除いて **FORCE_UPDATABLE_CURSORS** は使用しないでください。

FP_LOOKUP_SIZE オプション

SAP Sybase IQ 15 データベースのルックアップ FP インデックスのルックアップページ数とキャッシュメモリ割り付けを指定します。

指定できる値

1MB ~ 4096MB

デフォルト値

16 MB

スコープ

オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。すぐに有効になります。

依存性

FP_LOOKUP_SIZE は **FP_NBIT_IQ15_COMPATIBILITY='ON'** が設定されている実行中のデータベースに適用されます。**FP_NBIT_IQ15_COMPATIBILITY='OFF'** の場合、データベースエンジンはこのオプションを無視します。

備考

FP_LOOKUP_SIZE は、**FP_NBIT_IQ15_COMPATIBILITY='ON'** の場合にルックアップページの最大数を制御します。

FP_LOOKUP_SIZE は public として設定する必要があります。許容される構文は次のとおりです。

```
SET OPTION public.FP_LOOKUP_SIZE = 1
```

重要： **FP_NBIT_IQ15_COMPATIBILITY** オプションは、完全なデータベース互換性ではなく、SAP Sybase IQ 15 で利用可能なものと同様のトークン化 FP サポートを提供します。すべての SAP Sybase IQ 15 ランタイム動作は 16.0 インタフェースを使

用して実行可能です。16.0 データベースは、**FP_NBIT_IQ15_COMPATIBILITY='ON'** で実行しないでください。

参照：

- FP_LOOKUP_SIZE_PPM オプション (666 ページ)
- MINIMIZE_STORAGE オプション (722 ページ)
- FP_NBIT_IQ15_COMPATIBILITY オプション (670 ページ)

FP_LOOKUP_SIZE_PPM オプション

SAP Sybase IQ 15 データベースの FP インデックスに対するメインバッファキャッシュの割り付け量を制御します。

指定できる値

1000000 ~ 1

デフォルト値

2500

スコープ

オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。すぐに有効になります。

依存性

FP_LOOKUP_SIZE_PPM は **FP_NBIT_IQ15_COMPATIBILITY='ON'** が設定されている実行中のデータベースに適用されます。**FP_NBIT_IQ15_COMPATIBILITY='OFF'** の場合、SAP Sybase IQ はこのオプションを無視します。

備考

FP_LOOKUP_SIZE_PPM は、**FP_NBIT_IQ15_COMPATIBILITY='ON'** の場合に、ルックアップページの最大数を制御し、この数をメインメモリの 100 万分の 1 の値に制限します。この値は、**FP_LOOKUP_SIZE_PPM** * size of main memory / 1,000,000 として計算されます。メインメモリのサイズは **-iqmc** サーバ起動パラメータで指定されたサイズです。

重要： **FP_NBIT_IQ15_COMPATIBILITY** オプションは、完全なデータベース互換性ではなく、SAP Sybase IQ 15 で利用可能なものと同様のトークン化 FP サポートを提供します。すべての SAP Sybase IQ 15 ランタイム動作は 16.0 インタフェースを使用して実行可能です。16.0 データベースは、**FP_NBIT_IQ15_COMPATIBILITY='ON'** で実行しないでください。

参照：

- FP_LOOKUP_SIZE オプション (665 ページ)
- MINIMIZE_STORAGE オプション (722 ページ)
- FP_NBIT_IQ15_COMPATIBILITY オプション (670 ページ)

FP_NBIT_AUTOSIZE_LIMIT オプション

NBit FP として暗黙的にロードするカラム内の重複しない値の数を制限します。

指定可能な値

0 ~ 2,147,483,647

デフォルト

1,048,576

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

依存性

FP_NBIT_AUTOSIZE_LIMIT は、**FP_NBIT_IQ15_COMPATIBILITY='OFF'** が設定されている実行中のデータベースに適用されます。**FP_NBIT_IQ15_COMPATIBILITY='ON'** の場合、データベースエンジンはこのオプションを無視します。

備考

FP_NBIT_AUTOSIZE_LIMIT は、新規に作成されたすべてのカラムの重複しない値の数を、明示的な IQ UNIQUE 設定がなくても制限します。

FP_NBIT_AUTOSIZE_LIMIT オプションの制約を受けたカラムは、Flat FP または NBit FP インデックスとともにロードされます。

- **FP_NBIT_AUTOSIZE_LIMIT** が 0 より大きく、2,147,483,647 より小さい場合、カラムは NBit FP インデックスとともにロードされます。
- **FP_NBIT_AUTOSIZE_LIMIT** が 0 の場合、カラムは Flat FP インデックスとともにロードされます。

FP_NBIT_AUTOSIZE_LIMIT と **FP_NBIT_LOOKUP_MB** によって、データロード時の NBit カラムの最大サイズが設定されます。重複しない値の数が

FP_NBIT_AUTOSIZE_LIMIT よりも小さく、ディクショナリの合計サイズ (値とカウント) が **FP_NBIT_LOOKUP_MB** よりも小さければ、カラムは NBit としてロードされます。 **FP_NBIT_AUTOSIZE_LIMIT** を超えるロードの場合、

FP_NBIT_ROLLOVER_MAX_MB よりも小さければ、カラムは Flat FP にロールオーバーされます。

FP_NBIT_ROLLOVER_MAX_MB を超える DML 操作は、**FP_ENFORCE_LIMITS='ON'** の場合は、ロールバックされ、エラーがレポートされます。

FP_NBIT_ENFORCE_LIMITS='OFF' の場合は、カラムは次の NBit レベルに遷移します。

sp_iqindexmetadata はフラット FP カラムまたは NBit FP カラムについての詳細情報を返します。 **sp_iqrebuildindex** は、明示的または暗黙的な NBit FP カラムの制限を変更したり、デフォルトカラムインデックスをフラット FP または NBit FP として再フォーマットできます。

その他の情報

- 『リファレンス：ビルディングブロック、テーブル、およびプロシージャ』 > 「システムプロシージャ」 > 「システムストアプロシージャのアルファベット順リスト」 > 「**sp_iqrebuildindex**」
- 『リファレンス：ビルディングブロック、テーブル、およびプロシージャ』 > 「システムプロシージャ」 > 「システムストアプロシージャのアルファベット順リスト」 > 「**sp_iqindexmetadata**」

参照：

- FP_NBIT_ENFORCE_LIMITS オプション (668 ページ)
- FP_NBIT_IQ15_COMPATIBILITY オプション (670 ページ)
- FP_NBIT_LOOKUP_MB オプション (672 ページ)
- FP_NBIT_ROLLOVER_MAX_MB オプション (673 ページ)

FP_NBIT_ENFORCE_LIMITS オプション

明示的および暗黙的な NBit カラムのサイズ制限を強制します。

指定可能な値

ON、OFF

デフォルト

OFF

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

依存性

FP_NBIT_ENFORCE_LIMITS は、**FP_NBIT_IQ15_COMPATIBILITY=OFF** が設定されている実行中のデータベースに適用されます。**FP_NBIT_IQ15_COMPATIBILITY=ON** の場合、データベースエンジンはこのオプションを無視します。

備考

DML 操作では、IQ UNIQUE 制約に設定されている明示的な制限値が **FP_NBIT_AUTOSIZE** よりも大きいときにカラム内の重複しない値の数がこの制限を超えている場合、または明示的な NBit のロールオーバーにおけるディクショナリサイズが **FP_NBIT_ROLLOVER_MAX_MB** の制限を超えている場合に、**FP_NBIT_ENFORCE_LIMITS** オプションがチェックされます。

- **FP_NBIT_ENFORCE_LIMITS=ON** の場合、DML 操作はエラーをスローし、ロールバックします。
- **FP_NBIT_ENFORCE_LIMITS=OFF** の場合、DML 操作は続行され、NBit ディクショナリ制限は無視されます。

sp_iqindexmetadata はフラット FP カラムまたは NBit FP カラムについての詳細情報を返します。**sp_iqrebuildindex** は、明示的または暗黙的な NBit FP カラムの制限を変更したり、デフォルトカラムインデックスをフラット FP または NBit FP として再フォーマットできます。

FP_NBIT_ENFORCE_LIMITS=ON が設定されているときに、**sp_iqrebuildindex** を使用して、重複しない値の数を Flat FP カラムの現在の制限よりも多い数に増やすと、エラーになります。**FP_NBIT_ENFORCE_LIMITS=OFF** の場合、**sp_iqrebuildindex** は、重複しない値の最大値である最大トークンに達するまでインデックスを再構築します。

その他の情報

- 『リファレンス：ビルディングブロック、テーブル、およびプロシージャ』 > 「システムプロシージャ」 > 「システムストアードプロシージャのアルファベット順リスト」 > 「**sp_iqrebuildindex**」
- 『リファレンス：ビルディングブロック、テーブル、およびプロシージャ』 > 「システムプロシージャ」 > 「システムストアードプロシージャのアルファベット順リスト」 > 「**sp_iqindexmetadata**」

参照：

- FP_NBIT_AUTOSIZE_LIMIT オプション (667 ページ)
- FP_NBIT_IQ15_COMPATIBILITY オプション (670 ページ)
- FP_NBIT_LOOKUP_MB オプション (672 ページ)
- FP_NBIT_ROLLOVER_MAX_MB オプション (673 ページ)

FP_NBIT_IQ15_COMPATIBILITY オプション

SAP Sybase IQ 15 で利用可能なものと同様のトークン化された FP インデックスをサポートします。

指定可能な値

ON、OFF

デフォルト

すべての新しい 16.0 データベースでは、**FP_NBIT_IQ15_COMPATIBILITY='OFF'**。
アップグレードされた SAP Sybase IQ 15 データベースでは、
FP_NBIT_IQ15_COMPATIBILITY='ON'。

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

FP_NBIT_IQ15_COMPATIBILITY オプションは、SAP Sybase IQ 15 で利用可能なものと同様のトークン化された FP をサポートします。16.0 で新規作成または変更された

トークン化 FP インデックスはすべて NBit になります。16.0 で利用可能な 15 スタイルの FP (1)、FP (2)、および FP (3) バイト FP インデックスは、読み取り専用アクティビティだけしか存在していなかったアップグレードされたデータベースのインデックスのみに限定されます。

FP_NBIT_IQ15_COMPATIBILITY ON/OFF 設定は、トークン化 FP の作成およびカットオフ動作にのみ関連します。

FP_NBIT_IQ15_COMPATIBILITY='ON' の場合、データベースエンジンは次のように動作します。

- **MINIMIZE_STORAGE**、**FP_LOOKUP_SIZE**、**FP_LOOKUP_SIZE_PPM** の各オプションを有効にします。
- **IQ UNIQUE (0)** が指定されている場合でも、**DATE** データ型を NBit FP として作成します。
- 3 バイト FP カットオフ (重複しない値の数 16,777,216) で、Flat FP にロールオーバーします。
- 幅が 255 以下のデータをトークン化できます。

FP_NBIT_IQ15_COMPATIBILITY='OFF' の場合、次のように動作します。

- **MINIMIZE_STORAGE**、**FP_LOOKUP_SIZE**、**FP_LOOKUP_SIZE_PPM** の各オプションを無視します。
- **DATE** データ型を自動的に NBit として生成しません。
- 幅が 32767 以下のデータをトークン化できます。
- NBit FP (トークン化) の上限は NBit 31 (重複しない値の数 2,147,483,648) です。
- NBit サイズオプションによって、ロールオーバーの動作が決まります。
 - **IQ UNIQUE (0)** が設定されている場合、カラムは Flat FP としてロードされます。
 - **IQ UNIQUE** が設定されていない場合、カラムは自動サイズ制限値に達するまで NBit としてロードされます。
 - 自動サイズ制限値よりも小さい **IQ UNIQUE (n)** が設定されている場合、カラムは NBit としてロードされます。

重要： **FP_NBIT_IQ15_COMPATIBILITY** オプションは、完全なデータベース互換性ではなく、SAP Sybase IQ 15 で利用可能なものと同様のトークン化 FP サポートを提供します。すべての SAP Sybase IQ 15 ランタイム動作は 16.0 インタフェースを使用して実行可能です。16.0 データベースは、**FP_NBIT_IQ15_COMPATIBILITY='ON'** で実行しないでください。

参照：

- FP_NBIT_AUTOSIZE_LIMIT オプション (667 ページ)
- FP_NBIT_ENFORCE_LIMITS オプション (668 ページ)
- FP_NBIT_LOOKUP_MB オプション (672 ページ)
- FP_NBIT_ROLLOVER_MAX_MB オプション (673 ページ)
- FP_LOOKUP_SIZE オプション (665 ページ)
- FP_LOOKUP_SIZE_PPM オプション (666 ページ)
- MINIMIZE_STORAGE オプション (722 ページ)

FP_NBIT_LOOKUP_MB オプション

暗黙的な NBit FP カラムの合計ディクショナリサイズを制限します。

指定可能な値

1 ~ 4,294,967,295

デフォルト

64MB

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

依存性

FP_NBIT_LOOKUP_MB は、FP_NBIT_IQ15_COMPATIBILITY='OFF' が設定されている実行中のデータベースに適用されます。

FP_NBIT_IQ15_COMPATIBILITY='ON' の場合、データベースエンジンはこのオプションを無視します。

備考

FP_NBIT_AUTOSIZE_LIMIT と **FP_NBIT_LOOKUP_MB** では、暗黙的な NBit カラムの最大サイズが設定されます。重複しない値の数が **FP_NBIT_AUTOSIZE_LIMIT** よりも小さく、ディクショナリの合計サイズ (値とカウント) が **FP_NBIT_LOOKUP_MB** よ

りも小さければ、カラムは NBit FP インデックスとともにロードされます。制限は FP_NBIT_ENFORCE_LIMITS オプションによって強制されます。

FP_NBIT_LOOKUP_MB の制限を超える DML 操作は、Flat FP インデックスにロールオーバーされます。

FP_NBIT_LOOKUP_MB と FP_NBIT_ROLLOVER_MAX_MB の制限を超える操作の場合、FP_NBIT_ENFORCE_LIMITS='OFF' であれば、NBit FP は次の NBit レベルに遷移します。

sp_iqindexmetadata はフラット FP カラムまたは NBit FP カラムについての詳細情報を返します。sp_iqrebuildindex は、明示的または暗黙的な NBit FP カラムの制限を変更したり、デフォルトカラムインデックスをフラット FP または NBit FP として再フォーマットできます。

その他の情報

- 『リファレンス：ビルディングブロック、テーブル、およびプロシージャ』 > 「システムプロシージャ」 > 「システムストアプロシージャのアルファベット順リスト」 > 「sp_iqrebuildindex」
- 『リファレンス：ビルディングブロック、テーブル、およびプロシージャ』 > 「システムプロシージャ」 > 「システムストアプロシージャのアルファベット順リスト」 > 「sp_iqindexmetadata」

参照：

- FP_NBIT_AUTOSIZE_LIMIT オプション (667 ページ)
- FP_NBIT_ENFORCE_LIMITS オプション (668 ページ)
- FP_NBIT_IQ15_COMPATIBILITY オプション (670 ページ)
- FP_NBIT_ROLLOVER_MAX_MB オプション (673 ページ)

FP_NBIT_ROLLOVER_MAX_MB オプション

暗黙的な NBit の Flat FP へのロールオーバーにおける合計ディクショナリサイズのスレッシュホールドを設定します。

指定可能な値

1 ~ 4,294,967,295

デフォルト

16384

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できません。データベースレベルで設定した場合、値は新しいユーザのデフォルト値にな

りますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

依存性

FP_NBIT_ROLLOVER_MAX_MB は、FP_NBIT_IQ15_COMPATIBILITY='OFF' が設定されている実行中のデータベースに適用されます。

FP_NBIT_IQ15_COMPATIBILITY='ON' の場合、データベースエンジンはこのオプションを無視します。

備考

FP_NBIT_AUTOSIZE_LIMIT と FP_NBIT_LOOKUP_MB では、暗黙的な NBit FP カラムの最大サイズが設定されます。これらの値を超える DML 操作では、暗黙的な NBit のロールオーバーにおけるディクショナリサイズ(値とカウント)を設定する FP_NBIT_ROLLOVER_MAX_MB 制限がチェックされます。

- ディクショナリの合計サイズが FP_NBIT_ROLLOVER_MAX_MB を超えていなければ、NBit カラムは Flat FP にロールオーバーされます。
- ディクショナリサイズが FP_NBIT_ROLLOVER_MAX_MB の制限を超えている場合、FP_NBIT_ENFORCE_LIMITS='ON' であれば、DML 操作はエラーをスローし、ロールバックします。
- ディクショナリサイズが FP_NBIT_ROLLOVER_MAX_MB の制限を超えている場合、FP_NBIT_ENFORCE_LIMITS='OFF' (デフォルト) であれば、DML 操作はエラーをスローし、NBit ディクショナリは増え続けます。
- FP_NBIT_ROLLOVER_MAX_MB='0' の場合、NBit カラムは Flat FP にロールオーバーされます。

sp_iqindexmetadata はフラット FP カラムまたは NBit FP カラムについての詳細情報を返します。sp_iqrebuildindex は、明示的または暗黙的な NBit FP カラムの制限を変更したり、デフォルトカラムインデックスをフラット FP または NBit FP として再フォーマットできます。

その他の情報

- 『リファレンス：ビルディングブロック、テーブル、およびプロシージャ』 > 「システムプロシージャ」 > 「システムストアプロシージャのアルファベット順リスト」 > 「sp_iqrebuildindex」

- 『リファレンス：ビルディングブロック、テーブル、およびプロシージャ』 > 「システムプロシージャ」 > 「システムストアードプロシージャのアルファベット順リスト」 > 「`sp_iqindexmetadata`」

FP_PREDICATE_WORKUNIT_PAGES オプション

デフォルトインデックスで使用される並列性の度合いを指定します。

指定できる値
整数値

デフォルト値
200

スコープ
オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考
デフォルトインデックスの作成時には SUM、RANGE、MIN、MAX、COUNT DISTINCT などの述部が並列的に計算されます。
FP_PREDICATE_WORKUNIT_PAGES は、各スレッドが操作するページ数を指定することで、並列性の度合いを制御します。並列度を上げたい場合は、このオプションの値を小さくします。

FPL_EXPRESSION_MEMORY_KB オプション

列挙記憶領域を持つカラムに対する関数式を含むクエリの最適化に使用するメモリを制御します。

指定できる値
0 - 20000

デフォルト値
1024 キロバイト

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

FPL_EXPRESSION_MEMORY_KB は、列挙記憶領域を持つカラムに対する関数式を含むクエリの最適化に使用するメモリを制御します。このオプションによって、DBA は、この最適化で使用されるメモリ量を制限し、他の SAP Sybase IQ メモリ要件（キャッシュなど）とのバランスを取ることができます。このオプションを 0 に設定すると、最適化が無効になります。

GARRAY_FILL_FACTOR_PERCENT オプション

既存のグループに対するインクリメンタルな挿入を今後行うために予約しておく領域が各 HG garray ページに占める割合を指定します。

指定できる値

0 - 1000

デフォルト値

25

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

garray は、各グループを増大することにより、この値によって設定された空き領域の埋め込みを含めようとしています。この領域は、既存のインデックスグループに追加するロー用に使用されます。

HG インデックスに対して、グループごとに決められた大きさの記憶領域を確保できます(ここで、グループは同じ値を持つローのまとまりとして定義されています)。領域を確保するには追加のディスク領域が必要となりますが、HG インデックスへのインクリメンタルな挿入処理のパフォーマンスが向上します。

HG インデックスに対するインクリメンタルな挿入が今後も発生し、またそれによってできる新しいローの値が以前のインデックスにすでに存在する場合は、このオプションに 0 以外の値を設定すると、インクリメンタルな挿入処理のパフォーマンスが向上する可能性があります。

インデックスをインクリメンタルに更新する予定がない場合は、このオプションの値を小さくして、ディスク領域を節約できます。

参照：

- GARRAY_PAGE_SPLIT_PAD_PERCENT オプション (678 ページ)

GARRAY_INSERT_PREFETCH_SIZE オプション

プリフェッチに使用するページ数を指定します。

指定できる値

0 - 100

デフォルト値

3

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

このオプションでは、HG インデックスを持つカラムに対する挿入処理中に先読みを行うデータベースページ数を定義します。

テクニカルサポートから指示があった場合を除いて、このオプションは設定しないでください。

参照：

- GARRAY_FILL_FACTOR_PERCENT オプション (676 ページ)

GARRAY_PAGE_SPLIT_PAD_PERCENT オプション

garray に対するページ分割での 1 ページあたりのフィルファクタを決定し、今後のインクリメンタルな挿入用に予約しておく領域が各 HG garray ページに占める割合を指定します。

指定できる値

0 - 100

デフォルト値

25

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

garray ページの分割は、このパーセンテージを空のままにしようとしています。この領域は、新しいインデックスグループに追加するロー用で使用されます。

HG インデックスは、ページレベルで記憶領域を予約できます。この記憶領域は、追加のローが挿入されたときに新しいグループに割り当てることができます。領域を確保するには追加のディスク領域が必要となりますが、HG インデックスへのインクリメンタルな挿入処理のパフォーマンスが向上します。

インクリメンタルな挿入を HG インデックスに行う予定があり、それによってできる新しいローの値が以前のインデックスにまだ存在しない場合は、

GARRAY_PAGE_SPLIT_PAD_PERCENT に 0 以外の値を設定すると、インクリメンタルな挿入処理のパフォーマンスが向上する可能性があります。

インデックスをインクリメンタルに更新する予定がない場合は、このオプションの値を小さくして、ディスク領域を節約できます。

参照：

- **GARRAY_FILL_FACTOR_PERCENT** オプション (676 ページ)

GARRAY_RO_PREFETCH_SIZE オプション

プリフェッチに使用するページ数を指定します。

指定できる値

0 - 100

デフォルト値

10

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

このオプションでは、HG インデックスを持つカラムに対するクエリの処理中に先読みを行うデータベースページ数を定義します。

テクニカルサポートから指示があった場合を除いて、このオプションは設定しないでください。

HASH_PINNABLE_CACHE_PERCENT オプション

ハッシュオブジェクトが保持できるユーザの一時メモリの最大パーセンテージを制御します。

指定できる値

0 - 100

デフォルト値

20

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

HASH_PINNABLE_CACHE_PERCENT は、任意のハッシュオブジェクトがメモリ内に保持できる、ユーザの一時メモリ割り付けのパーセンテージを制御します。デフォルトは 20% ですが、複雑なクエリを処理する場合は 10% と低く設定し、大きな IN サブクエリのような、大きなハッシュオブジェクトを 1 つだけ実行する必要がある単純なクエリの場合は 50% と高く設定してください。

HASH_PINNABLE_CACHE_PERCENT は主に テクニカルサポートが使用します。この値を変更する場合は、細心の注意が必要です。さまざまなクエリへの影響を分析してから実行してください。

参照：

- BIT_VECTOR_PINNABLE_CACHE_PERCENT オプション (614 ページ)
- SORT_PINNABLE_CACHE_PERCENT オプション (773 ページ)

HASH_THRASHING_PERCENT オプション

ハッシュアルゴリズムを使ったクエリを含む文の実行中、文がロールバックされ、エラーメッセージが返されるまでに許容するハードディスク I/O の割合を指定します。

指定できる値

0 - 100

デフォルト値

10

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

ハッシュアルゴリズムを使ったクエリによって大量のハードディスク I/O (メモリからディスクへのページングバッファ) が発生すると、クエリのパフォーマンスが低下し、サーバのパフォーマンスにも悪影響が及ぶ可能性があります。

HASH_THRASHING_PERCENT は、文で使用できるハードディスク I/O の割合を制御します。この割合を超えると、文はロールバックされ、エラーメッセージが返されます。エラーメッセージのテキストは、Hash insert thrashing detected または Hash find thrashing detected のいずれかです。

HASH_THRASHING_PERCENT のデフォルト値は 10% です。この値を大きくするとロールバック発生までのディスクへのページング量が増え、この値を小さくするとページング量が減ります。

参照：

- HASH_PINNABLE_CACHE_PERCENT オプション (679 ページ)

HG_DELETE_METHOD オプション

HG インデックスの削除処理で使用するアルゴリズムを指定します。

指定できる値

0-3

デフォルト値

0

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場

データベースオプション

合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

このオプションでは、削除操作中に HG インデックスが使用するアルゴリズムを選択します。コストモデルは、適切な削除アルゴリズムを選択する際に、CPU 関連のコストと I/O 関連のコストを考慮します。コストモデルで考慮されるのは以下の要素です。

- 削除されるロー
- インデックスサイズ
- インデックスデータ型の幅
- インデックスデータのカーディナリティ
- 利用可能なテンポラリキャッシュ
- マシンに関連する I/O と CPU の特性
- 利用可能な CPU とスレッド
- 参照整合性のコスト

"small" メソッドを強制するには、このオプションを 1 に設定します。"large" メソッドを強制するには、このオプションを 2 に設定します。"midsize" メソッドを強制するには、このオプションを 3 に設定します。

HG_SEARCH_RANGE オプション

HG インデックス内の範囲述部の評価に使用する B ツリーページの最大数を指定します。

指定できる値
整数

デフォルト値
10

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

ほとんどのクエリに対して、このオプションはデフォルトの設定が適しています。

このオプションは、オプティマイザが範囲述部に使用される最良のインデックスの検索にかかる時間を効率よく制御します。高い値に設定すると、クエリがオプティマイザで費やす時間が増加する可能性があります。結果としては範囲述部を解決する最適なインデックスが選択されます。

HTTP_SESSION_TIMEOUT オプション

クライアントが HTTP セッションがタイムアウトになったと判断するまで待機する時間 (分単位) を指定します。

指定できる値

整数値 (0 ~ 525600)

デフォルト値

30

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。すぐに有効になります。

備考

このオプションは、Web サービスアプリケーションに対してさまざまなセッションタイムアウト制御を提供します。Web サービスアプリケーションは、HTTP セッションを所有する任意の要求内でタイムアウト値を変更できます。ただし、タイムアウト値を変更すると、HTTP セッションがタイムアウトになった場合に、それ以降にキューイングされた要求に影響する場合があります。Web アプリケーションには、存在しなくなった HTTP セッションへのアクセスをクライアントが試行しているかどうかを検出するロジックを含める必要があります。そのためには、**SessionCreateTime** 接続プロパティの値を調べて、タイムスタンプが有効であるかどうかを確認します。対象の HTTP 要求が現在の HTTP セッションに関連付

データベースオプション

けられていない場合は、**SessionCreateTime** 接続プロパティには空の文字列が設定されています。

IDENTITY_ENFORCE_UNIQUENESS オプション

すでにプライマリーキーになっている場合を除き、各 Identity/Autoincrement カラムにユニーク HG インデックスを作成します。

指定できる値

ON、OFF

デフォルト値

OFF

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

このオプションを ON に設定すると、今後作成される identity カラムに対して HG インデックスが作成されます。このインデックスは、テーブルを使用しているユーザが 1 人しかおらず、テーブルがローカルテンポラリテーブルでない場合に、そのユーザによってのみ削除できます。

参照：

- QUERY_PLAN オプション (748 ページ)

IDENTITY_INSERT オプション

ユーザが IDENTITY または AUTOINCREMENT カラムに値を挿入したり、これらのカラムを更新したりできるようにします。

指定できる値

= 'tablename'

デフォルト値

" (空の文字列)

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

注意： ユーザレベルのオプションを現在のオプションに対して設定すると、対応する temporary オプションも同じように設定されます。「データベースオプションのスコープと継続期間」を参照してください。

備考

IDENTITY_INSERT が設定されている場合、挿入と更新が有効になります。挿入または更新を行うカラムを特定するには、テーブルを指定する必要があります。テーブルの所有者でないユーザは、テーブル名を所有者の名前で修飾します。

IDENTITY カラムを持つテーブルは、**IDENTITY_INSERT** がテーブルに設定されていると削除できません。

例

たとえば、明示的な挿入をテーブル Employees で行うには、次のように指定します。

```
SET TEMPORARY OPTION IDENTITY_INSERT = 'DBA.Employees'
```

オプションをオフにするには、等号と空の文字列を指定します。

```
SET TEMPORARY OPTION IDENTITY_INSERT = ''
```

ユーザレベルオプションが temporary オプションに与える影響 (注意事項を参照してください) を説明するために、データベースに DBA として接続している場合に、次のコマンドを発行したと想定します。

```
SET OPTION IDENTITY_INSERT = 'Customers'
```

このオプションの値は、ユーザ DBA に対して Customers に設定され、また、現在の接続にも一時的に設定されます。その後、他のユーザが DBA としてデータベースに接続した場合も、**IDENTITY_INSERT** オプションの値は Customers になります。

参照：

- データベースオプションのスコープと継続期間 (587 ページ)
- QUERY_PLAN オプション (748 ページ)

IN_SUBQUERY_PREFERENCE オプション

IN サブクエリを処理する際に使用するアルゴリズムを選択します。

指定できる値

-3 ~ 3

デフォルト値

0

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

IN サブクエリを処理する際、IQ オプティマイザはいくつかのアルゴリズムから 1 つを選択します。このオプションを使用すると、オプティマイザが処理量をもとに決定した使用アルゴリズムを無効できます。クエリエンジンに対してアルゴリズムが適切かどうかを判断するための内部規則を無効にするものではありません。

IN_SUBQUERY_PREFERENCE は通常、内部のテストと、オプティマイザがうまく処理できないクエリの手動チューニングに利用されます。経験豊富な DBA のみ使用してください。このオプションを使用するのは、サブクエリによって生成されるローの数をおプティマイザが少なく見積もりすぎているために、ハッシュオブジェクトがスラッシングしている場合のみです。このオプションを設定する前に、まず、失われたインデックスや依存述部をチェックして、誤った見積もりを修正してください。

IN_SUBQUERY_PREFERENCE を設定するのは、オプティマイザに変更を加えることが適切である場合のみであるため、このオプションを設定する必要がある場合は、テクニカルサポートにご連絡ください。

