



ユーティリティガイド

---

# SAP Sybase IQ 16.0 SP03

ドキュメント ID：DC00282-01-1603-01

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
<b>CP874toUTF8 データベース管理ユーティリティ</b> .....	<b>3</b>
<b>dbbackup データベース管理ユーティリティ</b> .....	<b>5</b>
<b>dbfhide データベース管理ユーティリティ</b> .....	<b>9</b>
.ini ファイルの内容の隠蔽 (dbfhide) .....	10
<b>dbinfo データベース管理ユーティリティ</b> .....	<b>11</b>
<b>dbisql Interactive SQL ユーティリティ</b> .....	<b>13</b>
Windows の [スタート] メニューから Interactive SQL を起動 .....	13
コマンドプロンプトから Interactive SQL を起動 .....	13
Interactive SQL ユーティリティの構文 .....	14
Interactive SQL コマンドラインオプション .....	14
クワイエットモードの前提条件の設定 .....	17
Interactive SQL の接続パラメータ .....	17
接続パラメータの指定 (接続アシスタント) .....	18
接続パラメータの指定 (手動) .....	18
[接続] ウィンドウのツール .....	21
SQL 文とコマンドファイルの実行 .....	22
テキスト補完の使用 .....	22
コメントの挿入 .....	23
テーブル、カラム、プロシージャの検索 .....	24
SQL 文のインデント .....	25
[SQL 文] ウィンドウ枠のクリア .....	25
すべての SQL 文の実行 .....	26
選択した SQL 文の実行 .....	26
一度に 1 つの SQL 文の実行 .....	26
[文の実行] ツールバーボタンの設定 .....	27
SQL 文のキャンセル .....	27

コマンドファイルの実行 .....	27
COMMIT 文と ROLLBACK 文の実行 .....	28
Interactive SQL インタフェースのカスタマイズ .....	28
[Interactive SQL] ウィンドウリファレンス .....	29
[データ] メニューのウィンドウリファレンス .....	30
[ツール] メニューのウィンドウリファレンス .....	30
Interactive SQL のキーボードショートカット .....	31
結果セット .....	33
複数の結果セットの表示 (UNIX) .....	33
複数の結果セットの表示 (Windows) .....	33
Interactive SQL での結果セットの編集 .....	34
結果セットからの SQL 文の生成 .....	37
SQL 文と結果セットの印刷 .....	38
[お気に入り] リスト .....	38
お気に入りへの .sql ファイルの追加 .....	38
お気に入りへの SQL 文の追加 .....	39
接続をお気に入りに追加 .....	39
お気に入りの表示 .....	39
お気に入りを開く .....	39
お気に入りの編集 .....	40
お気に入りのエクスポート .....	40
お気に入りのインポート .....	40
[お気に入り] メニューのウィンドウリファレンス .....	40
コマンドの呼び出し .....	41
[コマンド履歴] ウィンドウへのアクセス .....	41
コマンドの呼び出し .....	41
[コマンド履歴] ウィンドウからのコマンドのコピー .....	42
[コマンド履歴] ウィンドウからのコマンドの保存 .....	42

[コマンド履歴] ウィンドウからのコマンドの削除 .....	42
コマンドのロギング .....	42
クエリエディタでのクエリの作成 .....	43
クエリの作成 (クエリエディタ) .....	43
クエリエディタの設定 .....	44
クエリエディタの制限事項 .....	45
プランの表示 (Interactive SQL のプランビューア) .....	45
プランビューアの起動 .....	45
Interactive SQL でのグラフィカルなプランの表示 .....	46
グラフィカルなプランの設定 .....	46
プランの印刷 .....	47
ソース制御の統合 .....	47
ソース制御を使用するための Interactive SQL の設定 .....	48
Interactive SQL からのソース制御プロジェクトのオープン .....	49
Interactive SQL からのファイルのチェックアウト .....	49
Interactive SQL からのファイルのチェックイン .....	50
その他のソース制御操作 .....	50
Interactive SQL のための SQL 文 .....	51
CLEAR 文 [Interactive SQL] .....	51
CONFIGURE 文 [Interactive SQL] .....	52
CONNECT 文 [ESQL] [Interactive SQL] .....	53
DISCONNECT 文 [Interactive SQL] .....	56
OUTPUT 文 [Interactive SQL] .....	57
PARAMETERS 文 [Interactive SQL] .....	61
READ 文 [Interactive SQL] .....	62