表 16 : IN_SUBQUERY_PREFERENCE の有効な値

値	アクション
0	オプティマイザの選択に従う。
1	ソートベースの IN サブクエリを優先する。
2	垂直 IN サブクエリ (サブクエリがクエリプラン内のリーフノードの子) を優先する。
3	ハッシュベースの IN サブクエリを優先する。
-1	ソートベースの IN サブクエリを回避する。
-2	垂直 IN サブクエリを回避する。
-3	ハッシュベースの IN サブクエリを回避する。

INDEX_ADVISOR オプション

1つまたは複数のクエリのパフォーマンスを向上させる、追加のカラムインデックスを推奨するメッセージを生成します。

指定できる値

ON、OFF

デフォルト値

OFF

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できません。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

ON に設定されていると、インデックスアドバイザーは、クエリプランの一部として、または、クエリプランが無効の場合にメッセージログファイル内の独立したメッセージとして、推奨されるインデックスを表示します。これらのメッセージは、"Index Advisor:" という文字列で始まります。この文字列を検索すること

データベースオプション

で、メッセージファイルからこれらのメッセージをフィルタできます。出力は OWNER.TABLE.COLUMN 形式です。

インデックスアドバイスを累積するには、**INDEX_ADVISOR** と **INDEX_ADVISOR_MAX_ROWS** の両方を設定します。

注意： **INDEX_ADVISOR_MAX_ROWS** が ON に設定されている場合、インデックスアドバイスは独立したメッセージとしてメッセージファイルに書き込まれません。ただし、アドバイスはメッセージファイルのクエリプランに引き続き示されます。

表 17：インデックスアドバイザ

状況	推奨
HG、LF、HNG、DATE、TIME、または DATETIME インデックスが望ましい、単一カラム上のローカル述部	<index-type> インデックスを <col> カラムに追加することを推奨
LF または HG インデックスが有用と想定される単一カラムのジョインキー	LF または HG インデックスをジョインキー <col> に追加する
HG が存在するが、ユニークな HG または LF に変更可能な単一カラムの候補キーインデックス	ジョインキー <col> をユニークな LF または HG インデックスに変更する
ジョインキーに適切でないデータ型が割り当てられており、適切なデータ型で1つのカラムを再生成すると改善されると想定される場合	ジョインキー <col1> と <col2> を同じデータ型にする
LF または HG インデックスが有用であると想定されるサブクエリの述部のカラム	LF または HG インデックスをサブクエリのカラム <col> に追加する
LF または HG インデックスが有用であると想定されるグループ化カラム	LF または HG インデックスをグループ化カラム <col> に作成する
単一のテーブルでのカラム間の比較で、2つのカラムが同じデータ型で、CMP インデックスが推奨される場合	CMP インデックスを <col1>、<col2> に作成する
) カラムに LF または HG インデックスが存在し、許容される重複しない値の数から、1バイトまたは2バイトの FP インデックスへの FP の変換が推奨される場合	<col> を 'optimize storage=on' で再構築する
3バイト幅のデフォルトインデックスのルックアップをサポートする必要がある場合	FP インデックスを IQ UNIQUE 制約値 65537 で3バイト FP インデックスとして再構築する

インデックスを追加することでどれだけのクエリが改善されるか、また、インデックスを作成および維持するメリットがあるかどうかはユーザが判断する必要があります。場合によっては、推奨されたインデックスを追加することで、パフォーマンスが改善するかどうか、改善するとしたらどの程度かを判断できないこともあります。

たとえば、ジョインキーとして使われているカラムを例に考えてみます。SAP Sybase IQ は、クエリを実行するためのより適切で高速なクエリプランを生成するために、HG または LF インデックスによって提供されるメタデータを広範に使用します。HG インデックスまたは LF インデックスを持たないジョインカラムにこれらのインデックスを追加すると、IQ オプティマイザがより速いジョインプランを選択できる可能性が高くなりますが、インデックスを追加してクエリを再実行しないかぎり、新しいインデックスによってクエリのパフォーマンスが変わらないか向上するかを判断するのは非常に困難です。

例

クエリプランが OFF に設定されたインデックスアドバイザの出力

```
I. 03/30 14:18:45. 0000000002 Advice: Add HG or LF index
on DBA.ta.c1 Predicate: (ta2.c1 < BV(1))
```

クエリプランが ON に設定されたインデックスアドバイザの出力

注意： この方法では複数のクエリのインデックスアドバイザの情報が累計されるため、複数のクエリのアドバイスを中央のロケーションで一定期間にわたって追跡できます。

```
I. 03/30 14:53:24. 0000000008 [20535]: 6      ...#03: Leaf
I. 03/30 14:53:24. 0000000008 [20535]:      Table Name: tb
I. 03/30 14:53:24. 0000000008 [20535]:      Condition 1
(Invariant):
(tb.c3 =tb.c4)
I. 03/30 14:53:24. 0000000008 [20535]:      Condition 1 Index
Advisor:
Add a CMP index on DBA.tb (c3,c4)
```

参照：

- FP_LOOKUP_SIZE オプション (665 ページ)
- INDEX_ADVISOR_MAX_ROWS オプション (690 ページ)
- QUERY_PLAN オプション (748 ページ)

INDEX_ADVISOR_MAX_ROWS オプション

インデックスアドバイザーによって max_rows に格納されるユニークなアドバイスメッセージの最大数を設定します。

指定できる値

値	説明
0	インデックスアドバイザーの収集を無効にする最小値
4294967295	許可される最大値

デフォルト値

0

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

INDEX_ADVISOR_MAX_ROWS は、インデックスアドバイザーによって格納されるメッセージの数を制限します。指定した制限値に到達すると、**INDEX_ADVISOR** は新しいアドバイスの保存を停止します。ただし、既存のアドバイスメッセージのカウントとタイムスタンプの更新は続行します。

```
SET OPTION public.Index_Advisor_Max_Rows = max_rows;
```

参照：

- FP_LOOKUP_SIZE オプション (665 ページ)
- INDEX_ADVISOR オプション (687 ページ)

INDEX_PREFERENCE オプション

クエリ処理に使用するインデックスの選択を制御します。

指定できる値

値	アクション
0	オプティマイザの選択に従う。
1	LF インデックスを優先する。
2	HG インデックスを優先する。
3	HNG インデックスを優先する。
4	CMP インデックスを優先する。
5	デフォルトのインデックスを優先する。
6	WD インデックスを優先する。
8	DATE インデックスを優先する。
9	TIME インデックスを優先する。
10	DTTM インデックスを優先する。
-1	LF インデックスを回避する。
-2	HG インデックスを回避する。
-3	HNG インデックスを回避する。
-4	CMP インデックスを回避する。
-5	デフォルトのインデックスを回避する。
-6	WD インデックスを回避する。
-8	DATE インデックスを回避する。
-9	TIME インデックスを回避する。
-10	DTTM インデックスを回避する。

デフォルト値

0

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

通常、SAP Sybase IQ オプティマイザは、ローカルな WHERE 句の述部の処理など、1つの IQ インデックスの範囲内で実行できる操作を行うために、使用可能な最適なインデックスを選択します。INDEX_PREFERENCE は、テスト目的でオプティマイザの選択を無効にするために使用します。通常の使用では、このオプションの値を変更しないでください。

INFER_SUBQUERY_PREDICATES オプション

オプティマイザによる追加のサブクエリの述部の推測を制御します。

指定できる値

ON、OFF

デフォルト値

ON

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

INFER_SUBQUERY_PREDICATES は、単純な等号ジョイン述部の推移閉包によって、既存のサブクエリの述語から追加のサブクエリの述部をオプティマイザに推測させるかどうかを制御します。多くの場合、オプティマイザがこの推測を行ったほうがクエリの実行が速くなります。これには例外もあり、パフォーマンスが向上しない場合もあります。このオプションを使うことが環境において適切かどうかを確認する必要があります。

IQGOVERN_MAX_PRIORITY オプション

許容される **IQGOVERN_PRIORITY** 設定を制限します。

指定できる値

1-3

デフォルト値

2

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

許可される **IQGOVERN_PRIORITY** の設定を制限します。この設定は、ユーザのクエリが実行待ちのキューに入れられる順番に影響します。指定可能な値は、1 (優先度高)、2 (優先度中、デフォルト値)、3 (優先度低) です。ユーザが **IQGOVERN_MAX_PRIORITY** よりも大きい値を **IQGOVERN_PRIORITY** に指定すると、SAP Sybase IQ はエラーを返します。

参照：

- **IQGOVERN_PRIORITY** オプション (694 ページ)
- **IQGOVERN_PRIORITY_TIME** オプション (694 ページ)

IQGOVERN_PRIORITY オプション

-iqgovern キュー内で待機中の各クエリに優先度を割り当てます。

指定できる値

1 ~ 3

デフォルト値

2

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

ユーザのクエリが実行待ちのキューに入れられる順番を決定する値を割り当てます。指定可能な値は、1 (優先度高)、2 (優先度中、デフォルト値)、3 (優先度低) のいずれかです。このスイッチは、すべてのユーザが、一時的、ユーザごと、または public に指定できます。優先度の低いクエリは、より優先度の高いクエリがすべて実行されるまで実行されません。

このオプションは、IQGOVERN_MAX_PRIORITY オプションのユーザごとまたはグループごとの値によって制限されます。

参照：

- IQGOVERN_MAX_PRIORITY オプション (693 ページ)
- IQGOVERN_PRIORITY_TIME オプション (694 ページ)

IQGOVERN_PRIORITY_TIME オプション

高優先度のクエリが実行までにキューで待機する時間を制限します。

指定できる値

0 ~ 1,000,000 秒。IQGOVERN_MAX_PRIORITY よりも小さい値である必要があります。

デフォルト値
0 (無効)

スコープ
オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。すぐに有効になります。

備考
高優先度 (優先度 1) のクエリが実行までにキューで待機する時間を制限します。この時間を過ぎると、**-iqgovern** の設定で実行が許可されたクエリ数を超過しても、クエリが実行されます。指定できる値は、1 から 1,000,000 秒です。デフォルト値 (0) では、この機能が無効になります。**IQGOVERN_PRIORITY_TIME** は PUBLIC として設定する必要があります。

参照：

- IQGOVERN_MAX_PRIORITY オプション (693 ページ)
- IQGOVERN_PRIORITY オプション (694 ページ)

ISOLATION_LEVEL オプション

カタログストアテーブルのロック独立性レベルを制御します。

指定できる値

- 0 - ダーティリード、繰り返し不可能読み出し、幻ローが許可されます。
- 1 - ダーティリードは許可されません。繰り返し不可能読み出しと幻ローが許可されます。
- 2 - ダーティリードは許可されませんが、繰り返し可能読み出しが保証されません。幻ローは許可されます。
- 3 - 逐次化可能です。ダーティリードは許可されず、繰り返し可能読み出しは保証され、幻ローは許可されません。

デフォルト値
0

Open Client および JDBC 接続の場合は 1

スコープ
オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できません。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、

データベースオプション

そのユーザの **PUBLIC** 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、**SET ANY PUBLIC OPTION** システム権限が必要です。個々の接続または **PUBLIC** ロールに一時的に設定できます。すぐに有効になります。

備考

ISOLATION_LEVEL は、カタログストアのテーブルに対する独立性レベルを決定します。SAP Sybase IQ では、IQ ストア内のテーブルに常にレベル 3 を適用します。レベル 3 が ANSI のレベル 4 に相当します。

JAVA_LOCATION オプション

データベースの Java VM のパスを指定します。

指定できる値
文字列

デフォルト値
空の文字列

スコープ
オプションは、データベース (**PUBLIC**) レベルでのみ設定できます。

このオプションを設定するには、**SET ANY SYSTEM OPTION** システム権限が必要です。すぐに有効になります。

備考

デフォルトでは、このオプションには空の文字列が設定されます。この場合、データベースサーバは、**JAVA_HOME** 環境変数、パス、およびその他のロケーションで Java VM を検索します。

参照：

- **JAVA_VM_OPTIONS** オプション (696 ページ)

JAVA_VM_OPTIONS オプション

データベースサーバが Java VM を起動するときに使用するコマンドラインオプションを指定します。

指定できる値
文字列

デフォルト値
空の文字列

スコープ
オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。すぐに有効になります。

備考

JAVA_VM_OPTIONS では、**JAVA_LOCATION** オプションで指定された Java VM の起動時にデータベースサーバが使用するオプションが指定されます。これらの追加のオプションを使用して、Java VM をデバッグ用として、または UNIX プラットフォーム上でサービスとして実行するように設定できます。32 ビットモードではなく 64 ビットモードで Java VM を使用するには、追加のオプションが必要な場合があります。

参照：

- **JAVA_LOCATION** オプション (696 ページ)

JOIN_EXPANSION_FACTOR オプション

極端に複雑な状況において、最適化によるジョイン結果の見積もりをどの程度抑えるかを制御します。

指定できる値
1 - 100

デフォルト値
30

スコープ
オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

このオプションは、特定のジョインへの入力が最低でも 1 つの中間ジョインを経由しており、ジョインされるテーブルから同じローが複数コピーされる可能性がある場合に、ジョインオプティマイザによる見積もりサイズがどうなるかを制御します。

ゼロを指定した場合、オプティマイザは、中間拡張ジョインがある場合も、通常と同じ見積もり方法を使います。

この場合、予測されるジョインの結果サイズは最も楽観的に (小さく) なります。

100 を指定した場合、中間拡張ジョインがある場合のオプティマイザによる見積もりはかなり控えめになり、予測されるジョインの結果サイズはもっとも慎重に (大きく) なります。

通常、この値を変更する必要はありません。変更する場合は、**JOIN_EXPANSION_FACTOR** をテンポラリオプションまたはユーザオプションとして設定してください。

JOIN_OPTIMIZATION オプション

ジョイン順序の最適化を有効または無効にします。

指定できる値

ON、OFF

デフォルト値

ON

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

JOIN_OPTIMIZATION が ON の場合、SAP Sybase IQ はジョイン順序を最適化して中間結果とソートのサイズを小さくし、システム負荷のバランスをとります。この

オプションが OFF の場合、ジョイン順序は **SELECT** 文の **FROM** 句のテーブルの順序に従って決定されます。

JOIN_OPTIMIZATION は常に ON に設定してください。

JOIN_OPTIMIZATION は、ジョインの順序を制御しますが、テーブルの順序は制御しません。この差について、4つのテーブルがある次の **FROM** 句を例に説明します。

```
FROM A, B, C, D
```

この **FROM** 句は、デフォルトで左側が深いジョインプランを作成します。これは、明示的に表すと次のようになります。

```
FROM ((A, B), C), D)
```

JOIN_OPTIMIZATION が OFF の場合、一連のテーブルにおけるこれらのジョインの順序は **FROM** 句で指定されたとおりになります。つまり、A と B がまずジョインされ、その成果物がテーブル C にジョインされ、最後にこれがテーブル D にジョインされます。このオプションは、各ジョインの左右の深さは制御しません。

JOIN_OPTIMIZATION が OFF でも、オブティマイザは上記の **FROM** 句から次のようなジョインプランを作成します。

```
FROM ((C, (A, B)), D)
```

または

```
FROM ((B, A), C), D)
```

または

```
FROM (D, ((A, B), C))
```

いずれの場合でも、まず A と B がジョインされ、その成果物が C にジョインされ、最後にテーブル D にジョインされます。ジョインの順序は同じですが、テーブルが表示される順序が異なります。

通常、**JOIN_OPTIMIZATION** が OFF の場合は、意図したとおりの順序でジョインが行われるように、上の例のように **FROM** 句内でカッコを使用する必要があります。C と D をジョインしたものに A と B をジョインするには、カッコを使って次のように指定します。

```
FROM ((A, B), (C, D))
```

上記の **FROM** 句では、最初の例の **FROM** 句と同じ順序でテーブルが表示されていますが、異なる順序でジョインが行われます。

JOIN_OPTIMIZATION は、ジョインのパフォーマンスに関する隠れた問題を診断する場合、または少数の定義済みクエリを手動で最適化する場合にのみ OFF に設定してください。**JOIN_OPTIMIZATION** が OFF の場合、クエリで最大 128 個のテーブ

ルをジョインできますが、パフォーマンスに深刻な悪影響が及ぶ可能性があります。

警告！ JOIN_OPTIMIZATION を無効にすると、SAP Sybase IQ はジョインを含むクエリに最適なパフォーマンスを保証できません。クエリのパフォーマンスに関してユーザ自身が全面的に責任を負うことになります。

JOIN_PREFERENCE オプション

ジョイン処理で使用されるアルゴリズムを選択します。

指定できる値

値	アクション
0	オプティマイザの選択に従う。
1	ソート/マージを優先する。
2	ネストされたループを優先する。
3	ネストされたループのプッシュダウンを優先する。
4	ハッシュを優先する。
5	ハッシュプッシュダウンを優先する。
6	非対称ソート/マージのジョインを優先する。
7	ソート/マージのプッシュダウンを優先する。
8	非対称ソート/マージのプッシュダウンのジョインを優先する。
9	ジョインキーにハッシュ分割テーブルの全パーティションキーが含まれる場合、分割ハッシュのジョインを優先する。
10	ジョインキーにハッシュ分割テーブルの全パーティションキーが含まれる場合、分割ハッシュプッシュダウンのジョインを優先する。
11	ジョインキーにハッシュ分割テーブルの全パーティションキーが含まれる場合、分割ソート/マージのジョインを優先する。
12	ジョインキーにハッシュ分割テーブルの全パーティションキーが含まれる場合、分割ソート/マージプッシュダウンのジョインを優先する。
-1	ソート/マージを回避する。
-2	ネストされたループを回避する。
-3	ネストされたループのプッシュダウンを回避する。
-4	ハッシュを回避する。

値	アクション
-5	ハッシュプッシュダウンを回避する。
-6	非対称ソート/マージのジョインを回避する。
-7	ソート/マージのプッシュダウンを回避する。
-8	非対称ソート/マージのプッシュダウンのジョインを回避する。
-9	ジョインキーにハッシュ分割テーブルの全パーティションキーが含まれる場合、分割ハッシュのジョインを回避する。
-10	ジョインキーにハッシュ分割テーブルの全パーティションキーが含まれる場合、分割ハッシュプッシュダウンのジョインを回避する。
-11	ジョインキーにハッシュ分割テーブルの全パーティションキーが含まれる場合、分割ソート/マージのジョインを回避する。
-12	ジョインキーにハッシュ分割テーブルの全パーティションキーが含まれる場合、分割ソート/マージプッシュダウンのジョインを回避する。

デフォルト値

0

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

SAP Sybase IQ オプティマイザは、クエリ内のジョインを処理する場合に複数のアルゴリズムの選択肢から1つを選択します。JOIN_PREFERENCE を使用すると、オプティマイザが処理量をもとに決定した使用アルゴリズムを無効にできます。クエリエンジンに対してアルゴリズムが適切かどうかを判断するための内部規則を無効にするものではありません。0以外の値を設定すると、クエリ内のすべてのジョインに影響します。クエリ内のいくつかのジョインから1つを選択して修正することはできませんが、ジョイン条件ヒント文字列であれば可能です。

通常、このオプションは、内部的なテストまたはレポートクエリの調整に使用されます。また、経験豊富な DBA だけが使用してください。

単純な等号ジョイン述部に述部ヒントのタグを付けることができます。このヒントにより、まさにその1つのジョインのためにジョインの優先順位を指定できます。ローカルなジョインの優先順位が設定されたジョイン条件が、同じジョインに複数あり、しかもそれらのヒントの値が異なる場合、そのジョインに対するローカルな優先順位がすべて無視されます。ローカルなジョインの優先順位は、オプティマイザが選択したジョインの順序に影響を与えません。

次の例は、ハッシュジョインを要求します。

```
AND (T.X = 10 * R.x, 'J:4')
```

JOIN_SIMPLIFICATION_THRESHOLD オプション

ジョインオプティマイザの単純化が行われる前にジョインされるテーブルの最小数を指定します。

指定できる値

1 ~ 24

デフォルト値

12

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

クエリオプティマイザは、ルックアップテーブル (非選択的な次元テーブル) と有効な直積であるテーブルを別々に扱うことで、ジョインの順序の最適化を単純化します。単純化の後で、オプティマイザは **MAX_JOIN_ENUMERATION** によって指定されている範囲内で、残りのテーブルのジョインの順序を最適化します。

このオプションに **MAX_JOIN_ENUMERATION** の値よりも大きい値を指定しても何も起こりません。

このオプションに **MAX_JOIN_ENUMERATION** の値よりも小さい値を指定すると、大量のジョインを含むクエリの最適化にかかる時間を短縮できる可能性があります。ただし、オプティマイザが最適なジョインプランを見つけられなくなる可能性もあります。

この値を変更する場合は、**JOIN_SIMPLIFICATION_THRESHOLD** をテンポラリオプションまたはユーザオプションとして設定し、9 以上の値を指定してください。

参照：

- **MAX_JOIN_ENUMERATION** オプション (716 ページ)

LF_BITMAP_CACHE_KB オプション

LF インデックスへのロードに使用するメモリ量を指定します。

指定できる値

1-8

デフォルト値

4

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、**SET ANY PUBLIC OPTION** システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

LF_BITMAP_CACHE_KB では、LF インデックスへのロード時に重複しない各値に対して使用するヒープメモリの量 (KB 単位) を定義します。デフォルトは 4KB です。特定のテーブルのすべての LF インデックスの重複しない値の個数の合計が比較的大きい (10,000 超) 場合は、ヒープメモリの使用量が増加して、システムのページフォールトが発生し、ロードのパフォーマンスに影響する可能性があります。その場合は、**LF_BITMAP_CACHE_KB** の設定値を小さくしてください。

次の式では、ロード中に特定の LF インデックスで使用されるヒープメモリの量 (バイト単位) が計算されます。

```
Heap-memory-used = (lf_bitmap_cache_kb * 1024)
* lf-distinct-count-for-column
```

デフォルト値の4KBを指定した場合、重複しない値の数が1000のLFインデックスでロード時に最大4MBのヒープメモリを使用できます。

LOAD_ZEROLENGTH_ASNULL オプション

特定の条件での **LOAD** 文の動作を指定します。

指定できる値

ON、OFF

デフォルト値

OFF

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

このオプションは、次の条件下での **LOAD** 文の動作を指定します。

- データ型が CHAR、VARCHAR、LONG VARCHAR、BINARY、VARBINARY、または LONG BINARY のカラムに、長さゼロのデータ値を挿入する。
- 同じカラムへの NULL カラム指定 (NULL(ZEROS) や NULL(BLANKS) など) がある。

LOAD_ZEROLENGTH_ASNULL を ON にすると、これらの条件が満たされている場合に、長さゼロの値が NULL としてロードされます。

LOAD_ZEROLENGTH_ASNULL を OFF に設定すると、長さゼロの値は長さゼロとしてロードされます。ただし、これは **NON_ANSI_NULL_VARCHAR** オプションの設定に依存します。

参照：

- NON_ANSI_NULL_VARCHAR オプション (732 ページ)

- LOAD TABLE 文 (430 ページ)

LOG_CONNECT オプション

ユーザ接続のロギングを制御します。

指定できる値
ON、OFF

デフォルト値
ON

スコープ
オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SECURITY OPTION システム権限が必要です。すぐに有効になります。

備考
このオプションが ON の場合、ユーザが SAP Sybase IQ データベースに接続するたび、または接続を切断するたびに、IQ メッセージログ (.iqmsg ファイル) にメッセージが書き込まれます。

注意：ユーザの接続時にこのオプションが OFF (接続のロギングが無効) に設定されており、そのユーザが切断する前に ON にされた場合、メッセージログにはユーザの接続は記録されず、切断だけが記録されます。

LOG_CURSOR_OPERATIONS オプション

カーソル操作のログ動作を制御します。

指定できる値
ON、OFF

デフォルト値
OFF

スコープ
オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

データベースオプション

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

このオプションが ON の場合、カーソルを開いたり閉じたりするたびにメッセージが IQ メッセージログに書き込まれます。デフォルトは OFF で、通常はこのままにしてください。このオプションを ON にするのは、何か問題があって、テクニカルサポートにデバッグデータを送る必要があるときだけにしてください。

LOG_DEADLOCKS オプション

デッドロックレポートのオン/オフを制御します。

指定できる値

On、Off

デフォルト

Off

スコープ

オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。すぐに有効になります。

備考

このオプションを On に設定した場合、データベースサーバは、内部バッファ内のデッドロックに関する情報をログに記録します。バッファのサイズは、10000 バイトに固定されています。デッドロック情報は sa_report_deadlocks ストアドプロシージャを使用すると表示できます。このオプションを Off に設定すると、バッファの内容は保持されます。

デッドロックが発生すると、そのデッドロックに関係する接続のみの情報がレポートされます。接続のレポート順序は、どの接続がどの行を待っているかに応じて決定されます。スレッドデッドロックの場合、すべての接続の情報がレポートされます。

デッドロックレポートがオンになっているときは、デッドロックの発生時に Deadlock システムイベントを使用してアクションを実行することもできます。

LOGIN_MODE オプション

データベースの統合化ログインの使用を制御します。

指定できる値

- **Standard** – デフォルト設定。統合化ログインは使用できません。統合化ログインを使用して接続しようとする、エラーが発生します。
- **Mixed** – 統合化ログインと標準ログインの両方を使用できます。
- **Integrated** – データベースへのログインはすべて、統合化ログインを使用して実行する必要があります。
- **Kerberos** – データベースへのログインはすべて、Kerberos ログインを使用して実行する必要があります。
- **LDAPUA** – データベースへのログインはすべて、LDAP ログインを使用して実行する必要があります。

注意： Mixed は、「Standard,Integrated」と同じになります。

デフォルト値

Standard

スコープ

オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SECURITY OPTION システム権限が必要です。すぐに有効になります。

備考

値では大文字と小文字が区別されません。

警告！

- 混合環境で **LOGIN_MODE** を 1 つのモード (Integrated のみ、LDAPUA のみなど) に制限すると、対応するログインマッピングが付与されたユーザのみに接続が制限されます。その他の方法を使用して接続しようとする、エラーが発生します。唯一の例外は、完全な管理権 (SYS_AUTH_DBA_ROLE または SYS_AUTH_SSO_ROLE) が付与されたユーザです。
 - **LOGIN_MODE** を LDAPUA のみに制限すると、LDAPUA が許可されるユーザポリシーまたはログインポリシーが存在しない場合に、接続可能なユーザが存在しない設定が発生する可能性があります。この状況から復旧するには、**start_iq** ユーティリティでコマンドラインスイッチ **-al user-id-list** を指定してください。
-

LOGIN_PROCEDURE オプション

起動時の接続互換性オプションを設定するログインプロシージャを指定します。

指定できる値
文字列

デフォルト値
sp_login_environment システムプロシージャ

スコープ
オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY SECURITY OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考
接続互換性オプションの初期設定は、**LOGIN_PROCEDURE** オプションを使用して制御します。このオプションは、接続が有効であることを確認するすべてのチェックの終了後に呼び出されます。**LOGIN_PROCEDURE** には、ユーザが接続したときに実行するストアードプロシージャを指定します。デフォルト設定では、**sp_login_environment** システムストアードプロシージャが使用されます。異なるストアードプロシージャを指定できます。**LOGIN_PROCEDURE** オプションに指定したプロシージャは、イベント接続に対しては実行されません。

sp_login_environment は、接続が TDS によって行われたかどうかを確認します。TDS によって接続が行われている場合は、**sp_login_environment** が **sp_tsq_environment** プロシージャを呼び出して、いくつかのオプションに現在の接続に合った新しいデフォルト値を設定します。

参照：

- オプションの初期設定 (591 ページ)

MAIN_RESERVED_DBSPACE_MB オプション

SAP Sybase IQ が IQ メインストアに予約する領域の量を制御します。

指定できる値

200 またはそれ以上の整数 (メガバイト)

デフォルト値

200; SAP Sybase IQ は、IQ_SYSTEM_MAIN の最新の読み書きファイルの 1% ~ 50% の領域を予約します。

スコープ

オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。すぐに有効になります。

備考

MAIN_RESERVED_DBSPACE_MB を使用すると、セーブポイントの解放操作、コミット操作、チェックポイント操作で使用される小さいが重要なデータ構造体用に SAP Sybase IQ が IQ メインストア内に確保する領域の量を制御できます。運用データベースでは、この値を 200MB ~ 1GB、または少なくとも IQ_SYSTEM_MAIN のサイズの 20% に設定します。IQ ページサイズや同時接続数が大きくなればなるほど、より多くの領域を予約する必要があります。

予約領域サイズは、IQ_SYSTEM_MAIN の最新読み書きファイルの 1% ~ 50% の領域として計算されます。

実際の DB 領域のサイズが **MAIN_RESERVED_DBSPACE_MB** の値の 2 倍に満たない場合、SAP Sybase IQ は **MAIN_RESERVED_DBSPACE_MB** オプションを無視します。100MB 未満の DB 領域 (デモデータベースなど) では、使用可能な領域の半分が予約されることがあります。

MAX_CARTESIAN_RESULT オプション

直積ジョインからのローの数を制限します。

指定できる値

整数値

デフォルト値

100000000

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

MAX_CARTESIAN_RESULT は、直積ジョインを含むクエリからの結果ロー (通常、クエリ生成時に 1 つ以上のジョイン条件が見つからなかったという結果) の数を制限します。この制限以下の予想結果を持つその直積ジョインのクエリプランを SAP Sybase IQ が見つけることができない場合、クエリは拒否され、エラーが返されます。**MAX_CARTESIAN_RESULT** を 0 に設定すると、直積ジョインの結果ロー数のチェックが無効になります。

MAX_CLIENT_NUMERIC_PRECISION オプション

クライアントに送信される numeric データの最大精度を制御します。

指定できる値

0 - 126

デフォルト値

0

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

SAP Sybase IQ は、計算を行うとき、精度を保証するためにデータ型を適切なサイズに拡大します。拡大されたデータ型は、Open Client および一部の ODBC アプリケーションで正しく処理できるサイズを超えることがあります。

MAX_CLIENT_NUMERIC_PRECISION がゼロ以外の値の場合、SAP Sybase IQ は numeric の結果カラムがこの値を超過していないかどうかチェックします。結果カラムが **MAX_CUBE_RESULT** の制限をオーバーしており、SAP Sybase IQ が指定された精度にキャストすることもできない場合、クエリは次のようなエラーを返します。

```
Data Exception - data type conversion is not possible %1 SQLCODE = -1001006
```

注意： SQL Anywhere では、数値関数でサポートされている最大値は 255 です。数値関数の精度がサポートされている最大値を超えている場合、次のエラーが表示されます。「関数 '_funcname' の結果のデータ型が、サポートされている最大数値精度の 255 を超えています。数値関数の精度に適切な値を設定してください。'location'」

参照：

- MAX_CLIENT_NUMERIC_SCALE オプション (711 ページ)
- PRECISION オプション (742 ページ)

MAX_CLIENT_NUMERIC_SCALE オプション

クライアントに送信される numeric データの最大位取りを制御します。

指定できる値

0 - 126

デフォルト値

0

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

データベースオプション

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

SAP Sybase IQ は、計算を行うとき、精度を保証するためにデータ型を適切な位取りとサイズに拡大します。拡大されたデータ型は、最初に定義されたデータのサイズよりも大きくなる可能性があります。このオプションを設定して、numeric の結果の位取りを指定します。

掛け算、割り算、足し算、引き算、集合関数はすべて、結果が最大精度および最大位取りを超える可能性があります。

たとえば、DECIMAL(88,2) と DECIMAL(59,2) を掛けると、結果には DECIMAL(147,4) が必要です。MAX_CLIENT_NUMERIC_PRECISION が 126 の場合、126 桁のみが結果に残ります。MAX_CLIENT_NUMERIC_SCALE が 4 の場合、結果は DECIMAL(126,4) として返されます。MAX_CLIENT_NUMERIC_SCALE が 2 の場合、結果は DECIMAL(126,2) として返されます。どちらの場合も、オーバフローが起こる可能性があります。

参照：

- MAX_CLIENT_NUMERIC_PRECISION オプション (710 ページ)
- SCALE オプション (769 ページ)

MAX_CUBE_RESULT オプション

IQ オプティマイザが **GROUP BY CUBE** 操作を行う場合の対象とするロー数の最大値を設定します。

指定できる値

0 - 4294967295

デフォルト値

10000000

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

クエリプランを生成するとき、IQ オプティマイザは、**GROUP BY CUBE** ハッシュ操作で生成されるグループの合計数を見積もります。IQ オプティマイザは、**GROUP BY CUBE** 操作にハッシュアルゴリズムを使用します。このオプションは、実行可能なハッシュアルゴリズムに対してオプティマイザが見積もるロー数に上限を設定します。実際のロー数が MAX_CUBE_RESULT の値を超える場合、オプティマイザはクエリの処理を中止し、「予測数 : *nnn* は、GROUP BY CUBE または ROLLUP の MAX_CUBE_RESULT を超えています。」(*nnn* は IQ オプティマイザが見積もった数値) というエラーメッセージを返します。

MAX_CUBE_RESULT をゼロに設定すると、デフォルト値が無効になります。このオプションがゼロに設定されている場合、IQ オプティマイザはローの上限をチェックせずにクエリの実行を許可します。**MAX_CUBE_RESULT** をゼロに設定するとクエリが失敗する可能性があるため、この設定はおすすめしません。

MAX_CURSOR_COUNT オプション

各接続が一度に使用できるカーソルの最大数を制限するリソースガバナを指定します。

指定できる値
整数

デフォルト値
50

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

DBA は指定されたリソースガバナを使用して、ユーザが1つの接続で使用できるカーソルの数を制限できます。1つの接続に対してこの制限を超える操作を行うと、制限を超過していることを示すエラーを生成します。

ある接続でストアドプロシージャが実行される場合、それはプロシージャの所有者のパーミッションに従って実行されます。ただし、プロシージャが使用するリソースは現在の接続から割り当てられます。

リソースの制限を削除するには、**MAX_CURSOR_COUNT** を 0 (ゼロ) に設定します。

MAX_HASH_ROWS オプション

IQ オプティマイザがハッシュアルゴリズムを使用する際の対象とするロー数の最大値を設定します。

指定できる値

1 ~ 4294967295 の整数

デフォルト値

2500000

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、**SET ANY PUBLIC OPTION** システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

クエリプランを生成する際、IQ オプティマイザには、クエリの特典部分の処理用アルゴリズムとして複数の選択肢 (ハッシュ、ソート、インデックス) があります。選択されるアルゴリズムは、処理されるロー数の予測値、または検討の対象とされているクエリの特典部分に含まれているロー数の予測値によって変わります。このオプションは、ハッシュアルゴリズムを検討する際の予想ロー数の最大値を設定するのに使用されます。

たとえば、2つのテーブル間にジョインがあり、両方のテーブルからジョインに入力されるロー数が **MAX_HASH_ROWS** の値を超える場合、オプティマイザはハッ

シュジョインを選択肢から外します。1 ユーザに対し 50MB 以上のテンポラリバッファキャッシュ領域があるシステムでは、このオプションにより大きな値を設定することもできます。

MAX_IQ_THREADS_PER_CONNECTION オプション

各接続のスレッド数を制御します。

指定できる値
3 - 10000

デフォルト値
144

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

説明

接続上で実行されるコマンドが使用するスレッド数 (システム リソースの量) を制限します。大部分のアプリケーションには、デフォルト値を使用します。

MAX_IQ_THREADS_PER_TEAM オプション

接続内で実行される 1 つの操作 (カラム上の LIKE 述部など) を実行するために割り当てられるスレッドの数を制御します。

指定できる値
1 - 10000

デフォルト値
144

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、

そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

単一の操作に割り当てるスレッド数を制限できます(したがって、システムリソースの量を制限できます)。この接続で同時に実行されるすべてのチームにおける合計数は、関連オプション **MAX_IQ_THREADS_PER_CONNECTION** によって制限されます。大部分のアプリケーションには、デフォルト値を使用します。

参照：

- **MAX_IQ_THREADS_PER_CONNECTION** オプション (715 ページ)

MAX_JOIN_ENUMERATION オプション

オプティマイザによる単純化の適用後、ジョイン順を最適化するテーブルの最大数を制御します。

指定できる値

1 ~ 32

各 FROM 句に指定できるテーブル数は、最大 64 に制限されます。実際には、FROM 句内のテーブル数の実効制限値 (テーブル間のジョイン関係の複雑さなどに基づく) は、通常、この値よりもはるかに小さくなります。実効制限値は

MAX_JOIN_ENUMERATION の設定の制約を受けます。オプティマイザは、FROM 句内のジョイン関係のセットを単純化しようとします。このような単純化を行っても、同時に考慮すべきジョインセットを **MAX_JOIN_ENUMERATION** の現在の設定以下に削減できない場合、クエリはエラーを返します。

警告！ **MAX_JOIN_ENUMERATION** をデフォルト値の 15 より大きくする場合は注意が必要です。特に、枝分かれの多いジョイン関係を伴うクエリの場合、オプティマイザが必要とする処理時間が劇的に増加する可能性があります。リニアチェーンのジョイン関係のみを使用するクエリであっても、**MAX_JOIN_ENUMERATION** を 32 に設定すると、最適化にかなりの時間がかかることがあります。

デフォルト値

15

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

クエリオプティマイザは、ルックアップテーブル (非選択的な次元テーブル) と有効な直積であるテーブルを別々に扱うことで、ジョインの順序の最適化を単純化します。単純化の後で、オプティマイザは MAX_JOIN_ENUMERATION によって指定されている範囲内で、残りのテーブルのジョインの順序を最適化します。この制限をオーバーすると、エラーが発生し、クエリは拒否されます。ユーザはクエリを単純化するか、上限を上げることができます。

通常、この値を変更する必要はありません。変更する場合は、MAX_JOIN_ENUMERATION をテンポラリーオプションまたはユーザオプションとして設定してください。

MAX_PARTITIONED_HASH_MB オプション

ハッシュ分割されたハッシュベースのクエリ演算子に対して利用可能であるとオプティマイザがみなすことができるテンポラリーキャッシュ領域の容量の上限をメガバイトで設定します。

指定可能な値

0 ~ 4294967295 の整数

デフォルト

0

スコープ

個々の接続、ユーザ、または PUBLIC グループに対して一時的に設定できます。このオプションを設定する際に必要なシステム権限はありません。このオプションはすぐに有効になります。

説明

クエリプランの生成時に IQ オプティマイザは、クエリの特定の部分を処理する際
のアルゴリズムを複数のアルゴリズムのなかから選択することがあります。この
決定は、多くの場合、クエリの該当部分を処理するために必要なテンポラリ
キャッシュ領域の予測量と、現在利用可能なテンポラリキャッシュの量に左右さ
れます。このオプションでは、ハッシュ分割されたハッシュベースのアルゴリ
ズムを考慮することが可能な演算子に対するテンポラリ領域の使用予測量の上限を
設定します。

デフォルト値 0 は、ハードな上限はなく、したがってオプティマイザの選択を制
限する要素は、現在利用可能なテンポラリキャッシュ、現在のアクティブなユー
ザ接続数、および HASH_PINNABLE_PERCENT オプション設定のみであることを
表します。

このオプションはオプティマイザによるアルゴリズム選択決定にのみ影響します。
一部の状況では、実行時の使用量がこの制限を超えることがあります。

MAX_PREFIX_PER_CONTAINS_PHRASE オプション

テキスト検索式で許可するプレフィクス単語の数を指定します。

指定可能な値

0 ~ 300

デフォルト

1

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できま
す。データベースレベルで設定した場合、値は新しいユーザのデフォルト値にな
りますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、
そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場
合は、システム権限は必要ありません。自分以外のユーザのオプションをデー
タベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要
です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効に
なります。

備考

TEXT インデックスを使用し、全文検索を実行するには、非構造化データ分析オプ
ションのライセンスを取得している必要があります。

MAX_QUERY_PARALLELISM オプション

GROUP BY 操作の並列実行数と、**UNION** の分岐の数に上限を設定します。

指定できる値

CPU 数よりも少ないか、多いか、これに等しい整数

デフォルト値

64

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

このパラメータは、オプティマイザによってクエリ演算の並列度が制限される上限値を設定します。これは、多くのクエリジョイン、**GROUP BY**、**UNION**、**ORDER BY**、他のクエリ操作の CPU 使用率に影響を与えます。

64 以上の CPU コアを持つシステムでは、システムの CPU コアの総数(最大 512)以下の大きな値が有効です。いくつかの値を試し、システムおよびクエリでのこのパラメータの最適な値を決定できます。

64 またはそれ以下の CPU コアを持つシステムでは、過度なシステム時間が検出されないかぎり、この値を下げる必要はありません。検出された場合、この値を下げて、CPU のシステム時間が減少し、クエリの応答時間およびシステム全体のスループットが向上するかを判断します。

MAX_QUERY_TIME オプション

オプティマイザが長すぎるクエリを拒否できるよう、タイムリミットを設定します。

指定できる値

0 ~ 2³² - 1 分

デフォルト値

0 (無効)

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

クエリが **MAX_QUERY_TIME** の設定よりも長い時間実行されると、SAP Sybase IQ はクエリを中止して、ユーザと IQ メッセージファイルにメッセージを送信します。次に例を示します。

```
The operation has been cancelled -- Max_Query_Time exceeded.
```

MAX_QUERY_TIME はクエリのみにも適用され、データベースの内容を変更する SQL 文には適用されません。

MAX_STATEMENT_COUNT オプション

1つの接続で一度に使用できる準備文の最大数を制限するリソースガバナを指定します。

指定できる値

整数

デフォルト値

100

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

指定したリソースガバナを使用して、DBA はユーザが 1 つの接続で使用できる準備文の数を制限できます。1 つの接続に対してこの制限を超える操作を行うと、制限を超過していることを示すエラーを生成します。

ある接続でストアードプロシージャが実行される場合、それはプロシージャの所有者のパーミッションに従って実行されます。ただし、プロシージャが使用するリソースは現在の接続から割り当てられます。

リソースの制限を削除するには、MAX_STATEMENT_COUNT を 0 (ゼロ) に設定します。

MAX_TEMP_SPACE_PER_CONNECTION オプション

接続ごとに使用されるテンポラリストアの領域を制限します。

指定できる値
整数 (MB の数)

デフォルト値
0 (テンポラリストアの使用には制限なし)

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

DBA はこのオプションで接続ごとの領域を制御することによって、ロードとクエリの両方の領域を管理できます。接続が MAX_TEMP_SPACE_PER_CONNECTION で指定された実行時クォータを超える場合、SAP Sybase IQ は現在の文をロールバックし、IQ メッセージファイルまたはクライアントユーザに次のメッセージを返します。

データベースオプション

```
The current operation has been cancelled:  
Max_Temp_Space_Per_Connection exceeded
```

バッファキャッシュがいっぱいになる条件には、読み書きエラー、メインまたはテンポラリ領域の不足、メモリ不足が含まれます。SAP Sybase IQ は、これらの状況で最初に検出されたエラーを返す可能性があり、DBA は適切な解決法を決定する必要があります。

分散クエリ処理トランザクションでは、SAP Sybase IQ は、**QUERY_TEMP_SPACE_LIMIT** オプションと **MAX_TEMP_SPACE_PER_CONNECTION** オプションのそれぞれの設定値を共有テンポラリストアに適用して、分散クエリに参加するすべてのノードが使用する共有とローカルのテンポラリ領域の総量を制限します。この制限によって、参加ノードの数にかかわらず、どのクエリも (DB 領域 `IQ_SYSTEM_TEMP` および `IQ_SHARED_TEMP` の) テンポラリ領域総量制限を超えることができません。

たとえば、制限値が 100 で参加する 4 つのノードがテンポラリ領域にそれぞれ 25 ユニットを使用する場合、このクエリは制限内となります。ノードにより使用される領域の総量が 100 を超える場合、クエリはロールバックされます。

例

次の文はすべての接続に対して 500GB の制限を設定します。

```
SET OPTION  
PUBLIC.MAX_TEMP_SPACE_PER_CONNECTION = 512000
```

次の文はすべての接続に対して 10TB の制限を設定します。

```
SET OPTION  
PUBLIC.MAX_TEMP_SPACE_PER_CONNECTION = 10485760
```

次の文はユーザ `wilson` に対して 5000MB の制限を設定します。

```
SET OPTION  
wilson.MAX_TEMP_SPACE_PER_CONNECTION = 5000
```

参照：

- [QUERY_TEMP_SPACE_LIMIT オプション \(757 ページ\)](#)

MINIMIZE_STORAGE オプション

SAP Sybase IQ 15 データベースに新たに作成されるカラムでのディスク領域の使用を最小限に抑えます。

指定できる値

ON、OFF

デフォルト値

OFF

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

依存性

MINIMIZE_STORAGE は、**FP_NBIT_IQ15_COMPATIBILITY='ON'** が設定されている実行中のデータベースに適用されます。**FP_NBIT_IQ15_COMPATIBILITY='OFF'** の場合、データベースエンジンはこのオプションを無視します。

備考

FP_NBIT_IQ15_COMPATIBILITY='ON' の場合、**MINIMIZE_STORAGE** を有効に設定すると、可能な場合ディスク領域の使用をローごとに 1 バイトに抑えることで、新しいカラムの格納が最適化されます。デフォルトでは、このオプションは、PUBLIC ロールに対して 'OFF' に設定され、新たに作成されるどのカラムでも格納の特別な最適化は行われません。PUBLIC で **MINIMIZE_STORAGE='OFF'** になっていても、このオプションをテンポラリユーザオプションとして 'ON' に設定した場合、そのユーザ ID によって作成される新たなカラムの格納に 1 バイトの領域が使用されます。

SAP Sybase IQ 15.x データベースでは、**MINIMIZE_STORAGE=ON** の設定は、すべての新しいカラムで IQ UNIQUE 255 句を指定するのと同じですが、後者の場合、データ型によっては 1 バイトの格納領域では足りないことがあります。

MINIMIZE_STORAGE='ON' に設定した場合、ユニークな値の数が 65536 を超えるカラムを除いて、IQ UNIQUE を指定する必要はありません。

メインメモリとカラム数の比が大きい場合は、**MINIMIZE_STORAGE='ON'** に設定すると有益です。それ以外の場合は、このオプションを OFF にしたほうが、新しいカラムの格納が効果的に行われます。

重要： **FP_NBIT_IQ15_COMPATIBILITY='ON'** が設定された状態でデータベースを実行するのは避けてください。SAP Sybase IQ 15 のランタイム動作はすべて、SAP Sybase IQ 16.0 インタフェースで実行可能です。

参照：

- FP_LOOKUP_SIZE オプション (665 ページ)
- INDEX_ADVISOR オプション (687 ページ)
- FP_LOOKUP_SIZE_PPM オプション (666 ページ)
- FP_NBIT_IQ15_COMPATIBILITY オプション (670 ページ)

MIN_PASSWORD_LENGTH オプション

データベースの新しいパスワードの長さの最小値を設定します。

指定できる値

0 以上の整数

値はバイト単位です。シングルバイト文字セットの場合、これは文字数と同じになります。

デフォルト値

3 文字

スコープ

オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SECURITY OPTION システム権限が必要です。すぐに有効になります。

備考

このオプションを使用すると、すべての新しいパスワードの長さに最小値が設定され、セキュリティが強化されます。既存のパスワードには影響しません。

例

新しいパスワードの最小長を 6 バイトに設定します。

```
SET OPTION PUBLIC.MIN_PASSWORD_LENGTH = 6
```

MIN_ROLE_ADMIN オプション

すべてのロールについて、必要な管理者の最小数を設定します。

指定可能な値

1 - 10

デフォルト

1

スコープ

オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SECURITY OPTION システム権限が必要です。すぐに有効になります。

備考

このオプションは、すべてのロールについて、必要な管理者の最小数を設定します。この値は、ロール全体に対するロール管理者の最小数ではなく、各ロールに対するロール管理者の最小数に適用されます。この値は、ロールまたはユーザを削除する際に、残りのユーザとロールの管理に十分なシステム権限を持つユーザとロールがいなくなるというシナリオが発生しないことを保証します。

MONITOR_OUTPUT_DIRECTORY オプション

IQ バッファキャッシュモニタの出力ファイルの場所を制御します。

指定できる値
文字列

デフォルト値
データベースと同じディレクトリ

スコープ
オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

MONITOR_OUTPUT_DIRECTORY は、モニタ対象や使用するモニタモードに関係なく、IQ モニタ出力ファイルの作成先のディレクトリを制御します。モニタの起動に使用されるダミーテーブルには、テンポラリテーブルと永久テーブルのどちらも使用できます。任意の物理マシン上のディレクトリを指定できます。

すべてのモニタの出力ファイルは、モニタが実行されている期間使用されます。接続が無効になった後は使用されません。出力ファイルは、モニタリングの実行が停止した後も残ります。1つの接続で同時に実行できるパフォーマンスモニタの数は、最大2つ(メインバッファキャッシュ用に1つ、テンポラリバッファキャッシュ用に1つ)です。1つの接続で同じモニタを連続して何回でも実行できます。

データベースオプション

DBA は、PUBLIC 設定を使用して、すべてのモニタ出力を同じディレクトリに設定することも、ユーザごとに異なるディレクトリに設定することもできます。

例

この例では、モニタ出力用のテンポラリテーブルを宣言し、その配置場所を設定し、そこにファイルを送信するモニタをメインおよびテンポラリバッファキャッシュに対して開始する方法を示します。

注意： この例では、出力ディレクトリ文字列が `"/tmp"` と `"tmp/"` の両方に設定されています。末尾のスラッシュ (`/`) は正しく、インタフェースによってサポートされます。この例は、バッファキャッシュモニタが永久テーブルを必要とせず、テンポラリテーブルを使用できることを示します。

```
declare local temporary table dummy_monitor (dummy_column integer)

set option Monitor_Output_Directory = "/tmp"
iq utilities main into dummy_monitor start monitor '-debug -interval
2'

set option Monitor_Output_Directory = "tmp/"

iq utilities private into dummy_monitor start monitor '-debug -
interval 2'
```

MPX_AUTOEXCLUDE_TIMEOUT オプション

コーディネータノード上でのセカンダリノードの自動除外のタイムアウト。このオプションは、指定されたフェールオーバーノードには適用されません。

指定できる値

0 ~ 10080 分 (1 週間)。0 はノードが自動除外されないことを示します。値は、**MPX_HEARTBEAT_FREQUENCY** の設定でちょうど割り切れる分単位の値にする必要があります。たとえば、**MPX_HEARTBEAT_FREQUENCY** の設定が 120 (2 分) の場合、**MPX_AUTOEXCLUDE_TIMEOUT** は 2 で割り切れる値に設定する必要があります。

デフォルト値

60 分

スコープ

オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、**SET ANY SYSTEM OPTION** システム権限が必要です。設定はすぐに有効になり、サーバの再起動後も存続します。

MPX_HEARTBEAT_FREQUENCY オプション

ハートビートスレッドが起動し、周期的操作 (コーディネータの接続のチェックやセカンダリノード上の接続プールのクリーンアップなど) を実行するまでの間隔。ハートビートスレッドは、セカンダリサーバからコーディネータへの専用の内部接続を維持します。

指定できる値
2 秒 ~ 3600 秒

デフォルト値
60 秒

スコープ
オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。変更を有効にするには、サーバを再起動する必要があります。

MPX_IDLE_CONNECTION_TIMEOUT オプション

セカンダリノード上の接続プールにある未使用の接続がクローズされるまでの待機時間。

指定できる値
0 秒以上無制限

デフォルト値
600 秒

スコープ
オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。設定はすぐに有効になり、サーバの再起動後も存続します。

MPX_LIVENESS_TIMEOUT オプション

最初の切断後にハートビートがコーディネータへの再接続に失敗した場合に、セカンダリサーバのハートビートがコーディネータがオフライン状態であることを宣言するまでの、秒単位の時間です。このオプションにより、コーディネータがグローバルトランザクションのサスペンド状態を保持する時間も決定されます。

データベースオプション

指定可能な値

0 ~ 604800 秒 (1 週間)

デフォルト

3600 秒 (1 時間)

スコープ

このオプションはすべてのマルチプレックスノードに影響します。また、ノード固有の値または接続固有の値はありません。オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。稼働中のサーバの MPX_LIVENESS_TIMEOUT 値を変更すると、それ以降サスペンドされる接続に対して新しい値がすぐに有効になります。変更後の値は、現在サスペンドされているすべてのトランザクションの残余タイムアウト時間にも、即座に影響します。

備考

ライタが **MPX_LIVENESS_TIMEOUT** の期間内にサスペンドされたトランザクションの再開に失敗した場合、そのトランザクションはコミットできなくなり、ユーザはトランザクションをロールバックする必要があります。コーディネータは、 $2 \times$ **MPX_LIVENESS_TIMEOUT** の期間、グローバルトランザクションのサスペンド状態を保持します。対応するライタが $2 \times$ **MPX_LIVENESS_TIMEOUT** の期間内でのトランザクションの再開に失敗すると、コーディネータはサスペンドされたトランザクションをロールバックします。

MPX_LIVENESS_TIMEOUT の値は、常に有効性をチェックする期間を制御する現在の **MPX_HEARTBEAT_FREQUENCY** の値の倍数を指定します。コーディネータは、**MPX_LIVENESS_TIMEOUT** の値を内部で 2 倍にします。

MPX_MAX_CONNECTION_POOL_SIZE オプション

セカンダリノード上の接続プールで許容される接続の最大数。

指定できる値

1 ~ 1000

デフォルト値

10

スコープ

オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。設定はすぐに有効になり、サーバの再起動後も存続します。

備考

INC 接続はセカンダリノードとコーディネータノード間のサーバ間接続です。DDL または読み書き操作を実行しているセカンダリサーバ上のユーザ接続のそれぞれに関連付けられます。接続はコマンドがコミットするか、ロールバックするまで継続され、その後、プールに戻されます。トランザクションの存続時間が短い場合は、DDL または RW 操作を実行している多くのユーザ接続にとって、**MPX_MAX_CONNECTION_POOL_SIZE** のデフォルト設定で十分です。複数の接続が同時に DDL または読み書き操作を実行する場合やトランザクションの処理時間が長い場合は、**MPX_MAX_CONNECTION_POOL_SIZE** の値を増やしてください。たとえば、複数のユーザ接続がコミットなしで同時ロードを実行する場合などはこの値を増やします。

MPX_MAX_CONNECTION_POOL_SIZE を超えると、「SQL Anywhere エラー -1004000: 接続プール内の接続数が上限を超えました」が返されます。

必要なプールサイズを見積もる場合は、-gm サーバオプションの設定を考慮します。-gm 設定は、セカンダリサーバに接続可能なユーザ数を指定します。INC 接続は含まれませんが、この数字に加算します。ユーザあたりの読み書きまたは DDL 操作が発生する回数を見積もるためのアプリケーション要件を使用し、それに合わせてプールサイズを増やします。

-gn 設定とコア数に応じて、接続 (INC またはユーザ) ごとにメモリオーバーヘッドが加算されます。メモリとスレッド接続の負荷は、SAP Sybase IQ サーバの応答時間に影響する場合があります。

MPX_MAX_UNUSED_POOL_SIZE オプション

セカンダリノード上の接続プールでの未使用接続の最大数。

指定できる値

0 ~ 最大プールサイズ

デフォルト値

0

スコープ

オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。設定はすぐに有効になり、サーバの再起動後も存続します。

MPX_WORK_UNIT_TIMEOUT オプション

マルチプレックス DQP リーダが、完了していない分散作業を他の DQP ワーカーノードに再割り当てするまでの時間 (秒数)。

指定できる値

0 ~ 3600 秒。

DQP 作業単位は、通常わずか数秒の長さに分けられます。ワーカーノードがオフラインになるか異常に高い負荷をかけられた場合、以前にそのワーカーノードに割り当てられた DQP 作業は、指定されたタイムアウト後に他のノードに再割り当てされます。

デフォルト値

60 秒

通常、このオプションをデフォルト値から変更する必要はありません。ただし、まれに、クエリの間接結果が非常に大きいため個々の作業単位がタイムアウトする場合には、このオプションを大きくします。

ネットワークやサーバの信頼性が低いために分散作業が失われ、タイムアウト間隔が長くなりすぎる場合は、このオプションを小さくします。このオプションを下げすぎると、不必要にタイムアウトが早くなります。

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

NEAREST_CENTURY オプション [TSQL]

文字列から日付への変換における 2 桁の年の解釈を制御します。

指定できる値

0 - 100

デフォルト値

50

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

NEAREST_CENTURY は、文字列から日付またはタイムスタンプに変換する際の 2 桁の年の処理を制御します。

NEAREST_CENTURY の設定値は、ロールオーバーポイントとして動作する数値です。この値より小さい 2 桁の年は 20yy に変換され、この値以上の年は 19yy に変換されます。

Adaptive Server と SAP Sybase IQ は最も近い世紀を使用するため、yy が 50 より小さい年は 20yy に設定されます。

NOEXEC オプション

オプティマイザクエリプランを生成します。実行はしません。

指定できる値

ON、OFF

デフォルト値

OFF

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

クエリの処理方法を決定する際、IQ オプティマイザは、各クエリをクエリエンジンがどう処理するかを記述したマップとして、クエリプランを生成します。このオプションが ON の場合、オプティマイザはクエリのプランをクエリエンジンに送るのではなく、IQ メッセージファイルに送信します。NOEXEC は、クエリと、クエリを含むコマンドに影響します。

NOEXEC を ON に設定すると、**INSERT...VALUES**、**INSERT...SELECT**、**INSERT...LOCATION**、**SELECT...INTO**、**LOAD TABLE**、**UPDATE**、**TRUNCATE TABLE**、**DELETE**、および更新可能カーソルの各操作の実行も阻止されます。

EARLY_PREDICATE_EXECUTION オプションが ON に設定されている場合、NOEXEC オプションが ON でも、SAP Sybase IQ はクエリプランの生成に先立ってすべてのクエリのローカル述部を実行します。生成されるクエリプランは実行時プランと同一です。

参照：

- **EARLY_PREDICATE_EXECUTION** オプション (658 ページ)

NON_ANSI_NULL_VARCHAR オプション

挿入/ロード/更新時に長さゼロの VARCHAR データを NULL として扱うかどうかを制御します。

指定できる値

ON、OFF

デフォルト値

OFF

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザーレベルで設定できません。データベースレベルで設定した場合、値は新しいユーザーのデフォルト値になりますが、既存のユーザーには影響を与えません。ユーザーレベルで設定した場合は、そのユーザーの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザーのオプションをデータベースレベルまたはユーザーレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、**SET ANY PUBLIC OPTION** システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

NON_ANSI_NULL_VARCHAR は、ロードまたは更新時の長さゼロの VARCHAR データの処理を非 ANSI (Version 12.03.1) 動作に戻す場合に使用します。このオプションを OFF にした場合、長さゼロの VARCHAR データはロード/挿入/更新時に長さゼロとして保存されます。このオプションを ON にした場合、長さゼロの VARCHAR データはロード/挿入/更新時に NULL として格納されます。

NON_KEYWORDS オプション [TSQL]

個々のキーワードをオフにして、識別子として使用できるようにします。

指定できる値
文字列

デフォルト値
" (空の文字列)

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

NON_KEYWORDS は各キーワードを無効にします。データベース内のいずれかの識別子が現在キーワードとして使用されている場合、すべてのアプリケーションまたはスクリプトでその識別子を二重引用符で囲むか、または **NON_KEYWORDS** オプションを指定することによって、キーワードとしての機能を無効にできます。

次の文を記述すると、**TRUNCATE** と **SYNCHRONIZE** はキーワードとして認識されません。

```
SET OPTION NON_KEYWORDS = 'TRUNCATE, SYNCHRONIZE'
```

このオプションによる新規設定が以前の設定に置き換わります。次の文は以前の設定内容をすべてクリアします。

```
SET OPTION NON_KEYWORDS =
```