SET CONNECTION 文 [ESQL] [Interactive SQL]	65
SET OPTION 文 [Interactive SQL]	66
START DATABASE 文 [Interactive SQL]	68
START ENGINE 文 [Interactive SQL]	69
STOP DATABASE 文 [Interactive SQL]	71
STOP ENGINE 文 [Interactive SQL]	72
Interactive SQL 用の SQL オプション	73
DEFAULT_ISQL_ENCODING オプション [Interactive SQL]	73
ON_ERROR オプション [Interactive SQL]	74
<b>dblocate</b> データベース管理ユーティリティ	<b>75</b>
<b>dblog</b> データベース管理ユーティリティ	<b>77</b>
<b>dbping</b> データベース管理ユーティリティ	<b>81</b>
<b>dbstop</b> データベース停止ユーティリティ	<b>85</b>
データベースサーバの停止 (dbstop)	85
dbstop オプション	85
<b>dbtran</b> データベース管理ユーティリティ	<b>87</b>
<b>dbvalid</b> データベース管理ユーティリティ	<b>93</b>
<b>iqdscp</b> 設定ユーティリティ	<b>97</b>
<b>iqdsn</b> データベース管理ユーティリティ	<b>99</b>
Encryption 接続パラメータ [ENC]	102
LogFile 接続パラメータ [LOG]	104
CommBufferSize 接続パラメータ [CBSize]	104
DisableMultiRowFetch 接続パラメータ [DMRF]	106
LivenessTimeout 接続パラメータ [LTO]	106
<b>iqheader</b> データベース管理ユーティリティ	<b>109</b>
<b>iqinit</b> データベース管理ユーティリティ	<b>113</b>
<b>iqocscfg</b> 設定ユーティリティ	<b>125</b>
<b>isql</b> Interactive SQL ユーティリティ	<b>127</b>
<b>iqsqlpp</b> SQL プリプロセッサユーティリティ	<b>139</b>
<b>start_iq</b> データベースサーバ起動ユーティリティ	<b>145</b>

start_iq ユーティリティの起動 .....	145
すべての start_iq スイッチのリスト .....	145
start_iq のサーバオプション .....	146
@data iqsrv16 データベースサーバオプション .....	146
@envvar iqsrv16 サーバオプション .....	147
@filename iqsrv16 サーバオプション .....	148
-c iqsrv16 サーバオプション .....	149
-al iqsrv16 サーバオプション .....	149
-ca iqsrv16 データベースサーバオプション .....	150
-cc iqsrv16 データベースサーバオプション .....	151
-ch iqsrv16 サーバオプション .....	151
-cl iqsrv16 データベースサーバオプション .....	152
-cp iqsrv16 データベースサーバオプション .....	153
-cr iqsrv16 データベースサーバオプション .....	154
-cs iqsrv16 データベースサーバオプション .....	154
-cv iqsrv16 データベースサーバオプション .....	155
-dt iqsrv16 データベースサーバオプション .....	155
-ec iqsrv16 データベースサーバオプション .....	157
-ep iqsrv16 データベースサーバオプション .....	159
-es iqsrv16 データベースサーバオプション .....	160
-fips iqsrv16 データベースサーバオプション .....	161
-ga iqsrv16 データベースサーバオプション .....	162
-gb iqsrv16 データベースサーバオプション .....	162
-gc iqsrv16 データベースサーバオプション .....	163
-gd iqsrv16 データベースサーバオプション .....	163
-ge iqsrv16 データベースサーバオプション .....	165
-gf iqsrv16 データベースサーバオプション .....	165
-gk iqsrv16 データベースサーバオプション .....	165
-gl iqsrv16 サーバオプション .....	166
-gm iqsrv16 データベースサーバオプション .....	166
-gn iqsrv16 サーバオプション .....	167