データベースオプション

このオプションを使用すると、オフにしたキーワードを使用する SQL 文は構文エラーが発生します。

NOTIFY_MODULUS オプション

特定のコマンドから発行される通知メッセージのデフォルト周期を制御します。

指定できる値
整数値

デフォルト値

0 - 新しい SAP Sybase IQ 16.0 以降のデータベースの場合。

100000 - 16.0 より前のアップグレードしたデータベースの場合。

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

このオプションは、通知メッセージを生成する特定のコマンドに対して SAP Sybase IQ が発行するデフォルトの通知メッセージ数の設定に使用します。

CREATE INDEX、**LOAD TABLE**、**DELETE** などの一部のコマンドでは、NOTIFY 句を指定すると、このオプションの値が上書きされます。NOTIFY 句をサポートしていない他のコマンドでは、常にこのオプションの値が使用されます。デフォルトでは、受信可能なメッセージの数に制限はありません。

ODBC_DISTINGUISH_CHAR_AND_VARCHAR オプション

SAP Sybase IQ ODBC ドライバが CHAR カラムを表現する方法を制御します。

指定できる値
ON、OFF

デフォルト値
OFF

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

接続が開かれると、SAP Sybase IQ ODBC ドライバはこのオプションを使用して CHAR カラムの記述方法を決定します。ODBC_DISTINGUISH_CHAR_AND_VARCHAR が OFF (デフォルト値) に設定されている場合、CHAR カラムは SQL_VARCHAR として記述されます。このオプションが ON に設定されている場合、CHAR カラムは SQL_CHAR として記述されます。VARCHAR カラムは常に SQL_VARCHAR として記述されます。

ON_CHARSET_CONVERSION_FAILURE オプション

文字変換でエラーが起きた場合の対応を制御します。

指定できる値

文字変換エラー	アクション
IGNORE	エラーも警告も表示されない。
WARNING	置き換えと不正な文字が警告としてレポートされる。不正な文字は変換されない。
ERROR	置き換えと不正な文字がエラーとしてレポートされる。

デフォルト値**IGNORE****スコープ**

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場

合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

ON_CHARSET_CONVERSION_FAILURE は、文字変換でエラーが起きた場合の対応を制御します。

シングルバイトからシングルバイトへの変換では、置き換えと不正な文字を報告できないため、このオプションは IGNORE に設定する必要があります。

ON_ERROR オプション [Interactive SQL]

Interactive SQL の文の実行中にエラーが起こった場合の動作を制御します。

指定できる値

値	説明
STOP	Interactive SQL は、ファイルからの文の実行を停止し、入力用の文ウィンドウに戻る。
PROMPT	Interactive SQL は、ユーザに続行するかどうかを確認するプロンプトを表示する。
CONTINUE	[メッセージ] ウィンドウ枠にエラーが表示される。Interactive SQL は文の実行を続行する。
EXIT	Interactive SQL が終了する。
NOTIFY_CONTINUE	エラーがレポートされるとともに、[Enter] を押すか [OK] をクリックして続行するようにユーザに求めるメッセージが表示される。
NOTIFY_STOP	エラーがレポートされるとともに、[Enter] を押すか [OK] をクリックして文の実行を停止するようにユーザに求めるメッセージが表示される。
NOTIFY_EXIT	エラーがレポートされるとともに、[Enter] を押すか [OK] をクリックして Interactive SQL を終了するように求めるメッセージが表示される。

デフォルト値
PROMPT

備考

文の実行時にエラーが起きた場合の対応を制御します。 .SQL ファイルを実行している場合は、 STOP と EXIT のどちらに設定しても同じ結果になります。

ON_TSQL_ERROR オプション [TSQL]

ストアードプロシージャでのエラー処理を制御します。

指定できる値

値	説明
STOP	エラーの検出と同時に実行が停止される。
CONDITIONAL	プロシージャで ON EXCEPTION RESUME が使用されており、エラーのすぐ後の文でエラーが処理される場合は、実行が継続される。それ以外の場合は、終了する。
CONTINUE	次に続く文に関係なく実行が継続される。複数のエラーがある場合は、ストアードプロシージャで最初に検出されたエラーが返される。このオプションを指定した場合の動作は、Adaptive Server の動作とほとんど同じ。

デフォルト値

CONDITIONAL

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

ON_TSQL_ERROR はストアードプロシージャでのエラー処理を制御します。

ON_TSQL_ERROR の CONDITIONAL と CONTINUE の設定はどちらも、Adaptive Server との互換性のために使用されます。CONTINUE を指定した場合、Adaptive Server の動作に最も近いシミュレーションが行われます。CONDITIONAL の設定

では、エラーがより早い段階でレポートされるため、特に新規の Transact-SQL を作成する場合などには、こちらの設定をおすすめします。

Adaptive Server との互換性については、『リファレンス：ビルディングブロック、テーブル、およびプロシージャ』を参照してください。

このオプションを STOP または CONTINUE に設定すると、このオプションによって **CONTINUE_AFTER_RAISERROR** オプションの設定が上書きされます。このオプションが **CONDITIONAL** (デフォルト) の場合は、**RAISERROR** 文の後の動作は **CONTINUE_AFTER_RAISERROR** オプションの設定によって決まります。

参照：

- CREATE PROCEDURE 文 (210 ページ)
- CREATE PROCEDURE 文 [T-SQL] (218 ページ)
- RAISERROR 文 [T-SQL] (476 ページ)
- CONTINUE_AFTER_RAISERROR オプション [TSQL] (624 ページ)

OS_FILE_CACHE_BUFFERING オプション

IQ メイン DB 領域のファイルシステムバッファリングの使用を制御します。

指定できる値

ON、OFF

デフォルト値

OFF。デフォルトは新規作成されたデータベースに対してのみ有効です。

スコープ

オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。すぐに有効になります。

備考

OS_FILE_CACHE_BUFFERING を OFF に設定すると、IQ メインストアファイルに対するファイルシステムバッファリングは行われません。ファイルシステムバッファリングをオフにすると、ファイルシステムバッファキャッシュからのデータコピーがメイン IQ バッファキャッシュに保存されます。通常これにより、IQ バッファマネージャとオペレーティングシステムのファイルシステムバッファとのメモリ競合によるページングが減ります。**OS_FILE_CACHE_BUFFERING** によってページングが減少してパフォーマンスが向上することがありますが、データベースの IQ ページサイズがファイルシステムのブロックサイズより小さい場合 (通常、テスト時などにのみ該当)、特にマルチユーザによる操作でパフォーマンスが低下します。

このオプションを使用するかどうかは、各マシンの環境に応じて実際に動作させてから判断してください。新しい設定を有効にするには、データベースを再起動する必要があります。

マルチプレックスデータベースは、ダイレクト I/O ファイルシステムデバイスをサポートしていません。マルチプレックス機能には、共有ロー記憶領域が必要だからです。ダイレクト I/O パフォーマンスオプションは、シンプレックスデータベースに対してのみサポートされています。

このダイレクト I/O パフォーマンスオプションは、Sun Solaris UFS、Linux、Linux IBM、AIX、Windows ファイルシステムでのみ有効です。このオプションは HP-UX と HP-UXi には影響しません。また、ローディスク上に構築されたデータベースにも影響はありません。Linux では、ダイレクト I/O はカーネルバージョン 2.6.x でサポートされます。

Linux カーネルバージョン 2.6 および AIX でダイレクト I/O を有効にするには、環境変数 `IQ_USE_DIRECTIO` を 1 に設定します。Linux カーネルバージョン 2.6 および AIX では、ダイレクト I/O はデフォルトで無効になっています。

`IQ_USE_DIRECTIO` は、Sun Solaris と Windows には影響しません。

注意： SAP Sybase IQ は、Linux カーネルバージョン 2.4 でダイレクト I/O をサポートしていません。Linux カーネルバージョン 2.4 で `IQ_USE_DIRECTIO` 環境変数を設定すると、SAP Sybase IQ サーバは起動しません。エラー「エラー：無効なブロック I/O 引数です。<pathname> はディレクトリの可能性があります。またはプラットフォームの最大ファイルサイズの制限を超えているか、サポートされていない OS で直接 IO を使用しようとしてしました」がレポートされます。

LOAD TABLE 入力ファイルがデータファイルに対するシンボリックリンクである場合は、Linux カーネルバージョン 2.6 で、同じエラーが発生してダイレクト I/O が失敗することがあります。

OS_FILE_CACHE_BUFFERING_TEMPDB は、IQ テンポラリストアファイルのファイルシステムバッファリングを制御します。

参照：

- `OS_FILE_CACHE_BUFFERING_TEMPDB` オプション (739 ページ)

OS_FILE_CACHE_BUFFERING_TEMPDB オプション

IQ テンポラリ DB 領域のファイルシステムバッファリングの使用を制御します。

指定できる値

ON、OFF

デフォルト値
OFF

スコープ
オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。変更を有効にするには、データベースを停止し、再起動させてください。

備考

OS_FILE_CACHE_BUFFERING_TEMPDB を OFF に設定すると、IQ テンポラリストアファイルに対するファイルシステムバッファリングは行われません。ファイルシステムバッファリングをオフにすると、ファイルシステムバッファキャッシュからのデータコピーがメイン IQ バッファキャッシュに保存されます。通常これにより、IQ バッファマネージャとオペレーティングシステムのファイルシステムバッファとのメモリ競合によるページングが減ります。

OS_FILE_CACHE_BUFFERING_TEMPDB によってページングが減少してパフォーマンスが向上することがありますが、データベースの IQ ページサイズがファイルシステムのブロックサイズより小さい場合 (通常、テスト時など) のみ該当)、特にマルチユーザによる操作でパフォーマンスが低下します。

このオプションを使用するかどうかは、各マシンの環境に応じて実際に動作させてから判断してください。新しい設定を有効にするには、データベースを再起動する必要があります。

マルチプレックスデータベースは、ダイレクト I/O ファイルシステムデバイスをサポートしていません。マルチプレックス機能には、共有ロー記憶領域が必要だからです。ダイレクト I/O パフォーマンスオプションは、シンプレックスデータベースに対してのみサポートされています。

このダイレクト I/O パフォーマンスオプションは、Sun Solaris UFS、Linux、Linux IBM、AIX、Windows ファイルシステムでのみ有効です。このオプションは HP-UX と HP-UXi には影響しません。また、ローディスク上に構築されたデータベースにも影響はありません。Linux では、ダイレクト I/O はカーネルバージョン 2.6.x でサポートされます。

Linux カーネルバージョン 2.6 および AIX でダイレクト I/O を有効にするには、環境変数 `IQ_USE_DIRECTIO` を 1 に設定します。Linux カーネルバージョン 2.6 および AIX では、ダイレクト I/O はデフォルトで無効になっています。

`IQ_USE_DIRECTIO` は、Sun Solaris と Windows には影響しません。

注意：SAP Sybase IQ は、Linux カーネルバージョン 2.4 でダイレクト I/O をサポートしていません。Linux カーネルバージョン 2.4 で `IQ_USE_DIRECTIO` 環境変数を

設定すると、SAP Sybase IQ サーバは起動しません。エラー「エラー : 無効なブロック I/O 引数です。<pathname> はディレクトリの可能性があります。またはプラットフォームの最大ファイルサイズの制限を超えているか、サポートされていない OS で直接 IO を使用しようとしてしました」がレポートされます。

LOAD TABLE 入力ファイルがデータファイルに対するシンボリックリンクである場合は、Linux カーネルバージョン 2.6 で、同じエラーが発生してダイレクト I/O が失敗することがあります。

OS_FILE_CACHE_BUFFERING は、IQ メインストアファイルのシステムファイルバッファリングを制御します。

参照：

- OS_FILE_CACHE_BUFFERING オプション (738 ページ)

POST_LOGIN_PROCEDURE オプション

結果セットにユーザが正常にログインした直後にクライアントアプリケーションにより表示されるメッセージが記載されたログインプロシージャを指定します。

指定できる値
文字列

デフォルト値
dbo.sa_post_login_procedure

スコープ
オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY SECURITY OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考
デフォルトのポストログインプロシージャ **dbo.sa_post_login_procedure** は、ユーザが正常にログインした直後に実行されます。

SET ANY SECURITY OPTION システム権限がある場合は、ログイン後のアクションをカスタマイズできます。そのためには、新しいプロシージャを作成し、その新しいプロシージャを呼び出すように **POST_LOGIN_PROCEDURE** を設定します。

dbo.sa_post_login_procedure は編集しないでください。使用するすべてのデータベースで、カスタマイズされたポストログインプロシージャを作成してください。ポストログインプロシージャでは、クライアントアプリケーション Interactive SQL および Interactive SQL Classic をサポートしています。

参照：

- LOGIN_PROCEDURE オプション (708 ページ)

PRECISION オプション

カタログストアのクエリに限定して、10 進数の算術の結果に使用できる最大桁数を指定します。

指定できる値
126

デフォルト値
126

スコープ
オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。すぐに有効になります。

備考
精度は、小数点の左右の合計桁数です。**PRECISION** 値はデフォルトの 126 に固定されています。SCALE オプションで、カタログストアに対するクエリの計算結果が **PRECISION** で指定された最大値にトランケートされる場合に適用する小数点以下の最小桁数を指定します。

注意： SQL Anywhere では、数値関数でサポートされている最大値は 255 です。数値関数の精度がサポートされている最大値を超えていると、「関数 '_funcname' の結果のデータ型が、サポートされている最大数値精度の 255 を超えています。数値関数の精度に適切な値を設定してください。'location'」というエラーが表示されます。

参照：

- SCALE オプション (769 ページ)
- MAX_CLIENT_NUMERIC_PRECISION オプション (710 ページ)

PREFETCH オプション

フェッチの ON と OFF を切り替えることができます。また、ALWAYS 値を使用して、カーソル結果 (SENSITIVE カーソル タイプとプロキシテーブルに関わるカーソルに関するものを含む) をプリフェッチすることもできます。

指定できる値

ON、OFF、ALWAYS

デフォルト値

ON

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

PREFETCH は、ローがクライアントアプリケーションで使用できるようになる前にクライアント側にフェッチされるかどうかを制御します。これはカタログストアの場合にのみ適用されます。クライアントアプリケーションがローを一度に 1 つずつ要求している場合でも (たとえば、カーソルのローをループしている場合など)、一度に複数のローをフェッチすることによって、データベースへの要求の数が制限されるため、応答時間が最小化され、全体的なスループットが向上します。

PREFETCH の設定は、Open Client 接続と JDBC 接続の場合、および IQ ストアに対しては無視されます。

PREFETCH_BUFFER_LIMIT オプション

プリフェッチに使用するメモリ量を指定します。

指定できる値
整数

デフォルト値
0

スコープ
オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。変更を有効にするには、データベースを停止し、再起動させてください。

備考

PREFETCH_BUFFER_LIMIT では、SAP Sybase IQ がプリフェッチ (データベースの先読み) に使用できるキャッシュページ数を定義できます。

テクニカルサポートから指示があった場合を除いて、このオプションは設定しないでください。

参照：

- PREFETCH_BUFFER_PERCENT オプション (744 ページ)

PREFETCH_BUFFER_PERCENT オプション

プリフェッチに使用するメモリをパーセントで指定します。

指定できる値
0 - 100

デフォルト値
40

スコープ
オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。変更を有効にするには、データベースを停止し、再起動させてください。

備考

PREFETCH_BUFFER_PERCENT は、**PREFETCH_BUFFER_LIMIT** の代替として使用され、プリフェッチに使用できるキャッシュをパーセント単位で指定します。

テクニカルサポートから指示があった場合を除いて、このオプションは設定しないでください。

参照：

- PREFETCH_BUFFER_LIMIT オプション (744 ページ)

PREFETCH_GARRAY_PERCENT オプション

HG インデックスに挿入できるプリフェッチリソースをパーセントで指定します。

指定できる値

0 - 100

デフォルト値

60

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できません。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

PREFETCH_SORT_PERCENT と同様、このオプションは、HG インデックスへの挿入時に使用できるプリフェッチリソースのパーセント量の指定に使用します。

テクニカルサポートから指示があった場合を除いて、このオプションは設定しないでください。

PREFETCH_SORT_PERCENT オプション

ソートオブジェクトが使用するプリフェッチリソースをパーセントで指定します。

指定できる値

0 - 100

デフォルト値

20

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

PREFETCH_SORT_PERCENT は、1つのソートオブジェクトが使用するプリフェッチリソースをパーセンテージで指定します。この値を大きくすると、シングルユーザの場合の挿入と削除のパフォーマンスが向上しますが、マルチユーザ動作に対して悪影響を及ぼす可能性があります。

テクニカルサポートから指示があった場合を除いて、このオプションは設定しないでください。

PRESERVE_SOURCE_FORMAT オプション [database]

プロシージャ、ビュー、イベントハンドラのオリジナルのソース定義をシステムファイル内に保存するかどうかを制御します。保存する場合は、フォーマットされたソースが SYSTABLE、SYSPROCEDURE、および SYSEVENT のカラム source に保存されます。

指定できる値

ON、OFF

デフォルト値

ON

スコープ

オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。すぐに有効になります。

備考

PRESERVE_SOURCE_FORMAT が ON の場合、サーバは、プロシージャ、ビュー、イベントの CREATE 文と ALTER 文からフォーマットされたソースを保存し、オリジナルのソース定義を適切なシステムテーブルの source カラムに配置します。

フォーマットされていないソーステキストは、同じシステムテーブルの `proc_defn` カラムと `view_defn` カラムに格納されます。フォーマットされたソースカラムでは、スペース、コメント、大文字または小文字が使用された定義を参照できます。

このオプションを OFF にすると、オブジェクト定義の保存に使用されるデータベースの領域を減らすことができます。このオプションは PUBLIC ロールに対してのみ設定できます。

QUERY_DETAIL オプション

追加のクエリ情報をクエリプランの Query Detail セクションに含めるかどうかを指定します。

指定できる値
ON、OFF

デフォルト値
ON

スコープ
オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考
QUERY_DETAIL と **QUERY_PLAN** (または **QUERY_PLAN_AS_HTML**) が両方 ON になっていると、SAP Sybase IQ はクエリプランの生成時にクエリに関する追加情報を表示します。**QUERY_PLAN** と **QUERY_PLAN_AS_HTML** が OFF の場合、このオプションは無視されます。

QUERY_PLAN が ON (デフォルト値) の場合、**QUERY_DETAIL** も ON である場合は特に、メッセージログラッピングまたはメッセージログアーカイブを有効にしてメッセージログファイルが満杯にならないようにしてください。

参照：

- **QUERY_PLAN** オプション (748 ページ)
- **QUERY_PLAN_AS_HTML** オプション (750 ページ)

QUERY_NAME オプション

クエリプランで実行されるクエリに名前を付けます。

指定できる値

引用符で区切った文字列 (最大 80 字)

デフォルト値

" (空の文字列)

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

QUERY_NAME オプションには、引用符で区切られた任意の文字列 (80 字以内) を指定できます。次に例を示します。

```
set temporary option Query_Name = 'my third query'
```

このオプションが設定されていると、.iqmsg ファイルまたは .html ファイルに送信されるクエリプランの先頭近くに、次のような行が含まれます。

```
Query_Name: 'my third query'
```

スクリプト内の各クエリの前でこのオプションに別の値を設定することによって、特定のクエリの正しいクエリプランの識別がはるかに容易になります。クエリ名は HTML クエリプランのファイル名にも追加されます。このオプションが、クエリに対してその他の影響を及ぼすことはありません。

QUERY_PLAN オプション

追加のクエリプランを SAP Sybase IQ メッセージファイルに出力するかどうかを指定します。

指定できる値

ON、OFF

デフォルト値
OFF

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

このオプションを ON にすると、SAP Sybase IQ では IQ メッセージファイルにテキストクエリプランが作成されます。これらのクエリプランは、クエリツリートポグラフィおよび最適化と実行についての詳細を表示します。OFF の場合、このようなメッセージは生成されません。情報は <dbname>.iqmsg ファイルに送信されます。

参照：

- QUERY_DETAIL オプション (747 ページ)
- QUERY_PLAN_AFTER_RUN オプション (749 ページ)
- QUERY_PLAN_AS_HTML オプション (750 ページ)

QUERY_PLAN_AFTER_RUN オプション

クエリの実行完了後に、クエリプラン全体を出力します。

指定できる値
ON、OFF

デフォルト値
ON

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場

合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

QUERY_PLAN_AFTER_RUN を ON にすると、クエリ実行の完了後にクエリプランが出力されます。これにより、クエリの各ノードから渡された実際のロー数などの追加情報をクエリプランに追加できます。

このオプションを機能させるには、**QUERY_PLAN** オプションが ON に設定されている必要があります。このオプションを **QUERY_DETAIL** と組み合わせて使用すると、クエリプランレポートに追加情報を生成できます。

参照：

- **QUERY_DETAIL** オプション (747 ページ)
- **QUERY_PLAN** オプション (748 ページ)
- **QUERY_PLAN_AS_HTML** オプション (750 ページ)

QUERY_PLAN_AS_HTML オプション

Web ブラウザで表示できるグラフィカルなクエリプランを HTML フォーマットで生成します。

指定できる値

ON、OFF

デフォルト値

OFF

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

QUERY_PLAN_AS_HTML を指定することによって、グラフィカルなクエリプランを HTML フォーマットで生成できます。

このオプションを設定する場合は、クエリプランに関連付けられているクエリを識別できるように、各クエリに対して **QUERY_NAME** オプションも設定してください。

SAP Sybase IQ は、.iqmsg ファイルと同じディレクトリにプランを書き込みます。クエリプランのファイル名には、次の規則が適用されます。

user-name_query-name_server-type_server-number_YYYYMMDD_HHMMSS_query-number_fragment-number.html.

たとえば、ユーザ DBA がテンポラリーオプション **QUERY_NAME** を 'Query_1123' に設定した場合、2012 年 11 月 8 日の午前 8 時 30 分ちょうどに作成されたファイルは、DBA_QUERY_1123_L_0_20121108_083000_4.html という名前になります。既存のファイルを上書きしないように、日付、時刻、およびユニークな **query-number** がファイル名に付加されます。**server-type** パラメータは、プランの発生元がリーダ (L) ノードとワーカ (W) ノードのどちらであることを示します。**server-number** は、すべての html ファイルが 1 つのディレクトリにルーティングされた場合に、プランの発生元のサーバを示します。

マルチプレックスサーバでは、ワーカが実行したフラグメントごとに html ファイルがワーカノードで生成されるため、1 つのクエリで複数の html ファイルが生成されることがあります。これらのファイルは **fragment-number** によって識別されません。

注意： この機能を使用する場合は、.iqmsg ファイルとログファイルの増大に対応できるだけの領域があるかどうか、ディスク領域の使用状況をモニタします。メッセージログラッピングまたはメッセージログアーカイブを有効にしてメッセージログファイルが満杯にならないようにしてください。

QUERY_PLAN_AS_HTML は、**QUERY_PLAN** オプションの設定とは関係なく機能します。つまり、**QUERY_PLAN_AS_HTML** が ON の場合、**QUERY_PLAN** が ON であるかどうかにかかわらず、HTML フォーマットのクエリプランが生成されます。

広く利用されているブラウザの新しいバージョンではこの機能がサポートされています。一部のブラウザでは、非常に複雑なクエリに対して生成されたプランで表示の問題が発生することもあります。

シンプレックスの出力例

シンプレックスサーバで Q1123 という名前のクエリプランが実行されたときに生成された出力：

```
DBA_QUERY_Q1123_L_0_20121108_083000_4.html
```

シングルプレックスサーバは常に、プランの発生元がリーダー (L) であることを示す **server-type** パラメータと、値 0 の **server-number** パラメータを返します。シングルプレックスの出力に **fragment-number** は含まれません。

マルチプレックスの出力例

マルチプレックスサーバで Q101 という名前のクエリプランが実行されたときに生成された出力。リーダーノード (L) の **server-number** は 1、ワーカー (W) ノードの番号は 2 と 3 です。ワーカー出力のクエリ番号の後ろに **fragment-number** があることに注意してください。

リーダーノードで生成された出力：

```
DBA_L_1_Q101_20121108_083000_94.html
```

いずれかのワーカーノードからの出力：

```
DBA_W_2_Q101_20121108_083000_94_2.html
```

```
DBA_W_2_Q101_20121108_083000_94_1.html
```

別のワーカーからの対応する出力：

```
DBA_W_3_Q101_20121113-054928_94_2.html
```

```
DBA_W_3_Q101_20121113-054933_94_1.html
```

参照：

- QUERY_NAME オプション (748 ページ)
- QUERY_PLAN オプション (748 ページ)
- QUERY_PLAN_AFTER_RUN オプション (749 ページ)

QUERY_PLAN_AS_HTML_DIRECTORY オプション

SAP Sybase IQ が HTML のクエリプランを書き込むディレクトリを指定します。

指定できる値

ディレクトリのパス名を含む文字列

デフォルト値

" (空の文字列)

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場

合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

QUERY_PLAN_AS_HTML オプションが ON で、**QUERY_PLAN_AS_HTML_DIRECTORY** オプションでディレクトリが指定されている場合、SAP Sybase IQ は指定されたディレクトリに HTML のクエリプランを書き込みます。このオプションでは、HTML クエリプランをサーバディレクトリの外部で作成できるため、セキュリティが強化されます。**QUERY_PLAN_AS_HTML_DIRECTORY** オプションが使用されていない場合、クエリプランはデフォルトのディレクトリ (.iqmsg ファイルのディレクトリ) に送信されます。

QUERY_PLAN_AS_HTML オプションが ON で、**QUERY_PLAN_AS_HTML_DIRECTORY** に存在しないディレクトリが指定されている場合、SAP Sybase IQ は HTML クエリプランを保存しませんが、エラーは生成されません。この場合、クエリの実行は続けられ、メッセージが IQ メッセージファイルにログ記録されます。DBA はこの記録によって、HTML クエリプランが書き込まれなかったことを確認できます。指定されたディレクトリパスまたはディレクトリのパーミッションが適切でない場合、「HTML クエリプランを開くときにエラーが発生しました。file-name」というメッセージが .iqmsg ファイルに書き込まれます。

例

例として /system1/users/DBA/html_plans というディレクトリを作成し、このディレクトリに適切なパーミッションを設定します。次に、このオプションを設定してクエリを実行します。

```
SET TEMPORARY OPTION QUERY_PLAN_AS_HTML = 'ON';
SET TEMPORARY OPTION QUERY_PLAN_AS_HTML_DIRECTORY = '/system1/users/DBA/html_plans';
SELECT coll FROM tabl;
```

指定したディレクトリ /system1/users/DBA/html_plans 内のファイルに HTML クエリプランが書き込まれます。

参照：

- **QUERY_PLAN_AS_HTML** オプション (750 ページ)

QUERY_PLAN_MIN_TIME オプション

クエリ実行のスレッシュホールドを指定します。クエリ後のプランが生成されるのは、クエリ実行時間がスレッシュホールドを超過した場合のみです。

指定可能な値

整数値 (ミリ秒単位)

デフォルト

0

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

実行時間が非常に短いクエリ (マイクロクエリ) はクエリプランが生成されていない場合に処理が高速化します。このオプションを設定すれば、クエリプランの生成とこれらのクエリに関するクエリプラン生成コストを抑えることができます。QUERY_PLAN_MIN_TIME オプションは以下のオプションが設定されていない場合に無視されます。

- QUERY_PLAN = ON または QUERY_PLAN_AS_HTML = ON
- QUERY_PLAN_AFTER_RUN = ON
- QUERY_TIMING = ON

これらのオプションが設定されている場合は、**QUERY_PLAN_MIN_TIME** クエリ実行スレッシュホールドを設定することによって、実行時間が指定されたスレッシュホールドを超えていないクエリに対するクエリプランの生成が抑えられます。

QUERY_PLAN_TEXT_ACCESS オプション

Interactive SQL クライアントからクエリプランにアクセスすること、または SQL のさまざまな関数を使用してプランを取得することの許可/禁止を切り替えます。

指定できる値

ON、OFF

デフォルト値
OFF

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

QUERY_PLAN_TEXT_ACCESS オプションが ON の場合、ユーザは Interactive SQL クライアントからクエリプランを表示、保存、出力できます。オプションが OFF の場合、クエリプランはキャッシュされず、クエリプランに関連する他のデータベースオプションが Interactive SQL クライアントからのクエリプランの表示に影響を及ぼすことはありません。次のエラーメッセージが表示されます。

```
No plan available. The database option QUERY_PLAN_TEXT_ACCESS is OFF.
```

参照：

- QUERY_DETAIL オプション (747 ページ)
- QUERY_PLAN_AFTER_RUN オプション (749 ページ)
- QUERY_PLAN_AS_HTML オプション (750 ページ)
- QUERY_PLAN_TEXT_CACHING オプション (755 ページ)
- OUTPUT 文 [Interactive SQL] (463 ページ)

QUERY_PLAN_TEXT_CACHING オプション

ユーザが実行するクエリに対して SAP Sybase IQ が IQ プランを生成し、キャッシュするかどうかを指定できます。

指定できる値
ON、OFF

デフォルト値
OFF

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

IQ クエリプランのサイズはさまざまで、クエリが複雑な場合は非常に大きくなる場合があります。プランを Interactive SQL クライアントで表示できるようにキャッシュするには、多くのリソースが必要になることがあります。ユーザは、**QUERY_PLAN_TEXT_CACHING** オプションを使用することによって、プランのキャッシュに使用するリソースを制御できます。このオプションを OFF にすると (デフォルト)、クエリプランはそのユーザの接続ではキャッシュされません。

注意： **QUERY_PLAN_TEXT_ACCESS** を OFF にすると、クエリプランは、**QUERY_PLAN_TEXT_CACHING** の設定にかかわらず、そのユーザからの接続ではキャッシュされません。

参照：

- **QUERY_DETAIL** オプション (747 ページ)
- **QUERY_PLAN_AFTER_RUN** オプション (749 ページ)
- **QUERY_PLAN_AS_HTML** オプション (750 ページ)
- **QUERY_PLAN_TEXT_ACCESS** オプション (754 ページ)
- **OUTPUT** 文 [Interactive SQL] (463 ページ)

QUERY_ROWS_RETURNED_LIMIT オプション

結果セットの予想サイズにもとづき、クエリを拒否するためのローのスレッシュホールドを設定します。

指定できる値
整数値

デフォルト値
0

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

SAP Sybase IQ は、結果ローの予想数が **QUERY_ROWS_RETURNED_LIMIT** の値より大きいクエリを受け取ると、次のメッセージを生成してクエリを拒否します。

```
Query rejected because it exceeds resource:
Query_Rows_Returned_Limit
```

このオプションが 0 の場合 (デフォルト)、制限はなくなり、出力中のロー数によってクエリが拒否されることはありません。

QUERY_TEMP_SPACE_LIMIT オプション

クエリが拒否される前のテンポラリ領域の予測最大容量を指定します。

指定できる値
整数値

デフォルト値
0 (制限なし)

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

SAP Sybase IQ は、このオプションの値よりも大きいテンポラリ結果領域を必要とすることが予想されるクエリを受け取ると、次のメッセージを表示してクエリを拒否します。

```
Query rejected because it exceeds total space resource limit
```

ゼロに設定すると (デフォルト)、クエリによるテンポラリストアの使用に制限はありません。

ユーザは自分の環境でこのオプションを上書きし、全テンポラリストアを満杯にする可能性のあるクエリを実行できます。SET ANY SYSTEM OPTION システム権限を持つユーザの場合は、オプション MAX_TEMP_SPACE_PER_CONNECTION を設定して、暴走クエリがテンポラリストアを満杯にするのを防ぐことができます。MAX_TEMP_SPACE_PER_CONNECTION オプションでは、クエリだけでなくすべての DML 文での実際のテンポラリストアの使用がモニタされ、制限されます。

分散クエリ処理トランザクションでは、SAP Sybase IQ は、

QUERY_TEMP_SPACE_LIMIT オプションと **MAX_TEMP_SPACE_PER_CONNECTION** オプションのそれぞれの設定値を共有テンポラリストアに適用して、分散クエリに参加するすべてのノードが使用する共有とローカルのテンポラリ領域の総量を制限します。この制限によって、参加ノードの数にかかわらず、どのクエリも (DB 領域 IQ_SYSTEM_TEMP および IQ_SHARED_TEMP の) テンポラリ領域総量制限を超えることができません。

たとえば、制限値が 100 で参加する 4 つのノードがテンポラリ領域にそれぞれ 25 ユニットを使用する場合、このクエリは制限内となります。ノードにより使用される領域の総量が 100 を超える場合、クエリはロールバックされます。

参照：

- MAX_TEMP_SPACE_PER_CONNECTION オプション (721 ページ)

QUERY_TIMING オプション

特定のタイミング統計を収集して、クエリプランに表示するかどうかを決定します。

指定できる値

ON、OFF

デフォルト値

ON

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

このオプションは、サブクエリのタイミング統計の収集などのクエリエンジンの反復的な機能を制御するのに使用します。

クエリタイミングは、クエリプランの詳細で、一連のタイムスタンプとして表されます。これらのタイムスタンプは、クエリ演算フェーズ (Conditions、Prepare、Fetch、Complete) に対応しています。HTML と Interactive SQL クエリプランは、クエリのタイミングをタイムラインとして図示します。

QUOTED_IDENTIFIER オプション [TSQL]

二重引用符内の文字列の解釈を制御します。

指定できる値

ON、OFF

デフォルト値

ON

Open Client 接続の場合は OFF。

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

QUOTED_IDENTIFIER は、二重引用符で囲まれた文字列を識別子 (ON) とリテラル文字列 (OFF) のどちらとして解釈するかを制御します。このオプションは、Transact-SQL との互換性を維持するために組み込まれています。

SAP Control Center と Interactive SQL は、**QUOTED_IDENTIFIER** が OFF に設定されている場合、一時的に ON にします。その場合、この変更を通知するメッセージが表示されます。この変更は SAP Control Center または Interactive SQL 接続でのみ有効です。JDBC ドライバも **QUOTED_IDENTIFIER** を一時的に ON に設定します。

RECOVERY_TIME オプション

データベースサーバがシステム障害からの回復にかかる最長時間を分単位で設定します。

指定できる値
整数値 (分単位)

デフォルト値
2

スコープ
オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。変更を有効にするには、サーバを再起動する必要があります。

備考

このオプションを **CHECKPOINT_TIME** オプションとともに使用して、チェックポイントの実行タイミングを決定します。

ヒューリスティックを使用して、直前のチェックポイントから後に実行された操作をもとにリカバリ時間を計測します。そのため、リカバリ時間は正確ではありません。

参照：

- CHECKPOINT_TIME オプション (622 ページ)

RESERVED_KEYWORDS オプション

デフォルトで無効になっている個々のキーワードをオンにします。

指定できる値
文字列

デフォルト値
空の文字列

スコープ
オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。すぐに有効になります。

備考
このオプションは、デフォルトで無効になっている個々のキーワードをオンにします。LIMIT キーワードのみをオンにできます。

例
次の文を使用すると、LIMIT キーワードがキーワードとして認識されます。

```
SET OPTION RESERVED_KEYWORDS = 'LIMIT';
```

キーワード SET、OPTION および OPTIONS をオンにすることはできません。次の条件に従って単語がキーワードとして識別されるかどうかが判別されます (優先度順)。

- その単語が SQL Anywhere の予約語リストに載っている。
- IRESERVED_KEYWORDS オプションによってオンに設定されている。
- NON_KEYWORDS オプションによってオフに設定されている。

このオプションを設定するたびに、以前の設定が置き換えられます。次の文は以前の設定内容をすべてクリアします。

```
SET OPTION RESERVED_KEYWORDS = ;
```

RETURN_DATE_TIME_AS_STRING オプション

クエリされた日付、時間、またはタイムスタンプの値をクライアントに渡す方法を制御します。

指定できる値
ON、OFF

デフォルト値
OFF

スコープ
オプションは、現在の接続の存続期間に、または PUBLIC ロールに、テンポラリーオプションとしてのみ設定できます。

データベースオプション

このオプションを設定するには、**SET ANY PUBLIC OPTION** システム権限が必要です。すぐに有効になります。

備考

RETURN_DATE_TIME_AS_STRING は、日付、時刻、タイムスタンプの値をアプリケーションに返すときに、日付データ型または時刻データ型とするか、文字列とするかを指定します。

このオプションを ON に設定すると、サーバは、**TIMESTAMP_FORMAT**、**DATE_FORMAT**、または **TIME_FORMAT** オプションの設定を保持するために、日付、時刻、またはタイムスタンプの値を文字列に変換してからクライアントに送信します。

SAP Control Center と Interactive SQL では、**RETURN_DATE_TIME_AS_STRING** オプションは自動的に ON になります。

参照：

- **DATE_FORMAT** オプション (639 ページ)
- **TIME_FORMAT** オプション (801 ページ)
- **TIMESTAMP_FORMAT** オプション (802 ページ)

REVERT_TO_V15_OPTIMIZER オプション

このオプションを ON に設定すると、SAP Sybase IQ 15.x の動作を模倣するようにクエリ最適化に強制します。

指定可能な値

ON、OFF

デフォルト

- 15.x からアップグレードされたすべての 16.0 データベースでは ON
- 新規に作成されたすべての 16.0 データベースでは OFF

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、**SET ANY PUBLIC OPTION** システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。パーミッショ

ンが付与されている場合、任意の他のユーザまたはロール、あるいはロールを介してすべてのユーザに設定できます。すぐに有効になります。

備考

SAP Sybase IQ 16.0 では、ハッシュ分割テーブルとハッシュ範囲分割テーブルを活用した複数のジョインおよびグループ化アルゴリズムをサポートしているほか、他の新しいアルゴリズムもいくつかサポートしています。デフォルトでは、これらの新しいアルゴリズムがオプティマイザで考慮され、有効かつ該当する場合は選択されます。**REVERT_TO_V15_OPTIMIZER='ON'**を設定すると、オプティマイザのコストモデルに対して 16.0 で適用されたすべての変更が無効になります。また、これらの新しいジョインおよびグループ化アルゴリズムのすべてが無効になります(そのアルゴリズムが有効であり、AGGREGATION_PREFERENCE オプション、JOIN_PREFERENCE オプション、またはジョイン条件ヒント文字列で正の値を指定して、特に要求した場合を除く)。

REVERT_TO_V15_OPTIMIZER オプションは、通常、内部テストや手動調整クエリで使用されます。経験豊富な DBA のみ使用してください。

注意： **REVERT_TO_V15_OPTIMIZER='ON'** 時にクエリで RLV 対応テーブルを参照すると、エラーになります。

ROUND_TO_EVEN オプション

SQL 関数 **ROUND** の動作を制御します。

指定可能な値

ON、OFF

デフォルト

OFF

スコープ

SET ANY SYSTEM OPTION システム権限が必要です。PUBLIC ロールに対してのみ設定できます。すぐに有効になります。

備考

ROUND_TO_EVEN オプションが ON に設定されている場合、**ROUND** 関数は端数を最も近い偶数に丸めます。オプションが OFF に設定されている場合、**ROUND** 関数は SAP Sybase IQ の端数を四捨五入します。

SAP Control Center および Interactive SQL では、**ROUND_TO_EVEN** オプションが自動的に OFF になります。

ROUND 関数は、**ROUND_TO_EVEN** オプションが ON か OFF かによって返される値が異なります。

ROUND_TO_EVEN が ON に設定されている場合、`SELECT ROUND (convert(double, 0.25), 1) from iq_dummy` は 0.2 を返します。

ROUND_TO_EVEN が OFF に設定されている場合、`SELECT ROUND (convert(double, 0.25), 1) from iq_dummy` は 0.3 を返します。

ROW_COUNT オプション

クエリから返されるロー数を制限します。

指定できる値
整数値

デフォルト値
0 (返されるロー数の制限なし)

スコープ
オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、**SET ANY PUBLIC OPTION** システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考
このランタイムオプションにゼロでない値が設定されている場合、指定された数のローの処理が終わるとクエリ処理が停止されます。

このオプションは、キーワード **SELECT** を持つ文にのみ影響します。**UPDATE** 文と **DELETE** 文には影響しません。

SELECT 文の **FIRST** キーワードと **TOP** キーワードでも、クエリが返すローの数を制限できます。**FIRST** を使用することは、**ROW_COUNT** データベースオプションを 1 に設定するのと同じです。**TOP** を使用することは、**ROW_COUNT** をロー数と同じ数に設定するのと同じです。**TOP** と **ROW_COUNT** を両方とも設定した場合、**TOP** の値が優先されます。

ROW_COUNT オプションは、グローバル変数、システム関数、またはプロキシテーブルを伴うクエリで使用された場合、非決定的な結果を生成する可能性があります。これらのクエリは、部分的に CIS (Component Integrated Services) を使用して実行されます。このような場合は、**ROW_COUNT** を設定する代わりに **SELECT TOP *n*** を使用するか、またはグローバル変数をローカル変数に設定してそのローカル変数をクエリで使用してください。

参照：

- **QUERY_ROWS_RETURNED_LIMIT** オプション (756 ページ)
- **SELECT** 文 (523 ページ)

RV_AUTO_MERGE_EVAL_INTERVAL オプション

このオプションは評価期間を設定します。この期間はローレベルバージョン管理 (RLV) ストアと IQ メインストアの自動マージが実行される時点の決定に使用されます。

指定可能な値

1 ~ MAX_UINT (分)

デフォルト

15 (分)

スコープ

オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、**SET ANY SYSTEM OPTION** システム権限が必要です。すぐに有効になります。

備考

このオプションは、マージ評価機能の有効化の待機時間の間隔を分単位で設定するため使用されます。マージ評価機能は、設定されたスレッシュホールドと比較してローレベルバージョン管理 (RLV) 対応の各テーブルのマージパラメータを調べ、IQ メインストアへの RLV テーブルの非ブロッキング (バックグラウンド) マージを実行するかどうかを決定します。

間隔が終了したのに評価機能が有効な場合や、マージがすでに進行している場合、間隔は再設定されます。

間隔の新しい値はいずれも、マージ評価機能が次に有効化されるときに使用されます。

RV_MAX_ACTIVE_SUBFRAGMENT_COUNT オプション

この値により、マシン上のコアの使用量が最大になります。

指定可能な値

>=0

デフォルト

0

注意： デフォルト以外の値を使用することは推奨しません。CPU 使用率とバルクロードのスケラビリティに悪影響を及ぼす可能性があるためです。

スコープ

オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。テーブルに対する最初の書き込み操作により、このオプションはテーブルで有効になります。最初の書き込み操作が実行された後でこのオプションの値が変更されると、サーバの再起動後まで、新しい値はこのテーブルで有効になりません。

備考

値がデフォルト以外に設定されている場合、指定された値またはマシン上のコアの合計数のどちらか小さい方が使用されます。

RV_MERGE_NODE_MEMSIZE オプション

RV_MERGE_NODE_MEMSIZE などのマージスレッシュホールドに基づいて発生する、ローレベルのバージョン管理 (RLV) ストアと IQ メインストアとの自動的なマージです。このノードのスレッシュホールドを超過した場合、マージがトリガされます。

指定可能な値

0 ~ 100 (パーセント)

デフォルト

75 (パーセント)

スコープ

オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。すぐに有効になります。

備考

このオプションにより、RLV メモリサイズの合計のパーセンテージが、ノードのマージスレッシュォルドとして設定されます。RLV メモリサイズの合計がスレッシュォルドを超えると、マージ条件評価機能により、どのテーブルをマージするかが決定されます。ノードのスレッシュォルドを満たすため、複数テーブルのマージが必要な場合、マージされる各テーブルに対して並列マージがトリガされます。

RV_MERGE_TABLE_MEMPERCENT オプション

RV_MERGE_TABLE_MEMPERCENT などのマージスレッシュォルドに基づいて発生する、ローレベルのバージョン管理 (RLV) ストアと IQ メインストアとの自動的なマージです。このテーブルスレッシュォルドを超過した場合、特定のテーブルのマージがトリガされます。

指定可能な値

0 ~ 100 (パーセント)

デフォルト

0 (パーセント)

注意： **RV_MERGE_TABLE_MEMPERCENT = 0** の場合、(テーブルごとに) $100\%/N$ のスレッシュォルドが使用されます。ここで N は、ロード済の RLV 対応テーブルの数です。

スコープ

オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、**SET ANY SYSTEM OPTION** システム権限が必要です。すぐに有効になります。

備考

このオプションにより、RLV 対応テーブルのマージスレッシュォルドとして使用されるメモリのパーセンテージが設定されます。使用されたメモリがスレッシュォルドを超えると、マージが発生します。

RLV ストアと IQ メインストアをマージするかどうかは、テーブル単位で評価されます。ロードされた RLV テーブルのすべてが列挙され、そのそれぞれについてマージが妥当であるかが判定されます。個別のテーブルのマージは、次の場合に妥当であるとみなされます。

1. テーブルがメモリのスレッシュォルド (**RV_MERGE_TABLE_MEMPERCENT**) またはローのスレッシュォルド (**RV_MERGE_TABLE_NUMROWS**) のいずれかを超過した場合、および

2. RLV のローのかなり多くの割合がコミットされていないため、マージができないとシステムにより判断されていない場合

RV_MERGE_TABLE_NUMROWS オプション

RV_MERGE_TABLE_NUMROWS などのマージスレッシュホールドに基づいて発生する、ローレベルのバージョン管理 (RLV) ストアと IQ メインストアとの自動的なマージです。このテーブルスレッシュホールドを超過した場合、特定のテーブルのマージがトリガされます。

指定可能な値
1000 - 100000000

デフォルト
10000000

スコープ
オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。すぐに有効になります。

備考
このオプションにより、RLV 対応テーブルのマージスレッシュホールドとして使用されるローの数が設定されます。使用されたローの数がスレッシュホールドを超えると、マージが発生します。

RLV ストアと IQ メインストアをマージするかどうかは、テーブル単位で評価されます。ロードされた RLV テーブルのすべてが列挙され、そのそれぞれについてマージが妥当であるかどうかが判定されます。個別のテーブルのマージは、次の場合に妥当であるとみなされます。

1. テーブルがメモリのスレッシュホールド (RV_MERGE_TABLE_MEMPERCENT) またはローのスレッシュホールド (RV_MERGE_TABLE_NUMROWS) のいずれかを超過した場合、および
2. RLV のローのかなり多くの割合がコミットされていないため、マージができないとシステムにより判断されていない場合

RV_RESERVED_DBSPACE_MB オプション

重要な操作中に、データ構造で使用されるメモリ用に予約する必要がある、RLV ストアの一部分です。

指定可能な値
50 以上の整数 (メガバイト)

デフォルト

50 Mb または RLV DB 領域の半分のサイズの、いずれか小さい方です。

スコープ

オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。すぐに有効になります。予約領域サイズを変更するためにサーバを再起動する必要はありません。

説明

このオプションを使用すると、セーブポイントの解放操作、コミット操作、ロールバック操作で使用される、小さいが重要なデータ構造体用に RLV ストア内で確保する領域の量を制御できます。

SCALE オプション

カタログストアのクエリに限定して、計算結果が最大 PRECISION にトランケートされる場合に適用する小数点以下の最小桁数を指定します。

指定できる値

整数値 (最大 126)

デフォルト値

38

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。すぐに有効になります。

備考

このオプションでは、カタログストアに対するクエリの計算結果が最大 PRECISION にトランケートされる場合に適用する小数点以下の最小桁数を指定します。

乗算、除算、加算、減算、集合関数はすべて、結果が最大精度を超える可能性があります。

参照：

- MAX_CLIENT_NUMERIC_SCALE オプション (711 ページ)
- PRECISION オプション (742 ページ)

SNAPSHOT_VERSIONING オプション

RLV 対応テーブルのみに適用されます (データベース内のすべてのベーステーブルが対象、とは対照的です)。RLV 対応テーブルへのアクセスを、単一ライタのテーブルレベルバージョン管理を使用して行うのか、複数ライタのローレベルバージョン管理を使用して行うのかを制御します。このオプションは、IQ カタログストアには適用されません。

指定可能な値

値	アクション
row-level	<p>RLV 対応テーブルの同時ライタアクセスおよびローレベルのバージョン管理を有効にする。</p> <p>テーブルのローを変更する最初のトランザクションにより、ローの書き込みロックが確立され、トランザクションの終了まで保持される。</p> <p>後続のトランザクションがロックされたローの変更を試行すると、ロックエラーまたは将来バージョンエラーで失敗する。つまり、BLOCKING オプションの値に従ってロックが開放されるまでブロックされる。</p>
table-level	<p>単一ライタアクセスおよびテーブルレベルのバージョン管理を有効にする。</p> <p>テーブルにアクセスする最初のトランザクションにより、テーブルの書き込みロックが確立され、トランザクションの終了まで保持される。</p> <p>後続のトランザクションがロックされたテーブルへの書き込みを試行すると、ロックエラーまたは将来バージョンエラーで失敗する。つまり、BLOCKING オプションの値に従って、ロックが開放されるまでブロックされる。</p>

デフォルト

table-level

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。すぐに有効になります。

SIGNIFICANTDIGITSFORDOUBLEEQUALITY オプション

2つの複雑な算術式の等価性テストでは、指数表記で小数点の右側に有効桁数を指定する方法が使われます。

指定できる値
0–15

デフォルト値
0

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

double 型のデータは 10 進数 (基数 10) ではなく 2 進数 (基数 2) で格納されるので、この設定では、使用する 10 進有効桁数 (概数) を指定します。0 に指定するとすべての桁が使用されます。

たとえば、SIGNIFICANTDIGITSFORDOUBLEEQUALITY を 12 に設定すると、次の 2 つの数値は等しいとみなされます。13 に設定すると、等しくないとみなされません。

- 1.23456789012345
- 1.23456789012389

SIGNIFICANTDIGITSFORDOUBLEEQUALITY は、2つの複雑な算術式の等価性テストに影響しますが、インデックスによって行われるものには影響しません。

SORT_COLLATION オプション

ORDER BY 式に対して **SORTKEY** 関数の暗黙の使用を許可します。

指定できる値

Internal、照合名または照合 ID

デフォルト値

Internal

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

SORT_COLLATION の値が Internal である場合、ORDER BY 句は変更されません。

このオプションの値を、有効な照合名または照合 ID に設定すると、ORDER BY 句の文字列式は、SORTKEY 関数が呼び出されたものとして扱われます。

関数については、『リファレンス：ビルディングブロック、テーブル、およびプロシージャ』を参照してください。

例

ソート照合をバイナリに設定します。

```
SET TEMPORARY OPTION sort_collation='binary';
```

ソート照合をバイナリに設定すると、次のクエリを変換します。

```
SELECT Name, ID  
FROM Products  
ORDER BY Name, ID;  
SELECT Name, ID  
FROM Products  
ORDER BY 1, 2;
```

クエリは以下のように変換されます。


```
SELECT Name, ID
FROM Products
ORDER BY SORTKEY(Name, 'binary'), ID;
```

SORT_PINNABLE_CACHE_PERCENT オプション

現在使用可能なバッファに対してソートオブジェクトが保持する最大パーセンテージを設定します。

指定できる値
0 - 100

デフォルト値
20

スコープ
オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考
大規模なソートの場合、大きな値を設定することによって、必要なマージフェーズの数が削減される場合があります。その一方で、大きな値を設定すると、同じシステム上で実行される他のユーザのソートやハッシュ処理に影響を及ぼす可能性があります。このオプションを変更する場合、不適切な値を設定するとかえってパフォーマンスを低下させるおそれがあるため、いくつかの設定を試したうえでパフォーマンス向上に最適な値を判断する必要があります。

ヒント： SORT_PINNABLE_CACHE_PERCENT では、デフォルト値を使用してください。

このオプションは、主に テクニカルサポートで使用されます。

SORT_PINNABLE_CACHE_PERCENT の値を変更する場合は、細心の注意が必要です。

SQL_FLAGGER_ERROR_LEVEL オプション [TSQL]

指定された標準の一部ではない SQL コードへの応答動作を制御します。

指定できる値

- OFF
- SQL:1992/EntrySQL:1992/Intermediate
- SQL:1992/Full
- SQL:1999/Core
- SQL:1999/Package
- SQL:2003/Core
- SQL:2003/Package

デフォルト値

OFF

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

指定された標準の一部ではない SQL コードをエラーとして通知します。たとえば、SQL:2003/Package を指定すると、データベースサーバは、上級 SQL/2003 構文でない構文を通知します。

以前のバージョンの SAP Sybase IQ との互換性を確保するため、次の表に示す値も受け入れられ、指定どおりにマップされます。

表 18 : SQL_FLAGGER_ERROR_LEVEL の互換値

値	アクション
E	初級レベル SQL92 構文でない構文を通知する。SQL:1992/Entry に対応する。
I	中級レベル SQL92 構文でない構文を通知する。SQL:1992/Intermediate に対応する。

値	アクション
F	上級レベル SQL92 構文でない構文を通知する。SQL:1992/Full に対応する。
W	サポートされている構文をすべて許可する。OFF に対応する。

SQL_FLAGGER_WARNING_LEVEL オプション [TSQL]

指定された規格の一部ではない SQL への応答を制御します。

指定できる値

- OFF
- SQL:1992/Entry
- SQL:1992/Intermediate
- SQL:1992/Full
- SQL:1999/Core
- SQL:1999/Package
- SQL:2003/Core
- SQL:2003/Package

デフォルト値

OFF

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

警告として、指定された標準の一部ではない SQL コードをエラーとして通知します。たとえば、SQL:2003/Package を指定すると、データベースサーバは、上級 SQL/2003 構文でない構文を通知します。

デフォルトの動作 OFF では、警告の通知がオフになります。

以前のバージョンの SAP Sybase IQ との互換性を確保するため、次の表に示す値も受け入れられ、指定どおりにマップされます。

表 19 : SQL_FLAGGER_WARNING_LEVEL の互換値

値	アクション
E	初級レベル SQL92 構文でない構文を通知する。SQL:1992/Entry に対応する。
I	中級レベル SQL92 構文でない構文を通知する。SQL:1992/Intermediate に対応する。
F	上級レベル SQL92 構文でない構文を通知する。SQL:1992/Full に対応する。
W	サポートされている構文をすべて許可する。OFF に対応する。

STRING_RTRUNCATION オプション [TSQL]

INSERT または **UPDATE** によって CHAR または VARCHAR 文字列がトランケートされるときに、エラーを通知するかどうかを決定します。

指定できる値

ON、OFF

デフォルト値

ON

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

トランケートされる文字がスペースだけで構成されている場合、例外は発生しません。ON は SQL92 動作に準拠します。**STRING_RTRUNCATION** を OFF に設定すると、例外は発生せず、文字列は通知なしにトランケートされます。このオプションが ON の場合、エラーが発生すると **ROLLBACK** が行われます。

SAP Sybase IQ 15.0 より前のバージョンでは、このオプションはデフォルトで OFF に設定されていました。下位互換性のために OFF に設定しても問題はありません。しかし、ON 設定では、トランケーションによってデータが失われる可能性がある文を検出できるため、ON 設定を推奨します。

SUBQUERY_CACHING_PREFERENCE オプション

関連サブクエリの述部の処理に使用するアルゴリズムを制御します。

指定できる値

値	アクション
1	最初のサブクエリの述部にはソートベースの処理を使用する。同じ順序付けキーを持たない他のサブクエリ述部は、ハッシュテーブルを使用して処理し、サブクエリの結果をキャッシュする。
2	妥当な場合、ハッシュテーブルを使用して、すべてのサブクエリ述部の結果をキャッシュする。使用可能なテンポラリキャッシュがサブクエリの結果のすべてに対応できない場合、パフォーマンスが低下する可能性がある。
3	前のサブクエリの結果を1つキャッシュする。 SORT と HASH は使用しない。
0	オプティマイザの選択に従う。
-1	SORT の使用を回避する。妥当な場合、IQ オプティマイザは HASH を選択する。
-2	HASH の使用を回避する。妥当な場合、IQ オプティマイザは SORT またはキャッシュの値を選択する。
-3	キャッシュの値の使用を回避する。妥当な場合、IQ オプティマイザは HASH または SORT を選択する。

デフォルト値

0

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

IQ オプティマイザでは、関連サブクエリの述部に対して、サブクエリの実行コストを減らす外部参照とサブクエリ結果のキャッシュについて選択できます。

SUBQUERY_CACHING_PREFERENCE では、使用するアルゴリズムを選択する際のオプティマイザのコスト決定を無効にできます。クエリエンジンに対してアルゴリズムが適切かどうかを判断するための内部規則を無効にするものではありません。

ゼロ以外の値を設定すると、クエリ内の各サブクエリの述部に影響します。クエリ内のサブクエリの述部の1つを選択し、それにゼロ以外の値を使用することはできません。

SUBQUERY_CACHING_PREFERENCE は、通常、内部テスト目的で、経験豊富なDBAだけが使用します。**IN** サブクエリには適用されません。

参照：

- **IN_SUBQUERY_PREFERENCE** オプション (686 ページ)

SUBQUERY_FLATTENING_PERCENT オプション

ユーザは、オプティマイザがスカラサブクエリのジョインへの変換を決定するスレッシュホールドを変更できます。

指定できる値

値	アクション
0	オプティマイザのコストモデル決定に従う。
1 ~ (2 ³² - 1)	フラット化する参照の割合。

デフォルト値

100

スコープ

このオプションは、相関スカラサブクエリだけに適用されます。オプションは、データベース (**PUBLIC**) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの **PUBLIC** 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、**SET ANY PUBLIC OPTION** システム権限が必要です。個々の接続または **PUBLIC** ロールに一時的に設定できます。すぐに有効になります。**SUBQUERY_FLATTENING_PERCENT** をデフォルト以外の値に設定すると、クエリ内のすべてのスカラサブクエリ述部に影響します。クエリ内の1つのスカラサブクエリ述部を選択し、それに対してこのオプションを使用することはできません。

備考

SAP Sybase IQ クエリオプティマイザは、関連スカラサブクエリを等価のジョイン操作に変換して、クエリのパフォーマンスを向上させることができます。

SUBQUERY_FLATTENING_PERCENT オプションでは、この最適化を行うスレッシュホールドをユーザが調整できます。

SUBQUERY_FLATTENING_PERCENT は、スカラクエリ内にある重複しない予想外部値に対する予想内部値の割合 (%) を表します。予想割合が 100% に近づくと、ジョインとしてのサブクエリの評価コストは、個別のインデックスによる検査よりも低くなります。予想された内部値が予想された外部値よりも少ないという保証はないため、100% より大きな値を設定できます。

参照：

- **SUBQUERY_FLATTENING_PREFERENCE** オプション (779 ページ)

SUBQUERY_FLATTENING_PREFERENCE オプション

スカラまたは **EXISTS** サブクエリをジョインに変換(フラット化)する際のオプティマイザの決定をユーザが無効にできます。

指定できる値

値	アクション
-3	EXISTS およびスカラサブクエリのジョイン操作へのフラット化を回避する。
-2	スカラサブクエリのジョイン操作へのフラット化を回避する。
-1	EXISTS サブクエリのジョイン操作へのフラット化を回避する。
0	IQ オプティマイザによるサブクエリのフラット化の決定を許可する。
1	可能であれば、 EXIST のフラット化のコストを無視する。
2	可能であれば、スカラのフラット化のコストを無視する。
3	EXISTS およびスカラサブクエリのコストを無視する。

デフォルト値

0

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場

合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。オプションをゼロ以外の値に設定すると、クエリ内のすべてのサブクエリ述部に影響します。クエリ内の1つのサブクエリ述部を選択し、それに対してこのオプションを使用することはできません。

備考

SAP Sybase IQ オプティマイザは、相関スカラサブクエリ、**EXISTS** サブクエリ、または **NOT EXISTS** サブクエリを等価のジョイン操作に変換して、クエリのパフォーマンスを向上させることができます。この最適化はサブクエリのフラット化と呼ばれます。SUBQUERY_FLATTENING_PREFERENCE では、使用するアルゴリズムを選択する際のオプティマイザのコストの高い決定を上書きできます。

SUBQUERY_FLATTENING_PREFERENCE を 0 (IQ オプティマイザにサブクエリのフラット化を決定させる) に設定することは、以前のバージョンの SAP Sybase IQ で現在廃止されている **FLATTEN_SUBQUERIES** オプションを ON に設定することと同じです。

参照：

- SUBQUERY_FLATTENING_PERCENT オプション (778 ページ)