-gp iqsrv16 データベースサーバオプション .....	168
-gr iqsrv16 サーバオプション .....	168
-gss iqsrv16 サーバオプション .....	168
-gt iqsrv16 データベースサーバオプション .....	169
-gtc iqsrv16 データベースサーバオプション .....	170
-gu iqsrv16 データベースサーバオプション .....	172
-iqfrec iqsrv16 サーバオプション .....	173
-iqgovern iqsrv16 サーバオプション .....	173
-iqlm iqsrv16 サーバオプション .....	174
-iqmc iqsrv16 サーバオプション .....	175
-iqmpx_failover iqsrv16 サーバオプション .....	175
-iqmpx_ov iqsrv16 サーバオプション .....	176
-iqmpx_reclaimwriterfreelist iqsrv16 サーバオプション .....	176
-iqmpx_sn iqsrv16 サーバオプション .....	176
-iqmsgnum iqsrv16 サーバオプション .....	176
-iqmsgsz iqsrv16 サーバオプション .....	177
-iqmt iqsrv16 サーバオプション .....	177
-iqnotemp iqsrv16 サーバオプション .....	178
-iqnumbercpus iqsrv16 サーバオプション .....	178
-iqpartition iqsrv16 サーバオプション .....	179
-iqlvmem start_iq サーバオプション .....	179
-iqro iqsrv16 サーバオプション .....	180
-iqstart iqsrv16 サーバオプション .....	180
-iqtc iqsrv16 サーバオプション .....	181
-iqtss iqsrv16 サーバオプション .....	182
-iqwmem iqsrv16 サーバオプション .....	182
-k iqsrv16 データベースサーバオプション .....	182
-kl iqsrv16 データベースサーバオプション .....	183
-kr iqsrv16 データベースサーバオプション (廃止予定) .....	184
-krb iqsrv16 データベースサーバオプション .....	184



-ks iqsrv16 データベースサーバオプション .....	185
-ksc iqsrv16 データベースサーバオプション ...	186
-ksd iqsrv16 データベースサーバオプション ...	186
-m iqsrv16 データベースサーバオプション .....	186
-n iqsrv16 データベースサーバオプション .....	187
-o iqsrv16 データベースサーバオプション .....	189
-oe iqsrv16 データベースサーバオプション .....	190
-on iqsrv16 データベースサーバオプション .....	190
-os iqsrv16 データベースサーバオプション .....	191
-ot iqsrv16 データベースサーバオプション .....	192
-p iqsrv16 データベースサーバオプション .....	192
-pc iqsrv16 データベースサーバオプション .....	193
-pt iqsrv16 データベースサーバオプション .....	193
-qi iqsrv16 データベースサーバオプション .....	194
-qp iqsrv16 データベースサーバオプション .....	194
-qs iqsrv16 データベースサーバオプション .....	195
-qw iqsrv16 データベースサーバオプション .....	195
-s iqsrv16 データベースサーバオプション .....	196
-sb iqsrv16 データベースサーバオプション .....	196
-sf iqsrv16 データベースサーバオプション .....	197
-sk iqsrv16 データベースサーバオプション .....	205
-su iqsrv16 データベースサーバオプション .....	206
-tdsl iqsrv16 データベースサーバオプション ...	207
-ti iqsrv16 データベースサーバオプション .....	208
-tl iqsrv16 データベースサーバオプション .....	209
-tmf iqsrv16 データベースサーバオプション ...	210
-tq iqsrv16 データベースサーバオプション .....	210
-u iqsrv16 データベースサーバオプション .....	211
-ud iqsrv16 サーバオプション .....	211
-uf iqsrv16 データベースサーバオプション .....	211
-ut iqsrv16 サーバオプション .....	212
-v iqsrv16 サーバオプション .....	212