SUBQUERY_PLACEMENT_PREFERENCE オプション

クエリプラン内での、相関サブクエリの述語演算子の配置を制御します。

指定できる値

値	アクション
-1	クエリプランのできるかぎり下のほうの場所を選択し、クエリ内でのサブクエリの実行をできるだけ早める。
0	オプティマイザの選択に従う。
1	クエリプランのできるかぎり上のほうの場所を選択し、クエリ内でのサブクエリの実行をできるだけ遅らせる。

デフォルト値

0

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できません。データベースレベルで設定した場合、値は新しいユーザのデフォルト値にな

りますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

IQ オプティマイザは、クエリ内の関連サブクエリの演算子を配置する場所を、そのクエリのプラン内の有効な場所から選択できます。

SUBQUERY_PLACEMENT_PREFERENCE を使用すると、配置場所を選択する際のオプティマイザのコストベースの決定を無効にできます。場所が有効かどうかを確認する内部ルールを無効にすることはありません。一部のクエリでは、有効な選択肢が1つしかないこともあります。0以外の値を設定すると、クエリ内のすべての関連サブクエリの述語に影響します。クエリ内のいくつかのサブクエリから1つを選択してその配置場所を修正することはできません。

通常、このオプションは、内部的なテストに使用されます。また、経験豊富な DBA だけが使用してください。

このオプションでは、ほとんどの場合、デフォルトの設定が最適です。場合によっては、テクニカルサポートがこの値を変更するよう指示することがあります。

SUPPRESS_TDS_DEBUGGING オプション

TDS デバッグ情報をサーバ画面に表示するかどうかを決定します。

指定できる値

ON、OFF

デフォルト値

OFF

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

データベースオプション

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

サーバの起動時に **-z** オプションが指定されていれば、TDS プロトコルのデバッグ情報を含むデバッグ情報がサーバ画面に表示されます。

SUPPRESS_TDS_DEBUGGING オプションは、サーバ画面に表示される TDS 関連のデバッグ情報を制限します。このオプションが OFF (デフォルト値) に設定されている場合、TDS デバッグ情報がサーバ画面に表示されます。

SWEEPER_THREADS_PERCENT オプション

バッファキャッシュのクリーンアップに使用する SAP Sybase IQ スレッドの割合を指定します。

指定できる値

1 - 40

デフォルト値

10

スコープ

オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。変更を有効にするには、データベースを停止し、再起動させてください。

備考

SAP Sybase IQ は処理スレッドのうち数パーセントをスリーパスレッドとして使用します。これらのスリーパスレッドは、メインおよびテンポラリバッファキャッシュのダーティページをクリーンにします。

IQ モニタの **-cache** レポート内の GDirty カラムには、LRU バッファが "dirty" (変更済み) 状態で取り込まれた回数が表示されます。GDirty が一定時間 0 より大きい数であれば、**SWEEPER_THREADS_PERCENT** または

WASH_AREA_BUFFERS_PERCENT の値を大きくする必要がある場合があります。

このオプションでは、ほとんどの場合、デフォルトの設定が最適です。場合によっては、SAP Sybase 製品の保守契約を結んでいるサポートセンタがこの値を大きくするように指示することもあります。

参照：

- **WASH_AREA_BUFFERS_PERCENT** オプション (809 ページ)

TDS_EMPTY_STRING_IS_NULL オプション [database]

TDS 接続で、空の文字列を NULL として返すか、ブランク文字を 1 つ含む文字列として返すかを制御します。

指定できる値
ON、OFF

デフォルト値
OFF

スコープ
オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考
デフォルトでは、TDS_EMPTY_STRING_IS_NULL は OFF に設定され、TSD 接続の場合、空の文字列はブランク文字を 1 つ含む文字列として返されます。このオプションが ON の場合、空の文字列は、TDS 接続では NULL 文字列として返されます。TDS 以外の接続では、空の文字列と NULL 文字列が区別されます。

TEMP_EXTRACT_APPEND オプション

データ抽出機能で抽出されたローを、出力ファイルの末尾に追加するよう指定します。

指定できる値
ON、OFF

デフォルト値
OFF

スコープ
オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値にな

りますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

このオプションは、データ抽出機能で抽出されたローを出力ファイルの末尾に追加するよう指定するために使用します。自分書き込み/実行パーミッションを持っているディレクトリに出力ファイルを作成し、SAP Sybase IQ の起動に使用したユーザ名 (sybase など) に対してそのディレクトリと出力ファイルの書き込みパーミッションを設定します。必要に応じて、出力ファイルのパーミッションを他のユーザに与えることもできます。出力ファイルの名前は **TEMP_EXTRACT_NAME1** オプションで指定します。ファイルが存在しない場合、データ抽出機能によって出力ファイルが作成されます。

TEMP_EXTRACT_APPEND は **TEMP_EXTRACT_SIZE_n** オプションと互換性がありません。抽出追加出力ファイルのサイズを制限しようとする、SAP Sybase IQ からエラーが返されます。

参照：

- TEMP_EXTRACT_NAME_n オプション (788 ページ)

TEMP_EXTRACT_BINARY オプション

TEMP_EXTRACT_SWAP オプションと組み合わせて、データ抽出機能で実行する抽出の種類を指定します。

指定できる値

ON、OFF

デフォルト値

OFF

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

このオプションを TEMP_EXTRACT_SWAP オプションとともに使用することによって、データ抽出機能で実行する抽出の種類を指定できます。

表 20 : 抽出の種類に対する抽出オプション

抽出の種類	TEMP_EXTRACT_BINARY	TEMP_EXTRACT_SWAP
バイナリ	ON	OFF
バイナリ/スワップ	ON	ON
ASCII	OFF	OFF

抽出の種類のリファレンスは ASCII です。

参照：

- TEMP_EXTRACT_SWAP オプション (797 ページ)

TEMP_EXTRACT_COLUMN_DELIMITER オプション

ASCII 抽出のデータ抽出機能の出力で使用するカラムのデリミタを指定します。

指定できる値
文字列

デフォルト値
' '

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

TEMP_EXTRACT_COLUMN_DELIMITER は、データ抽出機能の出力で使用するカラム間のデリミタの指定に使用します。ASCII 抽出の場合、デフォルトではカンマでカラムの値を区切ります。デフォルトでは、文字列は引用符で囲われません。

デリミタには 1 バイトから 4 バイト必要で、マルチバイト照合順を使用している場合はその照合順でデリミタが有効である必要があります。デリミタとして使用する文字列は、データ出力文字列内で使われないものにしてください。

ASCII 抽出で、このオプションが空の文字列 "" に設定されている場合は、抽出されたデータはカラムデリミタなしの固定幅の ASCII で書き込まれます。数値とバイナリデータ型は、*n* ブランクのフィールドに右揃えされます。ここで *n* は、その型の値を表すのに必要な最大バイト数を表します。文字データ型は、*n* ブランクのフィールドで左揃えされます。

注意： 固定幅の ASCII 抽出での最小のカラム幅は、NULL 値の場合に "NULL" という文字列を格納できるよう、4 バイトです。たとえば、抽出されたカラムが CHAR(2) で、**TEMP_EXTRACT_COLUMN_DELIMITER** が空の文字列 "" に設定されている場合、抽出されたデータの後に 2 つのスペースが付きます。

参照：

- TEMP_EXTRACT_QUOTE オプション (792 ページ)
- TEMP_EXTRACT_QUOTES オプション (793 ページ)
- TEMP_EXTRACT_QUOTES_ALL オプション (794 ページ)
- TEMP_EXTRACT_ROW_DELIMITER オプション (795 ページ)

TEMP_EXTRACT_DIRECTORY オプション

ユーザがデータ抽出機能を利用できるかどうかを制御します。また、一時抽出ファイルの配置先ディレクトリを制御し、**TEMP_EXTRACT_NAME_n** オプションで指定されたディレクトリパスを上書きします。

指定できる値
文字列

デフォルト値
" (空の文字列)

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場

合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

あるユーザに対して **TEMP_EXTRACT_DIRECTORY** オプションが FORBIDDEN (大文字と小文字の区別なし) という文字列に設定されている場合、そのユーザはデータ抽出を実行できません。このユーザがデータ抽出機能を使用しようとするときのようなエラーになります。「抽出を実行するためのパーミッションがありません。」

TEMP_EXTRACT_DIRECTORY が PUBLIC ロールに対して FORBIDDEN に設定されている場合は、どのユーザもデータ抽出を実行できません。

TEMP_EXTRACT_DIRECTORY が有効なディレクトリパスに設定されている場合、**TEMP_EXTRACT_NAME_n** オプションで指定されたパスが上書きされ、そのディレクトリパスに一時抽出ファイルが配置されます。

TEMP_EXTRACT_DIRECTORY が無効なディレクトリパスに設定されている場合は、次のようなエラーが発生します。「ファイルが存在しません。<invalid path>」

TEMP_EXTRACT_DIRECTORY が空白の場合、一時抽出ファイルは、**TEMP_EXTRACT_NAME_n** で指定されたディレクトリに配置されます。

TEMP_EXTRACT_NAME_n の一部としてパスが指定されていない場合、抽出ファイルはデフォルトでサーバ起動ディレクトリに配置されます。

このオプションは、大きなデータ抽出ファイルの作成を、ユーザが書き込みアクセスを持つディレクトリに制限することで、より充実したセキュリティを実現し、ディスク管理を制御しやすくします。

データ抽出機能の詳細と抽出オプションの使用法については、『管理：ロード管理』を参照してください。

参照：

- **TEMP_EXTRACT_NAME_n** オプション (788 ページ)

TEMP_EXTRACT_ESCAPE_QUOTES オプション

ASCII 抽出のデータ抽出機能の出力で、引用符が含まれたフィールドのすべての引用符をエスケープするかどうかを指定します。

指定できる値

ON、OFF

デフォルト値
OFF

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

TEMP_EXTRACT_QUOTE がデフォルトのままか、値""(二重引用符)に設定されており、TEMP_EXTRACT_BINARY が OFF であり、TEMP_EXTRACT_QUOTES または TEMP_EXTRACT_QUOTES_ALL が ON である場合を除き、このオプションは無視されます。

参照：

- TEMP_EXTRACT_BINARY オプション (784 ページ)
- TEMP_EXTRACT_QUOTES オプション (793 ページ)
- TEMP_EXTRACT_QUOTES_ALL オプション (794 ページ)

TEMP_EXTRACT_NAME_n オプション

データ抽出機能で使用される出力ファイルまたは名前付きパイプの名前を指定します。オプションは 8 つあります。TEMP_EXTRACT_NAME1 から TEMP_EXTRACT_NAME8 の 8 つです。

指定できる値
文字列

デフォルト値
" (空の文字列)

スコープ

PUBLIC または他のユーザ/ロールに対してこのオプションを設定するには SET ANY PUBLIC OPTION システム権限が必要です。

説明

TEMP_EXTRACT_NAME1 から **TEMP_EXTRACT_NAME8** では、データ抽出機能で使用する出力ファイルの名前を指定します。これらのオプションは順番に使用する必要があります。たとえば、**TEMP_EXTRACT_NAME1** オプションと **TEMP_EXTRACT_NAME2** オプションを先に設定していない状態で **TEMP_EXTRACT_NAME3** を設定した場合、このオプションは無効になります。

この中で最も重要なオプションは **TEMP_EXTRACT_NAME1** です。

TEMP_EXTRACT_NAME1 がデフォルト値 (空の文字列 "") に設定されている場合、抽出処理は使用できず、リダイレクトされる出力もありません。抽出処理を使用可能にするには、**TEMP_EXTRACT_NAME1** にパス名を設定してください。すると、そのパス名のファイルに対して抽出が始まります。パス名には、他で使用されていないファイルを選択してください。

ヒント： **TEMP_EXTRACT_NAME1** オプションは **TEMPORARY** として設定してください。

TEMP_EXTRACT_NAME1 は、**TEMP_EXTRACT_APPEND** オプションを ON に設定したときに、出力ファイル名を指定する目的で使用することもできます。この場合、**SELECT** 文を実行する前に、SAP Sybase IQ の起動に使用されたユーザ名 (sybase など) に対し、指定のファイル、またはこのファイルを含むフォルダの書込みパーミッションを設定します。追加モードの場合、データ抽出機能は、ファイルにすでに書き込まれているデータに上書きするのではなく、抽出されたローをファイルの末尾に追加します。出力ファイルが存在しない場合、データ抽出機能によってファイルが作成されます。

警告！ 既存のファイルのパス名を選択し、**TEMP_EXTRACT_APPEND** オプションを OFF (デフォルト) に設定した場合は、ファイルの内容が上書きされます。これは、ウィークリーレポートなどのファイルであれば問題ありませんが、データベースファイルに対して行ってはなりません。

TEMP_EXTRACT_NAME2 から **TEMP_EXTRACT_NAME8** までのオプションは、**TEMP_EXTRACT_NAME1** に追加して、複数の出力ファイル名を指定するために使用できます。

抽出先のディスクファイルまたは名前付きパイプが 1 つだけの場合は、**TEMP_EXTRACT_NAME2** から **TEMP_EXTRACT_NAME8** と、**TEMP_EXTRACT_SIZE1** から **TEMP_EXTRACT_SIZE8** の各オプションはデフォルト値のままにします。

TEMP_EXTRACT_NAME1 を設定したときには、次の操作は実行できません。

- ジョインのトップテーブルに対する **LOAD**、**DELETE**、**INSERT**、または **INSERT...LOCATION**
- **INSERT...SELECT**

さらに、データ抽出機能の次の制限に留意してください。

- 抽出は、IQ ストアに格納されたデータに対してのみ実行できます。
- 抽出は、システムテーブルやデータベース間ジョインに対しては実行できません。
- 抽出は、ユーザ定義関数やシステム関数 (システム関数 **suser_id()** と **suser_name()** は除く) を使用したクエリでは実行できません。
- コマンドファイルに保存したデータ抽出コマンドを使用して Interactive SQL を **-q** (クワイエットモード) オプション付きで実行する場合は、あらかじめ Interactive SQL オプション [複数の結果セットを表示] を設定し、これを永続化する必要があります。このオプションが設定されていないと、出力ファイルは作成されません。
[複数の結果セットを表示] オプションを設定するには、Interactive SQL ウィンドウの [ツール]-[オプション] を選択し、[複数の結果セットを表示] ボックスをクリックして [永続化する] をクリックします。

TEMP_EXTRACT_NAME_n オプションを使用して指定したディレクトリパスは、**TEMP_EXTRACT_DIRECTORY** オプションを使用して上書きできます。

参照：

- **TEMP_EXTRACT_APPEND** オプション (783 ページ)
- **TEMP_EXTRACT_DIRECTORY** オプション (786 ページ)
- **TEMP_EXTRACT_SIZE_n** オプション (796 ページ)

TEMP_EXTRACT_NULL_AS_EMPTY オプション

ASCII 抽出のデータ抽出機能の出力で使用する null 値の表示方法を指定します。

指定できる値

ON、OFF

デフォルト値

OFF

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

TEMP_EXTRACT_NULL_AS_EMPTY は、ASCII 抽出のデータ抽出機能の出力に含まれる null 値の表示方法を制御します。**TEMP_EXTRACT_NULL_AS_EMPTY** オプションが ON に設定されている場合、すべてのデータ型で null 値が " (空の文字列) で示されます。

上記の引用符は、抽出出力ファイルには含まれないので注意してください。

TEMP_EXTRACT_NULL_AS_EMPTY オプションを OFF にすると、NULL 値の表示に常に文字列 'NULL' が使用されます。デフォルトは OFF です。

TEMP_EXTRACT_NULL_AS_ZERO オプション

ASCII 抽出のデータ抽出機能の出力で使用する null 値の表示方法を指定します。

指定できる値

ON、OFF

デフォルト値

OFF

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

TEMP_EXTRACT_NULL_AS_ZERO は、ASCII 抽出のデータ抽出機能の出力に含まれる null 値の表示方法を制御します。**TEMP_EXTRACT_NULL_AS_ZERO** が ON に設定されている場合、null 値は次のように表示されます。

- 算術型には '0'
- CHAR 文字型と VARCHAR 文字型には " (空文字列)

データベースオプション

- 日付には " (空文字列)
- 時刻には " (空文字列)
- タイムスタンプには " (空文字列)

上記の引用符は、抽出出力ファイルには含まれないので注意してください。

TEMP_EXTRACT_NULL_AS_ZERO オプションを OFF にすると、NULL 値の表示に常に文字列 'NULL' が使用されます。デフォルトは OFF です。

注意： SAP Sybase IQ 12.5 では、テーブル内の CHAR カラムまたは VARCHAR カラムからの ASCII 抽出を行うと、出力ファイルに少なくとも 4 文字が返されます。SAP Sybase IQ は、null 値が含まれるカラム内のローに NULL という単語を書き込む必要があるため、**TEMP_EXTRACT_NULL_AS_ZERO** が OFF に設定されている場合、これが必要となります。**TEMP_EXTRACT_NULL_AS_ZERO** が ON の場合は、4 文字分の領域を確保する必要はありません。

SAP Sybase IQ 12.6 では、**TEMP_EXTRACT_NULL_AS_ZERO** が ON に設定されている場合は、ASCII 抽出で CHAR カラムまたは VARCHAR カラムに対してファイルに書き込まれる文字数は、そのカラムの文字数に等しくなります。4 文字未満のカラムの場合でも同じです。

TEMP_EXTRACT_QUOTE オプション

TEMP_EXTRACT_QUOTES オプションまたは **TEMP_EXTRACT_QUOTES_ALL** オプションが ON に設定されているときに、ASCII 抽出のデータ抽出機能の出力のフィールドを囲むための引用符として使用する文字列を指定します。

指定できる値
文字列

デフォルト値
" (空の文字列)

スコープ
オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

このオプションでは、ASCII 抽出のデータ抽出機能の出力のフィールドを囲むための引用符として使用する文字列を指定します (デフォルトの文字列が適切でない場合)。**TEMP_EXTRACT_QUOTE** は、**TEMP_EXTRACT_QUOTES** オプションまたは **TEMP_EXTRACT_QUOTES_ALL** オプションとともに使用します。

TEMP_EXTRACT_QUOTE オプションで指定する引用符文字列には、ローとカラムのデリミタと同じ制限があります。このオプションのデフォルトは空文字列です。SAP Sybase IQ はこれを一重引用符に変換します。

TEMP_EXTRACT_QUOTE オプションに指定する文字列は、1～4 バイトの長さでなければなりません。また、マルチバイト照合順を使用している場合は、その照合順で有効な文字列でなければなりません。使用する文字列は、データ出力文字列内で使われないものにしてください。

参照：

- **TEMP_EXTRACT_COLUMN_DELIMITER** オプション (785 ページ)
- **TEMP_EXTRACT_QUOTES** オプション (793 ページ)
- **TEMP_EXTRACT_QUOTES_ALL** オプション (794 ページ)
- **TEMP_EXTRACT_ROW_DELIMITER** オプション (795 ページ)

TEMP_EXTRACT_QUOTES オプション

ASCII 抽出のデータ抽出機能の出力で、文字列フィールドを引用符で囲むよう指定します。

指定できる値

ON、OFF

デフォルト値

OFF

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、**SET ANY PUBLIC OPTION** システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

このオプションは、ASCII 抽出のデータ抽出機能の出力で文字列フィールドを引用符で囲むことを指定します。引用符として使用する文字列は、**TEMP_EXTRACT_QUOTE** オプションで指定します (デフォルトの文字列が適切でない場合)。

参照：

- TEMP_EXTRACT_COLUMN_DELIMITER オプション (785 ページ)
- TEMP_EXTRACT_QUOTES_ALL オプション (794 ページ)
- TEMP_EXTRACT_ROW_DELIMITER オプション (795 ページ)

TEMP_EXTRACT_QUOTES_ALL オプション

ASCII 抽出のデータ抽出機能の出力で、すべてのフィールドを引用符で囲むよう指定します。

指定できる値

ON、OFF

デフォルト値

OFF

スコープ

PUBLIC または他のユーザ／ロールに対してこのオプションを設定するには SET ANY PUBLIC OPTION システム権限が必要です。

備考

TEMP_EXTRACT_QUOTES_ALL は、ASCII 抽出のデータ抽出機能の出力で、すべてのフィールドを引用符で囲むことを指定します。引用符として使用する文字列は、**TEMP_EXTRACT_QUOTE** オプションで指定します (デフォルトの文字列が適切でない場合)。

参照：

- TEMP_EXTRACT_COLUMN_DELIMITER オプション (785 ページ)
- TEMP_EXTRACT_QUOTES オプション (793 ページ)
- TEMP_EXTRACT_QUOTES_ALL オプション (794 ページ)
- TEMP_EXTRACT_ROW_DELIMITER オプション (795 ページ)

TEMP_EXTRACT_ROW_DELIMITER オプション

ASCII 抽出のデータ抽出機能の出力で使用するローのデリミタを指定します。

指定できる値
文字列

デフォルト値
" (空の文字列)

スコープ
オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

TEMP_EXTRACT_ROW_DELIMITER では、データ抽出機能の出力で使用するロー間のデリミタを指定します。ASCII 抽出のデフォルトでは、UNIX プラットフォームの場合は改行、Windows プラットフォームの場合は復帰/改行の組み合わせでローが終了します。

デリミタには 1 バイトから 4 バイトが必要です。また、マルチバイト照合順を使用している場合は、その照合順で有効な文字列でなければなりません。デリミタとして使用する文字列は、データ出力文字列内で使われないものにしてください。

TEMP_EXTRACT_ROW_DELIMITER オプションのデフォルトは空の文字列です。SAP Sybase IQ は、このオプションの空文字列を、UNIX プラットフォームでは改行文字、Windows プラットフォームでは復帰/改行文字にそれぞれ変換します。

参照：

- TEMP_EXTRACT_COLUMN_DELIMITER オプション (785 ページ)
- TEMP_EXTRACT_QUOTES オプション (793 ページ)
- TEMP_EXTRACT_QUOTES_ALL オプション (794 ページ)

TEMP_EXTRACT_SIZE_n オプション

データ抽出機能で使用される、対応する出力ファイルの最大サイズを指定します。

指定可能な値

オプションは 8 つあります。TEMP_EXTRACT_SIZE1 から TEMP_EXTRACT_SIZE8 の 8 つです。

デバイスタイプ	サイズ
ディスクファイル	AIX と HP-UX : 0 ~ 64GB Sun Solaris と Linux : 0 ~ 512GB Windows : 0 ~ 128GB
テープ*	524288KB (0.5GB)
その他	9007199254740992KB (8192 ペタバイト、「無制限」)

*テープデバイスは現時点ではサポートされていません。

デフォルト値

0

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

TEMP_EXTRACT_SIZE1 から TEMP_EXTRACT_SIZE8 を使用して、データ抽出機能で使用する対応する出力ファイルの最大サイズを指定します。

TEMP_EXTRACT_SIZE1 では、TEMP_EXTRACT_NAME1 で指定した出力ファイルの最大サイズを指定し、TEMP_EXTRACT_SIZE2 では、TEMP_EXTRACT_NAME2 で指定した出力ファイルの最大サイズを指定します (以下、同様)。

JFS2 など、大規模なファイルシステムでデフォルト値よりも大きなサイズをサポートしている場合は、**TEMP_EXTRACT_SIZE_n** をそのシステムで許容される値に設定します。たとえば、ITB をサポートするには、次のように設定します。

```
TEMP_EXTRACT_SIZE1 = 1073741824 KB
```

抽出先のディスクファイルまたは名前付きパイプが 1 つだけの場合は、**TEMP_EXTRACT_NAME2** から **TEMP_EXTRACT_NAME8** と、**TEMP_EXTRACT_SIZE1** から **TEMP_EXTRACT_SIZE8** の各オプションはデフォルト値のままにします。

TEMP_EXTRACT_SIZE_n オプションは、**TEMP_EXTRACT_APPEND** オプションと互換性がありません。抽出追加出力ファイルのサイズを制限しようとする、SAP Sybase IQ からエラーが返されます。

参照：

- TEMP_EXTRACT_NAME_n オプション (788 ページ)

TEMP_EXTRACT_SWAP オプション

TEMP_EXTRACT_BINARY オプションと組み合わせて、データ抽出機能で実行する抽出の種類を指定します。

指定できる値

ON、OFF

デフォルト値

OFF

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

このオプションを **TEMP_EXTRACT_BINARY** オプションとともに使用することによって、データ抽出機能で実行する抽出の種類を指定できます。

表 21 : 抽出の種類に対する抽出オプション

抽出の種類	TEMP_EXTRACT_BINARY	TEMP_EXTRACT_SWAP
バイナリ	ON	OFF
バイナリ/スワップ	ON	ON
ASCII	OFF	OFF

抽出の種類のリフォルトは ASCII です。

参照 :

- TEMP_EXTRACT_BINARY オプション (784 ページ)

TEMP_RESERVED_DBSPACE_MB オプション

SAP Sybase IQ がテンポラリ IQ ストアに予約する領域の量を制御します。

指定できる値

200 またはそれ以上の整数 (メガバイト)

デフォルト値

200; SAP Sybase IQ は、IQ_SYSTEM_TEMP の最新の読み書きファイルの 1% ~ 50% の領域を予約します。

スコープ

オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。すぐに有効になります。予約領域サイズを変更するためにサーバを再起動する必要はありません。

備考

TEMP_RESERVED_DBSPACE_MB では、SAP Sybase IQ が、セーブポイントの解放、コミット、およびチェックポイント操作で使用される、小さいが重要な特定のデータ構造体のためにテンポラリ IQ ストア内に確保する領域の量を制御できます。運用データベースでは、この値を 200MB から 1GB の間で設定します。IQ ページサイズや同時接続数が大きくなればなるほど、より多くの領域を予約する必要があります。

予約領域サイズは、IQ_SYSTEM_TEMP の最新読み書きファイルの 1% ~ 50% の領域として計算されます。

TEMP_SPACE_LIMIT_CHECK オプション

カタログストアのテンポラリ領域を、接続ごとにチェックします。

指定できる値

ON または OFF (制限チェックなし)

デフォルト値

ON

スコープ

オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。すぐに有効になります。

備考

TEMP_SPACE_LIMIT_CHECK が ON の場合、データベースサーバは、接続が使用するカタログストアのテンポラリファイル領域の容量をチェックします。このオプションが OFF に設定されている場合、接続がそのクォータを超えるテンポラリファイル領域を要求すると、致命的なエラーが発生することがあります。このオプションが ON に設定されている場合、接続がそのクォータを超えるテンポラリファイル領域を要求すると、その要求は失敗し、「テンポラリ領域の制限を超えています」というエラーが返されます。

接続のテンポラリファイルのクォータを決定するには、テンポラリファイルの最大サイズ、そしてアクティブなデータベース接続の2つの要素が使用されます。テンポラリファイルの最大サイズは、ファイルの現在のサイズと、そのファイルが含まれるパーティションの空き領域を合計したサイズです。制限チェックがオンになっているとき、テンポラリファイルが最大サイズの80%を超えている場合に接続リクエストがテンポラリファイル領域をリクエストすると、サーバは接続がクォータをオーバーしていないかどうかチェックします。このとき、最大テンポラリファイル領域をアクティブな接続数で割ったサイズ以上の領域を使用している接続はすべて失敗します。

注意： このオプションは、IQ テンポラリストア領域とは無関係です。IQ テンポラリ領域の増大を制限するには、**QUERY_TEMP_SPACE_LIMIT** オプションと **MAX_TEMP_SPACE_PER_CONNECTION** オプションを使用します。

テンポラリファイルに使用できる領域に関する情報は、**sa_disk_free_space** system プロシージャを使用して取得できます。

例

100MB の空き領域を持ち、他にアクティブなファイルがないドライブ上で、テンポラリファイルと共にデータベースは起動されます。つまり、利用可能なテンポラリファイル領域は 100MB です。DBA は次の文を発行できます。

```
SET OPTION PUBLIC.TEMP_SPACE_LIMIT_CHECK = 'ON'
```

テンポラリファイル領域が 80MB に満たない場合、サーバは以前と変わらず動作します。80MB に達すると、新しい動作が発生する可能性があります。10 個のクエリが実行中で、テンポラリファイルのサイズを増やす必要がある場合を考えてみましょう。あるクエリが、テンポラリファイル領域のうち 8MB 以上を使用していることをサーバが検出すると、そのクエリは失敗します。

TEXT_DELETE_METHOD オプション

TEXT インデックスの削除処理で使用するアルゴリズムを指定します。

指定可能な値

0 ~ 2

デフォルト

0

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

TEXT インデックスを使用するには、非構造化データ分析オプションのライセンスを取得している必要があります。

TIME_FORMAT オプション

データベースから取り出した時刻に対して使用するフォーマットを設定します。

指定できる値

HH、NN、MM、SS を使用してコロンで区切って作られた文字列

デフォルト値

'HH:NN:SS.SSS'

Open Client および JDBC 接続の場合のデフォルトも、HH:NN:SS.SSS に設定されます。

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

フォーマットは次の記号を組み合わせた文字列です。

- hh - 2 桁の時間 (24 時間表示)
- nn - 2 桁の分
- mm - コロンの後の場合は、2 桁の分 (hh:mm など)
- ss[.s...s] - 2 桁の秒と、オプションで小数

各記号は、フォーマット対象の日付の対応するデータに置き換えられます。数字の出力ではなく文字を表すフォーマット記号は大文字で入力でき、その場合、置き換えられる文字も大文字になります。フォーマット文字列に大文字と小文字を混在させた場合、数字の先頭に 0 が付加されません。

フォーマット文字列の中でマルチバイト文字は使用できません。データベースの照合順が 932JPN などのマルチバイト照合順であっても、使用できるのは 1 バイト文字だけです。

参照：

- DATE_FORMAT オプション (639 ページ)

- RETURN_DATE_TIME_AS_STRING オプション (761 ページ)

TIMESTAMP_FORMAT オプション

データベースから取り出したタイムスタンプに対して使用するフォーマットを設定します。

指定できる値

下記の記号を組み合わせた文字列

デフォルト値

'YYYY-MM-DD HH:NN:SS.SSS'

スコープ

オプションは、データベース (PUBLIC) レベルまたは ユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

フォーマットは次の記号を組み合わせた文字列です。

表 22 : TIMESTAMP_FORMAT の文字列記号

記号	説明
yy	2 桁の年。
yyyy	4 桁の年。
mm	2 桁の月、またはコロンの後の場合 (hh:mm など) は 2 桁の分。
mmm	3 文字から成る省略形の月名。
mmmm[m...]	長形式の月名。指定された m の数が月名の文字数を超えるまで、m と同じ数の文字数が使用される。
dd	2 桁の日。
ddd	3 文字から成る省略形の曜日名。

記号	説明
dddd[d...]	長形式の曜日名。指定された d の数が曜日名の文字数を超えるまで d と同じ数の文字数が使用される。
hh	2 桁の時間。
nn	2 桁の分。
ss.SSS	秒 (ss) と小数 (SSS)。最大で小数点以下 6 桁まで。小数点以下 6 桁のタイムスタンプをサポートしていないプラットフォームもある。
aa	a.m. または p.m. (12 時間表記)。
pp	必要であれば p.m. (12 時間表記)。

各記号は、フォーマットされた日付の対応するデータで置き換えられます。数字の出力ではなく文字を表すフォーマット記号は大文字で入力でき、その場合、置き換えられる文字も大文字になります。数字の場合、フォーマット文字列に大文字と小文字を混在させると、出力に先行ゼロが付きません。

フォーマット文字列の中でマルチバイト文字は使用できません。データベースの照合順が 932JPN などのマルチバイト照合順であっても、使用できるのは 1 バイト文字だけです。

参照：

- DATE_FORMAT オプション (639 ページ)
- RETURN_DATE_TIME_AS_STRING オプション (761 ページ)

TOP_NSORT_CUTOFF_PAGES オプション

TOP N アルゴリズムを選択した場合の結果サイズのスレッシュホールドを設定します。

指定できる値

1 - 1000

デフォルト値

1

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

TOP_NSORT_CUTOFF_PAGES では、TOP 句と ORDER BY 句の両方を含む評価で順序付きリストベースの処理からソートベースの処理への切り替えを行うスレッシュホールドをページ数で設定します。TOP N の値が結果ローの数より小さい場合には、順序付きリスト処理のほうがパフォーマンスが向上します。TOP N の値が大きい場合には、ソートベースの処理のほうがパフォーマンスが向上します。

場合によっては、**TOP_NSORT_CUTOFF_PAGES** の値を大きくしてソートベースの処理を回避することによって、パフォーマンスが向上することもあります。

参照：

- SELECT 文 (523 ページ)

TRIM_PARTIAL_MBC オプション

部分的なマルチバイト文字データの自動削除を有効にします。

指定できる値

ON、OFF

デフォルト値

OFF

スコープ

オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。すぐに有効になります。

備考

シングルバイトとマルチバイト文字の両方を含む照合で、一貫したデータロードを実現します。**TRIM_PARTIAL_MBC** が ON の場合、次の処理が行われます。

- 部分的なマルチバイト文字が、CHAR カラムへのロード時にブランクに置き換えられます。
- 部分的なマルチバイト文字が、VARCHAR カラムへのロード時にトランケートされます。

TRIM_PARTIAL_MBC が OFF の場合は、通常の CONVERSION_ERROR セマンティックが適用されます。

参照：

- `CONVERSION_ERROR` オプション [TSQL] (625 ページ)

TRUSTED_CERTIFICATES_FILE オプション

LDAP ユーザ認証による送信トランスポートレイヤセキュリティ (TLS) 接続、INC 接続、および MIPC 接続のための信頼関係を指定します。

指定可能な値

サーバ証明書に署名する信頼された認証局のリストを含む TXT ファイルの場所を示す有効なネットワークパス。

デフォルト

NULL。信頼された認証局がないため送信 TLS 接続を開始できないことを意味します。

スコープ

オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SECURITY OPTION システム権限が必要です。すぐに有効になります。

備考

このオプションは、信頼された認証局のリストの場所を示すパスを指定します。リストは、TXT ファイルに保管する必要があります。このファイルは、Windows 環境内でそのマシン上のすべての SAP Sybase アプリケーションによって使用されるローカルドライブ上の場所で共有できます。

TSQL_VARIABLES オプション [TSQL]

@ 符号を、Embedded SQL ホスト変数名のプレフィクスとして使用できるかどうかを制御します。

指定できる値

ON、OFF

デフォルト値

OFF

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場

データベースオプション

合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

TSQL_VARIABLES を ON に設定することによって、Embedded SQL のホスト変数名のプレフィクスとしてコロンの代わりに @ 符号を使用できます。これは、主に Open Server Gateway との互換性を保つために実装されています。

USER_RESOURCE_RESERVATION オプション

現在のユーザ数に対するメモリの使用を調整します。

指定できる値
整数

デフォルト
1

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

SAP Sybase IQ は、開いたカーソルの数を追跡して、メモリを割り付けます。特定の状況においては、製品を現在使用していると SAP Sybase IQ がみなす現在のカーソル数の最小値を調整して、テンポラリキャッシュからのメモリの割り付けを節約できます。

このオプションは、慎重に分析したうえで、設定が実際に必要であると判断された場合にのみ設定してください。このパラメータを設定する必要がある場合は、詳細をテクニカルサポートにご連絡ください。

VERIFY_PASSWORD_FUNCTION オプション

パスワードルールの実装に使用できる、ユーザ指定の認証の関数を指定します。

指定できる値
文字列

デフォルト
" (空の文字列)。 (パスワードが設定されている場合、関数は呼び出されません)

スコープ
オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY SECURITY OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

VERIFY_PASSWORD_FUNCTION オプションの値が有効な文字列に設定されている場合、**GRANT CONNECT TO *userid* IDENTIFIED BY *password*** 文によってオプション値で指定された関数が呼び出されます。

ユーザが関数を上書きするのを防ぐために、オプション値は、*owner.function_name* の形式で指定する必要があります。

関数は、次の 2 つのパラメータを取ります。

- *user_name* VARCHAR(128)
- *new_pwd* VARCHAR(255)

戻り値のデータ型は、VARCHAR(255) です。

VERIFY_PASSWORD_FUNCTION が設定されている場合、**GRANT CONNECT** 文で複数のユーザ ID とパスワードを指定することはできません。

例

次のサンプルコードは、テーブルと関数を定義し、いくつかのログインポリシーオプションを設定します。同時に、パスワードにおける特定の種類の文字の要求、パスワードの再使用の禁止、パスワードの有効期限の適用などを含む詳細なパスワード規則も実装します。ユーザ ID の作成時、またはパスワードの変更時には、

データベースオプション

verify_password_function オプションを使用してデータベースサーバによってこの関数が呼び出されます。アプリケーションでは post_login_procedure オプションによって指定されるプロシージャが呼び出され、有効期限前にパスワードの変更が必要であることを通知することができます。

```
-- only DBA should have privileges on this table
CREATE TABLE DBA.t_pwd_history(
    pk          INT          DEFAULT AUTOINCREMENT PRIMARY KEY,
    user_name   CHAR(128),   -- the user whose password is set
    pwd_hash    CHAR(32) );  -- hash of password value to detect
                                -- duplicate passwords

-- called whenever a non-NULL password is set
-- to verify the password conforms to password rules
CREATE FUNCTION DBA.f_verify_pwd( uid      VARCHAR(128),
                                new_pwd   VARCHAR(255) )
RETURNS VARCHAR(255)
BEGIN
    -- enforce password rules
    -- enforce minimum length (can also be done with
    -- min_password_length option)
    IF length( new_pwd ) < 6 THEN
        RETURN 'password must be at least 6 characters long';
    END IF;

    -- number of lowercase characters IN new_pwd
    SELECT count(*) INTO num_lower_chars
        FROM pwd_chars WHERE CAST( c AS BINARY ) BETWEEN 'a' AND 'z';

    -- enforce rules based on characters contained in new_pwd
    IF ( SELECT count(*) FROM pwd_chars WHERE c BETWEEN '0' AND '9' )
        < 1 THEN
        RETURN 'password must contain at least one numeric digit';
    ELSEIF length( pwd_alpha_only ) < 2 THEN
        RETURN 'password must contain at least two letters';
    ELSEIF num_lower_chars = 0
        OR length( pwd_alpha_only ) - num_lower_chars = 0 THEN
        RETURN 'password must contain both upper- and lowercase
characters';
    END IF;

    -- not the same as any user name
    -- (this could be modified to check against a disallowed words
table)
    IF EXISTS( SELECT * FROM SYS.SYSUSER
                WHERE lower( user_name ) IN
( lower( pwd_alpha_only ),
                                lower( new_pwd ) ) ) THEN
        RETURN 'password or only alphabetic characters in password '
||
        'must not match any user name';
    END IF;

    -- not the same as any previous password for this user
    IF EXISTS( SELECT * FROM t_pwd_history
```

```

        WHERE user_name = uid
              AND pwd_hash = hash( uid || new_pwd, 'md5' ) ) THEN
    RETURN 'previous passwords cannot be reused';
END IF;

-- save the new password
INSERT INTO t_pwd_history( user_name, pwd_hash )
    VALUES( uid, hash( uid || new_pwd, 'md5' ) );

RETURN( NULL );
END;

ALTER FUNCTION DBA.f_verify_pwd SET HIDDEN;
GRANT EXECUTE ON DBA.f_verify_pwd TO PUBLIC;
SET OPTION PUBLIC.verify_password_function = 'DBA.f_verify_pwd';

-- All passwords expire in 180 days. Expired passwords can be changed
-- by the user using the NewPassword connection parameter.
ALTER LOGIN POLICY DEFAULT password_life_time = 180;

-- If an application calls the procedure specified by the
-- post_login_procedure option, then the procedure can be used to
-- warn the user that their password is about to expire. In
particular,
-- Interactive SQL calls the post_login_procedure.
ALTER LOGIN POLICY DEFAULT password_grace_time = 30;

```

オプションをオフにするには、空の文字列を指定します。

```
SET OPTION PUBLIC.VERIFY_PASSWORD_FUNCTION = ''
```

参照：

- ALTER FUNCTION 文 (22 ページ)
- GRANT CONNECT 文 (385 ページ)

WASH_AREA_BUFFERS_PERCENT オプション

ウォッシュマーカを超えるバッファキャッシュの割合を指定します。

指定できる値
1 - 100

デフォルト値
20

スコープ
オプションは、データベース (PUBLIC) レベルでのみ設定できます。

このオプションを設定するには、SET ANY SYSTEM OPTION システム権限が必要です。変更を有効にするには、データベースを停止し、再起動させてください。

備考

SAP Sybase IQ のバッファキャッシュは、長い MRU/LRU チェーンとして構成されます。ウォッシュマーカを超える領域は、ダーティページをディスクにクリーンアップ (書き込み) するために使用されます。

IQ モニタの **-cache** レポート内の Gdirty カラムには、LRU バッファが "dirty" (変更済み) 状態で取り込まれた回数が表示されます。GDirty が一定時間 0 より大きい数であれば、**SWEEPER_THREADS_PERCENT** または **WASH_AREA_BUFFERS_PERCENT** の値を大きくする必要がある場合があります。

注意： このオプションを変更する前に、**CACHE_AFFINITY_PERCENT** オプションの値を確認してください。**WASH_AREA_BUFFERS_PERCENT** はバッファキャッシュの LRU 側に影響し、**CACHE_AFFINITY_PERCENT** は MRU 側に影響します。これら 2 つの値の合計が 100% を超えないようにする必要があります。

このオプションでは、ほとんどの場合、デフォルトの設定が最適です。場合によっては、SAP Sybase 製品の保守契約を結んでいるサポートセンタがこの値を大きくするように指示することもあります。

参照：

- **SWEEPER_THREADS_PERCENT** オプション (782 ページ)

WAIT_FOR_COMMIT オプション

データが操作されるときに、いつ外部キー整合性をチェックするかを決定します。

指定できる値

ON、OFF

デフォルト値

OFF

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

このオプションを ON に設定すると、データベースは次の **COMMIT** 文まで外部キー整合性をチェックしません。OFF の場合は、**CHECK ON COMMIT** オプションで作成されていないすべての外部キーが、挿入、更新、削除時にチェックされます。

WD_DELETE_METHOD オプション

WD インデックスの削除処理で使用するアルゴリズムを指定します。

指定できる値

値	アクション
0	削除方法はコストモデルにより選択される。コストモデルが選択するのは、ミッドデリートとラージデリートのどちらか一方だけ。
1	スモールデリートが強制される。削除されるローの数が、テーブルの全ロー数に比べて非常に少ないときは、スモールデリートが便利。スモールデリートはインデックスにランダムにアクセス可能、大きいデータセットでキャッシュがスラッシングされる。
2	ラージデリートが強制される。このアルゴリズムは、削除するロー検索のため全インデックスをスキャンする。削除されるローの数が、テーブルの全ロー数に比べてかなり多いときは、ラージデリートが便利。
3	ミッドデリートが強制される。ミッドデリートはインデックスに順番にアクセスするスモールデリートの変形版であり、一般にスモールデリートより高速。

デフォルト値

0

スコープ

オプションは、データベース (PUBLIC) レベルまたはユーザレベルで設定できます。データベースレベルで設定した場合、値は新しいユーザのデフォルト値になりますが、既存のユーザには影響を与えません。ユーザレベルで設定した場合は、そのユーザの PUBLIC 値のみが上書きされます。自分のオプションを設定する場合は、システム権限は必要ありません。自分以外のユーザのオプションをデータベースレベルまたはユーザレベルで設定する場合は、システム権限が必要です。

このオプションを設定するには、SET ANY PUBLIC OPTION システム権限が必要です。個々の接続または PUBLIC ロールに一時的に設定できます。すぐに有効になります。

備考

WD_DELETE_METHOD では、WD インデックスの削除処理で使用するアルゴリズムを指定します。このオプションを設定しないか 0 に設定した場合、削除方法はコストモデルにより選択されます。コストモデルは、適切な削除アルゴリズムを選択する際に、CPU 関連のコストと I/O 関連のコストを考慮します。コストモデルで考慮されるのは以下の要素です。

- 削除されるロー
- インデックスサイズ
- インデックスデータ型の幅
- インデックスデータのカーディナリティ
- 利用可能なテンポラリキャッシュ
- マシンに関連する I/O と CPU の特性
- 利用可能な CPU とスレッド

例

次の文では、WD インデックスからのラージデリートが強制されます。

```
SET TEMPORARY OPTION WD_DELETE_METHOD = 2
```


索引

A

AES 暗号化アルゴリズム

CREATE DATABASE 文 137

AGGREGATION_PREFERENCE オプション
602

ALLOCATE DESCRIPTOR 文
構文 7

ALLOW_NULLS_BY_DEFAULT オプション
603

ALLOW_READ_CLIENT_FILE オプション 610

ALLOW_SNAPSHOT_VERSIONING オプシ
ョン 604

ALTER AGENT 文
構文 9

ALTER DATABASE UPGRADE 文 10

ALTER DATABASE 文
構文 10

ALTER DBSPACE 文
構文 13

ALTER DOMAIN 文
構文 18

ALTER EVENT 文
構文 20

ALTER FUNCTION 文
構文 22

ALTER INDEX 文
エラー 24

ALTER LDAP SERVER 文 28

ALTER LOGICAL SERVER 文
構文 31

ALTER LOGIN POLICY 文
構文 32

ALTER LS POLICY 文
構文 42

ALTER MULTIPLEX RENAME 文 45

ALTER MULTIPLEX SERVER 文 46

ALTER PROCEDURE 文
構文 48

ALTER ROLE 文 50

ALTER SERVER 文
構文 54

ALTER SERVICE 文
構文 57

ALTER TABLE 文
構文 67

ALTER TEXT CONFIGURATION
構文 87

ALTER TEXT INDEX
構文 86

ALTER USER 文 92

ALTER VIEW 文
RECOMPILE 67
構文 96, 99

ANSI_CLOSE_CURSORS_AT_ROLLBACK オ
プション 604

ANSI_PERMISSIONS オプション 605

ANSI_SUBSTRING オプション 607

ANSI_UPDATE_CONSTRAINTS オプション
608

ANSINULL オプション 606

ASE_BINARY_DISPLAY
データベースオプション 610

ASE_FUNCTION_BEHAVIOR
HEXTOINT との互換性 611
INTTOHEX との互換性 611
データベースオプション 611

AT 句

CREATE EXISTING TABLE 163

AUDITING オプション 612

AUTOINCREMENT カラムのデフォルト 257

B

B ツリーページ 616

BACKUP DATABASE 文
アーカイブデバイスの数 100
構文 100

BASE_TABLES_IN_RLV オプション 613

BEGIN DECLARE SECTION 文
構文 296

BEGIN PARALLEL IQ 文 111, 278

索引

- BEGIN TRANSACTION 文
 - Transact-SQL 113
- BEGIN... 108
- BLOCKING オプション 615, 617
- BLOCKING_TIMEOUT オプション 616
- BREAK 文
 - Transact-SQL 583
- BT_PREFETCH_MAX_MISS オプション 616
- BTREE_PAGE_SPLIT_PAD_PERCENT オプション 618
- BYE 文
 - 構文 355
- C**
- CACHE_PARTITIONS オプション 620
- CALL 文
 - Transact-SQL 351
 - 構文 115
- CASE 文
 - 構文 118
- CHAINED オプション 621
- CHECK ON COMMIT 句
 - 参照整合性 257
- CHECK 条件
 - 説明 257
- CHECKPOINT 文
 - 構文 120
- CHECKPOINT_TIME オプション 622
- CIS
 - リモートデータアクセス 623
- CIS_ROWSET_SIZE オプション
 - 説明 623
- CLEAR 文
 - 構文 121
- CLOSE 文
 - 構文 122
- CLOSE_ON_ENDTRANS オプション 623
- COMMENT 文
 - 構文 123
- COMMIT TRANSACTION 文
 - Transact-SQL 129
- COMMIT 文
 - 構文 129
- CONFIGURE 文
 - 構文 132
- CONNECT 権限
 - GRANT 文 385
- CONNECT 文
 - 取り消し 501
 - 構文 133
- connection_property 関数
 - 説明 586
- contains-expression
 - FROM 句 372
- CONTINUE 文
 - Transact-SQL 583
- CONTINUE_AFTER_RAISERROR オプション 624
- CONVERSION_ERROR オプション 625
- CONVERSION_MODE オプション 626
- CONVERT_VARCHAR_TO_1242 オプション 632
- COOPERATIVE_COMMIT_TIMEOUT オプション 633
- COOPERATIVE_COMMITS オプション 634
- CREATE AGENT 文
 - 構文 136
- CREATE DATABASE 文
 - 構文 137
- CREATE DBSPACE 文
 - 構文 150
- CREATE DOMAIN 文
 - 構文 154
- CREATE EVENT 文
 - 構成 157
- CREATE EXISTING TABLE 文
 - プロキシテーブル 163
- CREATE EXTERNLOGIN 文
 - INSERT...LOCATION 417
 - 構文 166
- CREATE FUNCTION 文
 - Java 175
 - UDF 175
 - 外部環境 175
 - 構文 169
- CREATE INDEX 文 111
 - テーブルでの使用 179
 - 構文 179
- CREATE LDAP SERVER 文 188

CREATE LOGICAL SERVER 文 192
 CREATE LOGIN POLICY 文
 構文 194
 CREATE LS POLICY 文
 構文 203
 CREATE MESSAGE 文
 Transact-SQL 206
 CREATE MULTIPLEX SERVER 文 208
 CREATE ON 文
 取り消し 502
 CREATE PROCEDURE 文
 Transact-SQL 218
 構文 210
 CREATE ROLE 文 235
 CREATE SCHEMA 文
 構文 238
 CREATE SERVER 文
 INSERT...LOCATION 417
 構文 241
 CREATE SERVICE 文
 構文 244
 CREATE TABLE 文
 構文 257
 CREATE TEXT CONFIGURATION
 構文 276
 CREATE TEXT INDEX
 構文 278
 CREATE USER 文 287
 CREATE VARIABLE 文
 構文 289
 CREATE VIEW 文
 構文 292
 CREATE 文
 付与 388
 CREATE_HG_AND_FORCE_PHYSICAL_DEL
 ETE 636
 CREATE_HG_WITH_EXACT_DISTINCTS 635
 CUBE 演算子 523
 SELECT 文 523
 CURSOR_WINDOW_ROWS オプション 637

D

DATE_FIRST_DAY_OF_WEEK オプション 638
 DATE_FORMAT オプション 639

DATE_ORDER オプション 641
 DB 領域をオンラインに設定 13
 DB 領域
 オフラインに設定 13
 作成 150
 削除 319
 変更 13
 DBCC_LOG_PROGRESS
 データベースオプション 642
 DBCC_PINNABLE_CACHE_PERCENT
 データベースオプション 643
 dbisql
 オプション 545
 データベースへの接続 133
 dbo ユーザ ID
 ビューの所有者 319
 dbspaces
 仮想バックアップ 100
 DEALLOCATE DESCRIPTOR
 構文 295
 DEBUG_MESSAGES オプション
 説明 643
 DECLARE CURSOR 文
 Transact-SQL 構文 305
 構文 299
 DECLARE LOCAL TEMPORARY TABLE 文
 構文 306
 DECLARE TEMPORARY TABLE 文
 構文 306
 DECLARE 文
 構文 108, 297
 DEDICATED_TASK オプション
 説明 644
 DEFAULT_DBSPACE オプション 645
 DEFAULT_DISK_STRIPING オプション 647
 DEFAULT_HAVING_SELECTIVITY_PPM オプ
 ション 647
 DEFAULT_ISQL_ENCODING オプション
 説明 648
 DEFAULT_KB_PER_STRIPE オプション 649
 DEFAULT_LIKE_MATCH_SELECTIVITY_PP
 M オプション 650
 DEFAULT_LIKE_RANGE_SELECTIVITY_PPM
 オプション 651

索引

- DEFAULT_PROXY_TABLE_ROW_COUNT オプション 652
- DEFAULT_TABLE_UDF_ROW_COUNT オプション 652
- DELAYED_COMMIT_TIMEOUT オプション 653
- DELAYED_COMMITS オプション 654
- DELETE (位置付け) 文
SQL 構文 312
- DELETE 文
構文 309
- DESCRIBE 文
構文 314
- DISCONNECT 文
構文 318
- DISK_STRIPING オプション 654
- DIVIDE_BY_ZERO_ERROR オプション 655
- DQP
パフォーマンス 42, 203
- DQP_ENABLED オプション 656
- DQP_ENABLED_OVER_NETWORK オプション 657
- DROP AGENT 文
構文 323
- DROP CONNECTION 文
構文 324
- DROP DATABASE 文
構文 325
- DROP DATATYPE 文
構文 319
- DROP DBSPACE 文
構文 319
- DROP DOMAIN 文
構文 319
- DROP EXTERNLOGIN 文
構文 327
- DROP FUNCTION 文
構文 319
- DROP INDEX 文
構文 319
- DROP LDAP SERVER 文 329
- DROP LOGICAL SERVER 文 331
- DROP LOGIN POLICY 文
構文 330
- DROP LS POLICY 文
構文 332
- DROP MULTIPLEX SERVER 文 333
- DROP PROCEDURE 文
構文 319
- DROP ROLE 文 335
- DROP SERVER 文
構文 337
- DROP SERVICE 文
構文 338
- DROP STATEMENT 文
構文 341
- DROP TABLE
IDENTITY_INSERT オプション 319
- DROP TABLE 文
構文 319
- DROP TEXT CONFIGURATION
構文 343
- DROP TEXT INDEX
構文 344
- DROP USER 文 346
- DROP VARIABLE 文
構文 347
- DROP VIEW 文
制限 319
構文 319
- DROP イベント
構文 319
- DROP メッセージ
構文 319
- DROP 文
構文 319
- DYNAMIC SCROLL カーソル 299
- ## E
- EARLY_PREDICATE_EXECUTION オプション 658
- Embedded SQL
DELETE (位置付け) 文構文 312
PUT 文の構文 474
- ENABLE_ASYNC_IO オプション 659
- ENABLE_LOB_VARIABLES オプション 660
- END DECLARE 文
構文 296

END PARALLEL IQ
 CREATE TEXT INDEX 278
 END PARALLEL IQ 文 111
 END キーワード 108
 EXCEPTION 文
 構文 108
 EXECUTE IMMEDIATE 文
 構文 352
 EXECUTE 文
 Transact-SQL 351
 取り消し 503
 付与 391
 構文 348
 EXIT 文
 構文 355
 EXTENDED_JOIN_SYNTAX オプション 661

F

FETCH 文
 構文 357
 FIRST
 1つのローを返す 523
 FLATTEN_SUBQUERIES オプション 779
 FOR JSON 文
 構文 364
 FOR 文
 構文 361
 FORCE_DROP オプション 662
 FORCE_NO_SCROLL_CURSORS オプション
 663
 FORCE_UPDATABLE_CURSORS オプション
 664
 FORWARD TO 文
 構文 370
 FP インデックス
 キャッシュ割り付け 666
 FP_LOOKUP_SIZE オプション 665
 FP_LOOKUP_SIZE_PPM オプション 666
 FP_NBIT_AUTO_LIMIT 667
 FP_NBIT_ENFORCE_LIMITS 668
 FP_NBIT_IQ15_COMPATIBILITY 670
 FP_NBIT_LOOKUP_MB 672
 FP_NBIT_ROLLOVER_MAX_MB 673
 FP_PREDICATE_WORKUNIT_PAGES オプシ
 ョン 675

FPL_EXPRESSION_MEMORY_KB オプション
 675
 FROM 句 372, 523
 contains-expression 372
 SELECT 文 523
 ストアドプロシージャ結果セットからの
 選択 523
 構文 372

G

GARRAY_FILL_FACTOR_PERCENT オプシ
 ョン 676
 GARRAY_PAGE_SPLIT_PAD_PERCENT オ
 プション 678
 GARRAY_PREFETCH_SIZE オプション 677,
 679
 GET DESCRIPTOR 文
 構文 380
 GOTO 文
 Transact-SQL 382
 GRANT CHANGE PASSWORD 文 383
 GRANT ROLE 文 394
 GRANT SET USER 文 399
 GRANT オブジェクトレベル権限 389
 GRANT システム権限文 402
 GRANT 文
 CONNECT 権限 385
 GROUP BY 句
 SELECT 文 523

H

HASH_THRASHING_PERCENT オプション
 680
 HEADER SKIP オプション
 LOAD TABLE 文 430
 HG インデックス
 NULL 値 179
 NULL 値のマルチカラム 179
 クエリパフォーマンスの改善 616
 HG_DELETE_METHOD オプション 681
 HG_SEARCH_RANGE オプション 682
 HTTP_SESSION_TIMEOUT オプション 683

索引

I

I/O

ダイレクト 738, 739

IDENTITY カラム

および DROP TABLE 319

IDENTITY_ENFORCE_UNIQUENESS オプション ン 684

IDENTITY_INSERT オプション 684 テーブルの削除 319

IF 文

Transact-SQL 409

構文 407

IN_SUBQUERY_PREFERENCE オプション 686

INCLUDE 文 構文 410

INDEX_ADVISOR オプション 687

INDEX_ADVISOR_MAX_ROWS オプション 690

INDEX_PREFERENCE オプション 691

INFER_SUBQUERY_PREDICATES オプション 692

INSERT

ワイド 348

構文 412

INSERT 文

WORD SKIP オプション 412

INSTALL JAVA 文

構文 421

INTEGRATED LOGIN

取り消し 504

INTEGRATED LOGIN 文

付与 392

Interactive SQL

OUTPUT 文の構文 463

ファイルの読み書き用のコードページを
指定する 648

Interactive SQL オプション

DEFAULT_ISQL_ENCODING 648

INTO 句

SELECT 文 523

IQ UNIQUE

代替方法 722

IQ UTILITIES 文

構文 424

IQ ストア

テンポラリ領域の予約 798

領域の予約 709

iq_dummy テーブル 372

IQGOVERN_PRIORITY オプション 694

IQGOVERN_PRIORITY_TIME オプション 694

ISOLATION_LEVEL オプション 695

isysserver システムテーブル

コンポーネント統合サービスのリモート
サーバ 241

J

jar ファイル

インストール 421

削除 484

Java

クラスのインストール 421

クラスの削除 484

メソッドシグネチャ 221

Java VM

起動 555

停止 559

Java のテーブル UDF 229

JAVA_LOCATION オプション 696

JAVA_VM_OPTIONS オプション 696

jConnect

サポートの無効化 10

サポートの有効化 10

JOIN_EXPANSION_FACTOR オプション 697

JOIN_OPTIMIZATION オプション 698

JOIN_PREFERENCE オプション 700

JOIN_SIMPLIFICATION_THRESHOLD オプシ ョン 702

K

KERBEROS LOGIN 文

取り消し 505

付与 393

L

LDAP サーバ設定オブジェクト

検証 576

削除 329
 作成 188
 変更 28
 LDAP ログインポリシーオプション 39, 201
 LEAVE 文
 構文 428
 LF_BITMAP_CACHE_KB オプション 703
 LOAD TABLE 文
 HEADER SKIP オプション 430
 ON PARTIAL INPUT ROW オプション 430
 QUOTES オプション 430
 STRIP キーワード 430
 USING キーワード 430
 新しい構文 430
 構文の変更 430
 パフォーマンス 430
 構文 430
 LOAD_ZEROLENGTH_ASNULL オプション
 704
 LOB 変数
 データ型変換 660
 LOCK TABLE
 構文 452
 LOG_CONNECT データベースオプション 705
 LOGIN_MODE オプション 707
 LOGIN_PROCEDURE オプション 708
 LOOP 文
 構文 455

M

MAIN_RESERVED_DBSPACE_MB オプション
 709
 MAX_CARTESIAN_RESULT オプション
 709-712
 MAX_CURSOR_COUNT オプション 713
 MAX_HASH_ROWS オプション 714
 MAX_IQ_GOVERN_PRIORITY オプション
 693
 MAX_IQ_THREADS_PER_CONNECTION オプ
 ション 715
 MAX_IQ_THREADS_PER_TEAM オプション
 715
 MAX_JOIN_ENUMERATION オプション 716
 MAX_PARTITIONED_HASH_MB 717