-x iqsrv16 データベースサーバオプション .....	212
-xd iqsrv16 サーバオプション .....	213
-xs iqsrv16 データベースサーバオプション .....	214
-z iqsrv16 データベースサーバオプション .....	215
-ze iqsrv16 データベースサーバオプション .....	216
-zl iqsrv16 データベースサーバオプション .....	216
-zn iqsrv16 データベースサーバオプション .....	217
-zo iqsrv16 データベースサーバオプション .....	218
-zoc iqsrv16 データベースサーバオプション ...	218
-zr iqsrv16 データベースサーバオプション .....	219
-zs iqsrv16 データベースサーバオプション .....	221
<b>start_iq</b> データベースファイルパラメータ .....	222
<b>start_iq</b> のデータベースオプション .....	222
-a iqsrv16 データベースオプション .....	223
-ad iqsrv16 データベースオプション .....	223
-al iqsrv16 データベースオプション .....	224
-ar iqsrv16 データベースオプション .....	224
-as iqsrv16 データベースオプション .....	225
-dh iqsrv16 データベースオプション .....	226
-ds iqsrv16 データベースオプション .....	226
-ek iqsrv16 データベースオプション .....	227
-iqfreq iqsrv16 データベースオプション .....	227
-m iqsrv16 データベースオプション .....	228
-n iqsrv16 データベースオプション .....	229
-r iqsrv16 データベースオプション .....	229
-sm iqsrv16 データベースオプション .....	230
-sn iqsrv16 データベースオプション .....	230
-xp iqsrv16 データベースオプション .....	230
<b>stop_iq</b> データベース停止ユーティリティ .....	233
サーバを停止または再起動する状況 .....	233
データベースサーバを停止する方法 .....	233
サーバを停止するために必要なパーミッション .....	236

データベースサーバの停止 (stop_iq) .....	236
stop_iq オプション .....	237
cron または at ジョブでのサーバの停止 .....	238
長いパスのサーバ .....	238
<b>付録：dbisqlc Interactive SQL Classic ユーティリティ</b>	
<b>(廃止) .....</b>	<b>241</b>
Interactive SQL Classic の構文 .....	241
Interactive SQL Classic のオプション .....	242
ファンクションキーと特殊キー (UNIX) .....	243
ファンクションキーと特殊キー (Windows) .....	243
コマンド呼び出しキー (Windows) .....	244
<b>索引 .....</b>	<b>245</b>

# 目次

## 対象読者

このマニュアルは、ユーティリティプログラムに関するリファレンス資料を必要としている SAP® Sybase® IQ ユーティリティプログラムユーザを対象としています。

ユーティリティプログラムは、オペレーティングシステムからコマンドを直接呼び出します。リレーショナルデータベースシステムの基礎知識と、SAP Sybase IQ のユーザレベルの基礎的な経験があることを前提にしています。このマニュアルを使用して、使用可能な構文、パラメータ、オプションの情報を確認してください。特定のタスクを実行する方法については、SAP Sybase IQ 16.0 マニュアルセットの他のマニュアルで詳しく説明しています。

対象読者

# CP874toUTF8 データベース管理ユーティリティ

**CP874toUTF8** ユーティリティは、CP874 文字セットのデータを UTF8 照合 (タイ語用に SAP Sybase IQ がサポートしている照合) に変換します。

このユーティリティを使用して、データを UTF8 に変換せずに CP874 文字セットのデータをロードすることもできます。

## 構文

```
CP874toUTF8 [CP874InputFile]
```

## 使用法

このユーティリティは、コマンドプロンプトからのみ実行できます。

**CP874toUTF8** は、指定された CP874 文字セットのファイル (ファイルが指定されない場合は標準入力) を読み込み、UTF8 への変換結果を標準出力に出力します。

---

**注意：** 埋め込み NULL 文字 ('¥0') があるファイルは、正しく変換されません。このような文字は、このユーティリティを実行する前に削除してください。

---

サイズの大きなデータファイルを変換する場合は注意してください。UTF8 出力のサイズは、入力データの最大 3 倍になることがあります。入力と出力のファイルサイズが、どちらもオペレーティングシステムの制限を超えないようにしてください。

**CP874toUTF8** は、変換が正常に完了した場合に終了ステータスとして 0 を返します。0 以外の終了ステータスは、エラーが発生したことを意味します。

**CP874toUTF8** ユーティリティは、すべてのエラーメッセージを stderr に書き込みます。

**CP874toUTF8** は、International Components for Unicode (ICU) ライブラリを呼び出して、データ変換を実行します。ICU がデータを変換できない場合は、**CP874toUTF8** によって変換エラーが報告されます。変換エラーが発生した場合、エラーの場所を隔離するために、ファイルは複数の変換セクションに分割されます。この問題を解決できない場合は、保守契約を結んでいるサポートセンタに問い合わせてください。