MAX_PREFIX_PER_CONTAINS_PHRASE オ
 プション 718
 MAX_QUERY_PARALLELISM オプション
 719
 MAX_QUERY_TIME オプション 719
 MAX_STATEMENT_COUNT オプション 720
 MAX_TEMP_SPACE_PER_CONNECTION オプ
 ション 721
 MDSR 暗号化アルゴリズム
 CREATE DATABASE 文 137
 MESSAGE ステートメント
 DEBUG_MESSAGES オプションの設定
 643
 MESSAGE 文
 SQL 構文 457
 MIN_PASSWORD_LENGTH オプション 724
 MIN_ROLE_ADMINS オプション 724
 MINIMIZE_STORAGE オプション 722
 MONITOR_OUTPUT_DIRECTORY オプション
 725
 MPX_AUTOEXCLUDE_TIMEOUT オプション
 726
 MPX_HEARTBEAT_FREQUENCY オプション
 727
 MPX_IDLE_CONNECTION_TIMEOUT オプシ
 ョン 727
 MPX_LIVENESS_TIMEOUT オプション 727
 MPX_MAX_CONNECTION_POOL_SIZE オプ
 ション 728
 MPX_MAX_UNUSED_POOL_SIZE オプション
 729
 MPX_WORK_UNIT_TIMEOUT オプション
 730

N

NEAREST_CENTURY オプション 730
 NO RESULT SET 句 210, 221
 NO SCROLL カーソル 299
 NOEXEC オプション 731
 NON_ANSI_NULL_VARCHAR オプション 732
 NON_KEYWORDS データベースオプション
 733
 NOTIFY_MODULUS オプション 734
 NULL
 マルチカラム HG インデックス 179

索引

NULL 値

マルチカラムの HG インデックス 179

O

ODBC

ODBC_DISTINGUISH_CHAR_AND_VAR
CHAR オプション 734

静的カーソル 299

ODBC_DISTINGUISH_CHAR_AND_VARCHA
R オプション

説明 734

ON EXCEPTION RESUME 句

ストアドプロシージャ 737

ON_CHARSET_CONVERSION_FAILURE オプ
ション

説明 735

ON_ERROR オプション

説明 736

ON_TSQL_ERROR

データベースオプション 737

OPEN 文

構文 461

ORDER BY 句 523

OS_FILE_CACHE_BUFFERING オプション
738

OS_FILE_CACHE_BUFFERING_TEMPDB オプ
ション 739

OUTPUT 文

SQL 構文 463

P

PARAMETERS 文

構文 467

POST_LOGIN_PROCEDURE オプション 741

PRECISION オプション 742

PREFETCH オプション 743

PREFETCH_BUFFER_LIMIT オプション 744

PREFETCH_BUFFER_PERCENT オプション
744

PREFETCH_GARRAY_PERCENT オプション
745

PREFETCH_SORT_PERCENT オプション 745

PREPARE 文

構文 469

PRESERVE_SOURCE_FORMAT オプション
説明 746

PRINT 文

Transact-SQL 構文 472

PURGE 句

FETCH 文 357

PUT 文

SQL 構文 474

Q

QUERY_DETAIL オプション 747

QUERY_NAME オプション 748

QUERY_PLAN オプション 748

QUERY_PLAN_AFTER_RUN オプション 749

QUERY_PLAN_AS_HTML オプション 750

QUERY_PLAN_AS_HTML_DIRECTORY オプ
ション 752

QUERY_PLAN_MIN_TIME オプション 754

QUERY_PLAN_TEXT_ACCESS オプション
754

QUERY_PLAN_TEXT_CACHING オプション
755

QUERY_ROWS_RETURNED_LIMIT オプシ
ョン 756

QUERY_TEMP_SPACE_LIMIT オプション 757

QUERY_TIMING オプション 758

QUIT 文

構文 355

QUOTED_IDENTIFIER オプション 759

R

RAISERROR 文

CONTINUE_AFTER_RAISERROR オプシ
ョン 624

構文 476

READ 文

構文 478

RECOVERY_TIME オプション 760

REFERENCES 句 67

REFRESH TEXT INDEX 文

構文 480

RELEASE SAVEPOINT 文

構文 483

REMOVE 文
 構文 484

RESERVED_KEYWORDS オプション 760

RESIGNAL 文
 構文 486

RESTORE DATABASE 文
 COMPATIBLE 句 487
 VERIFY 句 487
 構文 487
 処理速度の向上 100
 バックアップの検証 487

RESTRICT アクション 257

RESUME 文
 構文 495

RETURN 文
 構文 497

RETURN_DATE_TIME_AS_STRING オプション
 説明 761

REVERT_TO_V15_OPTIMIZER オプション
 762

REVOKE CHANGE PASSWORD 文 499

REVOKE ROLE 文 508

REVOKE SET USER 文 510

REVOKE システム権限文 512

REVOKE データベースオブジェクト権限文
 506

Rigndael 暗号化アルゴリズム
 CREATE DATABASE 文 137

ROLLBACK TO SAVEPOINT 文
 構文 519

ROLLBACK TRANSACTION 文
 Transact-SQL 520
 構文 520

ROLLBACK 文
 構文 518

ROLLUP 演算子 523
 SELECT 文 523

ROUND_TO_EVEN オプション
 説明 763

ROW_COUNT オプション 764

RV_AUTO_MERGE_EVAL_INTERVAL オプション
 ョン 765

rv_max_active_subfragment_count オプション
 766

RV_MERGE_NODE_MEMSIZE オプション
 766

RV_MERGE_TABLE_MEMPERCENT オプション
 767

RV_MERGE_TABLE_NUMROWS オプション
 768

RV_RESERVED_DBSPACE_MB オプション
 768

S

sa_conn_properties
 使用 586

sa_dependent_views システムプロシージャ 100

sa_post_login_procedure 741

SAVE TRANSACTION 文
 Transact-SQL 522
 構文 522

SAVEPOINT 文
 構文 521

SCALE オプション 769

SCROLL カーソル 299

SELECT * 67

SELECT INTO
 テンポラリテーブルに結果を返す 523
 ベーステーブルに結果を返す 523
 ホスト変数に結果を返す 523

SELECT 文内の ALL キーワード 523

SELECT 文内の DISTINCT キーワード 523

select リスト
 DESCRIBE 文 314
 SELECT 文 523

SELECT 文
 FIRST 523
 FROM 句の構文 372
 TOP 523
 構文 523

SET CONNECTION 文
 構文 539

SET DESCRIPTOR 文
 構文 540

SET OPTION 文
 dbisql 構文 600
 構文 542, 545

索引

- 使用 585
- SET SQLCA 文
 - 構文 546
- SET TEMPORARY OPTION 文
 - dbisql 構文 600
 - 構文 542, 545
 - 使用 585
- SET USER
 - 取り消し 510
 - 付与 399
- SET 文
 - Transact-SQL 536
 - 構文 534
- SETUSER 文
 - 同一化 548
- SIGNAL 文
 - 構文 550
- SIGNIFICANTDIGITSFORDOUBLEEQUALITY オプション 771
- SNAPSHOT_VERSIONING オプション 770
- SORT_COLLATION
 - データベースオプション 772
- sp_addmessage 206
- sp_iqcheckoptions システムプロシージャ 586
- sp_login_environment プロシージャ 708
- sp_tsq_environment プロシージャ 708
- SQL
 - 一般的な構文要素 3
 - 構文の表記規則 4
 - 文のインジケータ 5
- SQL 記述子領域
 - カーソルを使用したローの挿入 474
- SQL 標準
 - 準拠 774, 775
- SQL 文
 - ALTER FUNCTION 構文 22
 - DELETE (位置付け) 構文 312
 - MESSAGE 構文 457
 - OUTPUT 構文 463
 - PUT 構文 474
 - UPDATE (位置付け) 構文 571
 - WAITFOR 構文 579
- SQL 変数
 - SET VARIABLE 文 534
 - 作成 289
 - 削除 347
- SQL_FLAGGER_ERROR_LEVEL オプション 774
- SQL_FLAGGER_WARNING_LEVEL オプション 775
- SQLCA
 - INCLUDE 文 410
 - SET SQLCA 文 546
- SQLDA
 - DESCRIBE 文 314
 - Execute 文 348
 - INCLUDE 文 410
 - UPDATE (位置付け) 文 571
 - カーソルを使用したローの挿入 474
 - メモリの割り付け 7
 - 割り付けの解除 295
 - 設定 540
- START DATABASE 文
 - 構文 551
- START ENGINE 文
 - 構文 553
- START JAVA 文
 - 構文 555
- STOP DATABASE 文
 - 構文 556
- STOP ENGINE 文
 - 構文 557
- STOP JAVA 文
 - 構文 559
- STRING_RTRUNCATION オプション 776
- STRIP
 - LOAD TABLE キーワード 430
 - STRIP オプション 430
- SUBQUERY_CACHING_PREFERENCE オプション 777
- SUBQUERY_FLATTENING_PERCENT オプション 778
- SUBQUERY_FLATTENING_PREFERENCE オプション 779
- SUBQUERY_PLACEMENT_PREFERENCE データベースオプション 780
- SUPPRESS_TDS_DEBUGGING オプション 説明 781

- SWEEPER_THREADS_PERCENT オプション
782
- SYSTEM DB 領域 372, 523
- SYSWEBSERVICE システムテーブル
サーバの追加 57
- ## T
- TDS
パスワード暗号化 412
- TDS_EMPTY_STRING_IS_NULL オプション
説明 783
- TEMP_DATA_IN_SHARED_TEMP
論理サーバポリシーのオプション 42
- TEMP_EXTRACT_APPEND オプション 783
- TEMP_EXTRACT_BINARY オプション 784
- TEMP_EXTRACT_COLUMN_DELIMITER オ
プション 785
- TEMP_EXTRACT_DIRECTORY オプション
786
- TEMP_EXTRACT_ESCAPE_QUOTES オプシ
ョン 787
- TEMP_EXTRACT_NAME1 オプション 788
- TEMP_EXTRACT_NAME2 オプション 788
- TEMP_EXTRACT_NAME3 オプション 788
- TEMP_EXTRACT_NAME4 オプション 788
- TEMP_EXTRACT_NAME5 オプション 788
- TEMP_EXTRACT_NAME6 オプション 788
- TEMP_EXTRACT_NAME7 オプション 788
- TEMP_EXTRACT_NAME8 オプション 788
- TEMP_EXTRACT_NAME_n オプション 788
- TEMP_EXTRACT_NULL_AS_EMPTY オプシ
ョン 790
- TEMP_EXTRACT_NULL_AS_ZERO オプシ
ョン 791
- TEMP_EXTRACT_QUOTE オプション 792
- TEMP_EXTRACT_QUOTES オプション 793
- TEMP_EXTRACT_QUOTES_ALL オプション
794
- TEMP_EXTRACT_ROW_DELIMITER オプシ
ョン 795
- TEMP_EXTRACT_SIZE1 オプション 796
- TEMP_EXTRACT_SIZE2 オプション 796
- TEMP_EXTRACT_SIZE3 オプション 796
- TEMP_EXTRACT_SIZE4 オプション 796
- TEMP_EXTRACT_SIZE5 オプション 796
- TEMP_EXTRACT_SIZE6 オプション 796
- TEMP_EXTRACT_SIZE7 オプション 796
- TEMP_EXTRACT_SIZE8 オプション 796
- TEMP_EXTRACT_SIZE_n オプション 796
- TEMP_EXTRACT_SWAP オプション 797
- TEMP_RESERVED_DBSPACE_MB
データベースオプション 798
- TEMP_SPACE_LIMIT_CHECK
データベースオプション 799
- TEXT インデックス
作成 278
削除 344
変更 86
- TEXT_DELETE_METHOD オプション 800
- TIME_FORMAT オプション 801
- TIMESTAMP_FORMAT オプション 802
- TOP
ロー数の指定 523
- TOP_NSORT_CUTOFF_PAGES オプション 803
- Transact-SQL
BEGIN TRANSACTION 文 113
COMMIT TRANSACTION 129
CREATE MESSAGE 206
CREATE PROCEDURE 文 218
CREATE SCHEMA 文 238
ROLLBACK TRANSACTION 文 520
SAVE TRANSACTION 文 522
SET 文 536
Transact-SQL でのエラー処理 476
ストアドプロシージャの実行 351
プロシージャ 218
互換性オプション 598
- TRIGGER EVENT
構文 560
- TRIM_PARTIAL_MBC オプション 804
- TRUNCATE TEXT INDEX 文
構文 563
- TRUNCATE 文
構文 561
- TRUSTED_CERTIFICATES_FILE オプシ
ョン 805
- TSQL_VARIABLES オプション 805

U

UNION 演算 564
 UPDATE (位置付け) 文
 SQL 構文 571
 USAGE 文
 取り消し 517
 付与 406
 USER_RESOURCE_RESERVATION オプション
 806
 USING
 LOAD TABLE キーワード 430
 USING FILE 句
 LOAD TABLE 文 430
 Utilities 文 424

V

VALIDATE LDAP SERVER 文 576
 VARCHAR データ型
 圧縮フォーマットへの変換 632
 VERIFY_PASSWORD_FUNCTION オプション
 807

W

WAIT_FOR_COMMIT オプション 810
 WAITFOR 文
 SQL 構文 579
 WASH_AREA_BUFFERS_PERCENT データベ
 ースオプション 809
 WD インデックス
 CHAR カラム 179
 デリミタ 179
 WD_DELETE_METHOD オプション 811
 WHENEVER 文
 構文 582
 WHERE 句
 SELECT 文 523
 WHILE 文
 Transact-SQL 583
 構文 455
 WITH HOLD 句
 OPEN 文 461
 WORD SKIP オプション 412
 INSERT 文 412

あ

アーカイブデバイス
 並列バックアップの最大数 100
 アーカイブバックアップ
 リストア 493
 暗号化
 TDS パスワード 412
 暗号化アルゴリズム
 CREATE DATABASE 文 137

い

一意性
 制約 257
 イベント
 ディスク領域のモニタリング 157
 トリガ 560
 作成 157
 削除 319
 変更 20
 イベントハンドラ
 トリガ 560
 作成 157
 変更 20
 インデックス 111
 作成 179
 所有者 179
 テーブルでの使用 179
 マルチカラム 179
 マルチカラム HG および NULL 値 179
 命名 179
 ユニーク 179
 ルックアップページ 666
 削除 319

え

エイリアス
 DELETE 文でのエイリアス 309
 SELECT 文 523
 カラム 523
 エージェント
 作成 136, 323
 接続情報の変更 9

- エスケープ文字
 OUTPUT SQL 文 463
- エラー
 RAISERROR 文 476
 SIGNAL 文 550
 Transact-SQL プロシージャ 737
 文字変換 735
- エラー処理
 Transact-SQL プロシージャ 737
- お**
- オートインクリメント
 プライマリーキー値 348
- オプション
 AGGREGATION_PREFERENCE 602
 ALLOW_SNAPSHOT_VERSIONING 604
 ASE_FUNCTION_BEHAVIOR 611
 CIS_ROWSET_SIZE 623
 CONTINUE_AFTER_RAISERROR 624
 CONVERSION_ERROR 625
 CREATE_HG_AND_FORCE_PHYSICAL_
 DELETE 636
 CREATE_HG_WITH_EXACT_DISTINCTS
 635
 dbisql オプションの設定 132
 DBISQL オプションの設定 132
 DEBUG_MESSAGES オプション 643
 DEDICATED_TASK 644
 DEFAULT_ISQL_ENCODING 648
 DQP_ENABLED 656
 DQP_ENABLED_OVER_NETWORK 657
 ENABLE_ASYNC_IO 659
 ENABLE_LOB_VARIABLES 660
 EXTENDED_JOIN_SYNTAX 661
 FLATTEN_SUBQUERIES 779
 FORCE_DROP 662
 FP_LOOKUP_SIZE 665
 FP_LOOKUP_SIZE_PPM 666
 FP_NBIT_AUTO_LIMIT 667
 FP_NBIT_ENFORCE_LIMITS 668
 FP_NBIT_IQ15_COMPATIBILITY 670
 FP_NBIT_LOOKUP_MB 672
 FP_NBIT_ROLLOVER_MAX_MB 673
 MAX_PARTITIONED_HASH_MB 717
 MAX_PREFIX_PER_CONTAINS_PHRASE
 718
 MAX_TEMP_SPACE_PER_CONNECTION
 721
 MPX_AUTOEXCLUDE_TIMEOUT 726
 MPX_HEARTBEAT_FREQUENCY 727
 MPX_IDLE_CONNECTION_TIMEOUT
 727
 MPX_LIVENESS_TIMEOUT 727
 MPX_MAX_CONNECTION_POOL_SIZE
 728
 MPX_MAX_UNUSED_POOL_SIZE 729
 MPX_WORK_UNIT_TIMEOUT 730
 NON_ANSI_NULL_VARCHAR 732
 ODBC_DISTINGUISH_CHAR_AND_VAR
 CHAR 734
 ON_CHARSET_CONVERSION_FAILURE
 735
 ON_ERROR 736
 ON_TSQL_ERROR 737
 POST_LOGIN_PROCEDURE 741
 PRESERVE_SOURCE_FORMAT 746
 RETURN_DATE_TIME_AS_STRING 761
 REVERT_TO_V15_OPTIMIZER 762
 ROUND_TO_EVEN 763
 RV_AUTO_MERGE_EVAL_INTERVAL
 765
 rv_max_active_subfragment_count 766
 RV_MERGE_NODE_MEMSIZE 766
 RV_MERGE_TABLE_NUMROWS 768
 RV_RESERVED_DBSPACE_MBS 768
 SNAPSHOT_VERSIONING 767, 770
 SORT_COLLATION 772
 sp_iqcheckoptions 586
 SUBQUERY_CACHING_PREFERENCE
 777
 SUBQUERY_FLATTENING_PERCENT
 778
 SUBQUERY_FLATTENING_PREFERENC
 E 779
 SUPPRESS_TDS_DEBUGGING 781
 SYSOPTIONDEFAULTS システムテーブル
 586
 TDS_EMPTY_STRING_IS_NULL 783
 temporary の設定 545, 600
 TEXT_DELETE_METHOD 800
 Transact-SQL 536
 カーソル 587
 スコープ 587
 データベースの一般 592
 予期しない動作 372, 523
 リスト 601

索引

- ログインポリシー 40, 202
- 概要 585
- 継続期間 587
- 互換性 598
- 初期設定 591
- 設定 542, 585
- 値の検索 586
- 廃止予定 592
- 優先度 587
- オプション値
 - トランケーション 542, 586
- オフライン
 - DB 領域 13
- オンライン
 - DB 領域 13

か

- カーソル
 - DESCRIBE 314
 - FOR UPDATE 句 299
 - INSENSITIVE 299
 - OPEN 文 461
 - WITH HOLD 句 461
 - カーソルからローを削除 312
 - カーソルを使用したローの挿入 474
 - 感知性 299
 - クローズ 122
 - データベースオプション 587
 - フェッチ 357
 - ループ 361
 - 宣言 299, 305
- 改行
 - WD インデックスのデリミタ 179
- 外部キー
 - 整合性制約 257
 - 名前のない 257
- カタログストア 372, 523
- カタログテンポラリファイル
 - 接続のクォータ超過防止 799
- カラム
 - エイリアス 523
 - 制約 257
 - 命名 3
 - 変更 67

き

- 強力な暗号化
 - CREATE DATABASE 文 137

く

- クエリ
 - LIMIT キーワード 523
 - SELECT 文 523
 - 更新可能なカーソル 299
 - 処理 SQL Anywhere 372, 523
 - パフォーマンスの改善 616
- クライアントファイルのバルクロード
 - エラー 430
 - 文字セット 430
 - ロールバック 430
- クラス
 - インストール 421
 - 削除 484
- グループとして作成 111
- グループ化 111
- グローバルトランザクション
 - サスペンド 727

け

- 結果セット
 - SELECT 523
 - 変数 210, 221, 231, 469
- 権限, 付与
 - ALTER 389
 - DELETE 389
 - INSERT 389
 - LOAD 389
 - REFERENCES 389
 - SELECT 389
 - TRUNCATE 389
 - UPDATE 389
- 権限, 取り消し
 - ALTER 506
 - DELETE 506
 - INSERT 506
 - LOAD 506
 - REFERENCES 506
 - SELECT 506
 - TRUNCATE 506

UPDATE 506

こ

後続ブランク

削除 430

後続ブランクの削除 430

コードページ

DEFAULT_ISQL_ENCODING オプション
648

コマンドファイル

パラメータ 467

コンソール

コンソールへのメッセージ表示 457

さ

サーバ

Web サービスの変更 57

マルチプレックスの変更 46

作成 241

論理サーバの作成 192

論理サーバの削除 331

サービス

追加 244

最適化

MAX_HASH_ROWS オプション 714

MAX_JOIN_ENUMERATION オプション
716

既存のテーブルの定義 163

作成者 3

サスペンドされたトランザクション 727

サブクエリ

分離 523

サブクエリ述部の分離 523

し

シグネチャ

Java メソッド 221

システム権限

取り消し 512

付与 402

リスト 403, 514

システムテーブル

DUMMY 372

PRESERVE_SOURCE_FORMAT 746

SYSFILE 487

ソースカラム 746

システムプロシージャ

sa_dependent_views 100

述部

分離 523

ジョイン

FROM 句の構文 372

SELECT 文 523

ジョイン順の最適化 716

最適化 697, 698, 702

削除 309

ジョインカラム

データ型 379

照合

SORT_COLLATION オプション 772

クライアントファイルのバルクロード
430

所有者 3

シンボリックリンク 137

す

スキーマ

作成 238

スケジュール

WAITFOR 579

スケジュールされたイベント

WAITFOR 文 579

ストアードプロシージャ

sa_dependent_views 100

結果セットに対する SELECT INTO 523

プロキシ 210

作成 210

せ

セーブポイント

RELEASE SAVEPOINT 文 483

ROLLBACK TO SAVEPOINT 文 519

ROLLBACK TRANSACTION 文 520

SAVE TRANSACTION 文 522

名前 3

セキュリティ

パスワードの最小の長さ 724

監査 612

索引

セパレータ

WD インデックス 179

そ

相対パス 137

た

ダイレクト I/O 738, 739

タブ

WD インデックスのデリミタ 179

ダミー IQ テーブル 372

て

ディスク領域

低下時に通知 157

ディスク領域のモニタリング 157

低ディスク領域 157

データ

テーブルからファイルへのデータのエク
スポート 463

データのエクспорт

SELECT 文 523

テーブルからファイルへのデータのエク
スポート 463

データベース

jConnect サポートの無効化 10

jConnect サポートの有効化 10

アップグレード 10

データベースへのデータのロード 430

ファイルの削除 325

起動 551

作成 137

停止 556

変更 10

データベースオプション

DEBUG_MESSAGES オプション 643

DEDICATED_TASK 644

ENABLE_LOB_VARIABLES 660

FLATTEN_SUBQUERIES 779

FORCE_DROP 662

FP_LOOKUP_SIZE_PPM 666

ODBC_DISTINGUISH_CHAR_AND_VAR
CHAR 734

ON_CHARSET_CONVERSION_FAILURE
735

POST_LOGIN_PROCEDURE 741

PRESERVE_SOURCE_FORMAT 746

PUBLIC 589

RETURN_DATE_TIME_AS_STRING 761

ROUND_TO_EVEN 763

SECURITY 589

SUBQUERY_FLATTENING_PERCENT
778

SUBQUERY_FLATTENING_PREFERENC
E 779

SUPPRESS_TDS_DEBUGGING 781

SYSTEM 589

TDS_EMPTY_STRING_IS_NULL 783

カーソル 587

継続期間 587

最大文字列長 542, 586

初期設定 591

データベースサーバ

起動 553

停止 557

データベースのアップグレード 10

データベースのリストア

バックアップの検証 487

データベースの停止 556

データベースファイル

作成 150

変更 13

データ型

ジョインのパフォーマンス 379

ユーザ定義の削除 319

ユーザ定義の変更 18

作成 154

データ型変換

CONVERSION_MODE オプション 626

エラー 625

テーブル

GLOBAL TEMPORARY 257

iq_dummy 372

定義の変更 67

テーブルからファイルへのデータのエク
スポート 463

テンポラリ 217, 257, 306

トランケート 561

プロキシの作成 163

- ロード 430
- ロック 452
- 作成 257
- 削除 319
- 変更 67
- テーブルからのすべてのローの削除 561
- テーブル制約 257
- テーブルのクエリ 372, 523
- テキスト検索
 - FROM contains-expression 372
- テキスト設定オブジェクト
 - 作成 276
 - 削除 343
 - 変更 87
- デバッグ
 - DEBUG_MESSAGES オプション 643
 - MESSAGE 文の動作の制御 457
- デリミタ
 - 例 179
- テンポラリ DB 領域
 - 作成 150
- テンポラリオプション 585
- テンポラリテーブル 217, 257
 - 格納 523
 - 作成 257
 - 宣言 306
- テンポラリファイル (カタログ)
 - TEMP_SPACE_LIMIT_CHECK 799
- テンポラリ領域
 - IQ ストア用に予約済 798

と

- ドメイン 154
 - 変更 18
- トランザクション
 - ROLLBACK TO SAVEPOINT 文 519
 - ROLLBACK TRANSACTION 文 520
 - ROLLBACK 文 518
 - SAVE TRANSACTION 文 522
 - SAVEPOINT 文 521
 - コミット 129
 - サスペンド 727
- トランザクションログ
 - TRUNCATE 文 561
- トランザクション管理 129
 - BEGIN TRANSACTION 文 113
 - Transact-SQL 129

な

- なしのクエリの処理 372, 523
- 名前付きパイプ 430

は

- パーティション
 - 削除 67
 - パーティションの削除 67
 - パーティションの制限 620
- パーミッション
 - CONNECT 権限 385
- バイナリデータ
 - 暗黙の変換の制御 626
- バインド変数
 - DESCRIBE 文 314
 - EXECUTE 文 348
 - OPEN 文 461
- パス
 - 相対 137
- パスワード
 - TDS 暗号化 412
 - 暗号化 412
 - 変更 385
 - 最小の長さ 724
 - 有効期限切れの警告 741
- パスワード暗号化
 - TDS 412
- パスワード変更
 - 取り消し 499
 - 付与 383
- バックアップ
 - 検証 487
 - 速度 100
- バックアップの検証 487
- パッケージ
 - インストール 421
 - 削除 484
- バッファ
 - オペレーティングシステムバッファリングの無効化 738, 739

索引

バッファキャッシュ
パーティショニング 620
パフォーマンス
FROM 句の影響 372
メモリの増加 616
バルクロード 430

ひ

ビュー
インデックス 179
削除 319
変更されたテーブル 67
依存性 99
作成 292
説明 292
変更 96, 99
無効 99
無効なビューの再コンパイル 99

ふ

ファイル
DB 領域 13, 150
オフラインに設定 13
オンラインに設定 13
テーブルからファイルへのデータのエク
スポート 463
複数ローの挿入 348
プット
カーソルヘローをプット 474
ブランク
後続の削除 430
プリフェッチ
BT_PREFETCH_MAX_MISS 616
プロシージャ 469
RAISERROR 文 476
sa_post_login_procedure 741
sp_droplogin 501
sp_iqdroplogin 501
Transact-SQL CREATE PROCEDURE 文
218
結果セット 210, 221
結果セットからの選択 523
プロキシ 210
変数結果セット 210, 221, 231

レプリケート 48
作成 210
削除 319
実行 351
値を返す 497
動的 SQL 文 352
ブロックフェッチ
FETCH 文 357
分散クエリ処理
パフォーマンス 42, 203
文ラベル 3

へ

並列処理
バックアップデバイス 100
変数結果セット
プロシージャから 210, 221, 231, 469

ほ

ホスト変数
インジケータ変数 3
構文 3
宣言 296

ま

マテリアライズドビュー
削除 319
トランケート 561
マテリアライズドビューからの全ローの削除
561
マルチカラムインデックス 179
マルチプレックス
名前の変更 45
名前記憶領域 45
マルチプレックスデータベース
DB 領域の追加 150
作成 137
マルチローフェッチ
FETCH 文 357

み

見出し名 523

め

- メインキャッシュ
 - 削除 319
- メソッドシグネチャ
 - Java 221
- メッセージ
 - 作成 206
 - 削除 319
 - 表示 457
- メモリ
 - プリフェッチ 616

も

- 文字セット
 - クライアントファイルのバルクロード
 - 430
 - 変換エラー 735
- 文字列
 - データベースオプションの長さ 542, 586
- モニタ
 - IQ UTILITIES 文でのモニタ 424
 - 開始と停止 424
 - 出力ファイルの場所の指定 725

ゆ

- ユーザ
 - 削除 346, 501
 - 作成 287
 - 変更 92
- ユーザ ID
 - パスワードの変更 385
- ユーザ定義データ型
 - CREATE DOMAIN 文 154
 - 削除 319
 - 変更 18
- ユーザ定義の関数
 - RETURN 文 497
- ユニークインデックス 179

ら

- ラベル
 - 文 3

文のラベル 382

り

- リストア操作
 - バックアップの検証 487
- リモートサーバ
 - 接続 417
- リモートデータアクセス 24, 54, 566
 - CIS_ROWSET_SIZE 623
- リンク
 - シンボリック 137

る

- ルート論理サーバポリシー 42
- ルックアップページ
 - 最大 666

れ

- レプリケーション
 - プロシージャのレプリケーション 48

ろ

- ロー
 - カーソルからローを削除 312
 - カーソルを使用した挿入 474
- ロー制限句 5
- ローデバイス
 - 命名 137
- ロード
 - スケーラビリティ 620
- ロール
 - 削除 335
 - 作成 235
 - 取り消し 508
 - 付与 394
 - 変更 50
- ローレベルバージョン管理 613
- ログイン
 - パスワード有効期限の警告 741
 - 外部 166
 - 次も参照： 接続

索引

ログインポリシー
 オプション 35, 197
 作成 194
 削除 330
 変更 32, 40, 202

ログインリダイレクション 42

ログイン管理

 POST_LOGIN_PROCEDURE オプション
 741

ログイン管理機能 741

ログイン処理 741

ロック

 ROLLBACK による解放 518

 テーブル 452

わ

ワイド挿入 348