# dbbackup データベース管理ユーティリティ

dbbackup ユーティリティは、実行中の SAP Sybase IQ データベースのトランザクションログのコピーを作成します。

**注意：** SAP Sybase IQ データベース全体をバックアップする場合は、**dbbackup**ではなく、必ず **BACKUP DATABASE** を使用してください。**BACKUP DATABASE** では、すべてのデータベースファイルがバックアップされます。これは、カタログストアをバックアップする唯一の方法です。

シングルプレックスまたはマルチプレックスサーバのトランザクションログをトランケートするには、設定ファイルで **-m** オプションを使用するか、または **start iq** コマンドを使用します。サーバは、実行中、停止中のどちらでもかまいません。

## 構文

```
dbbackup [options] target-directory
```

## パラメータ

次の表は、**dbbackup** ユーティリティの利用可能なオプションを示します。

表 1 : dbbackup のオプション

オプション	説明
@data	指定した環境変数または設定ファイルからオプションを読み込む。両方存在し、どちらも同じ名前である場合、SAP Sybase IQ は環境変数を使用する。  設定ファイルに含まれるパスワードまたはその他の情報を保護する場合は、設定ファイルの内容を難読化する File Hiding ユーティリティ (dbfhide) を使用する。
-c "keyword=value; ..."	データベース接続パラメータを指定する。接続パラメータを指定しない場合、SQL CONNECT 環境変数が設定されていると、その接続パラメータが使用される。ユーザ ID は BACKUP DATABASE システム権限を持っていないなければならない。

オプション	説明
-l (lowercase L) <i>file</i>	<p>トランザクションログのライブバックアップをファイルに送る。サーバの障害が発生した場合に、セカンダリシステムをすばやく起動できるようにする。ライブバックアップは、サーバが実行されている間は終了しない。プライマリサーバが使用できなくなるまで実行される。プライマリサーバが使用できなくなった時点でライブバックアップは停止するが、バックアップされたログファイルはそのまま残り、セカンダリシステムをすばやく起動するために使用できる。</p> <p>トランザクションログのバックアップは、必ず実際のトランザクションログと同じ長さか、それより短くなる。ライブバックアップの実行中に別のバックアップがトランザクションログを再開した場合 (dbbackup -x)、ライブバックアップは自動的にライブバックアップログをトランケートし、新しいトランザクションログの最初からライブバックアップを再開する。</p> <p>『管理：バックアップ、リストア、およびデータリカバリ』&gt; 「データデータのバックアップ、リカバリ、アーカイブ」&gt; 「データベースをバックアップする方法」&gt; 「バックアップの種類」&gt; 「トランザクションログのライブバックアップの作成」を参照。</p>
-o	出力メッセージを指定した名前のファイルに書き込む。
-q	クワイエットモード - 出力メッセージは表示されない。このオプションは、このユーティリティをコマンドプロンプトから実行した場合にのみ使用できる。
-r	<p>古いトランザクションログを新しい名前で作成し、新しい空のログを開始する。次の処理が行われる。</p> <ul style="list-style-type: none"> <li>• SAP Sybase IQ は、現在作業中のトランザクションログをコピーし、コマンドで指定されたディレクトリに保存する。</li> <li>• SAP Sybase IQ は現在のトランザクションログを現在のディレクトリに保存するが、フォーマット <code>yyymmddxx.log</code> (<code>xx</code> は <code>AA</code> から <code>ZZ</code> までの連続した英字、<code>yyymmdd</code> は現在の年月日) を使用してファイル名を変更する。このファイルは、現在のトランザクションログではなくなる。</li> <li>• SAP Sybase IQ は、トランザクションを含まない新しいトランザクションログファイルを生成する。新しいファイルのは、これまで現在のトランザクションログとして使用されていたファイルの名前が付けられ、このファイルが現在のトランザクションログとなる。</li> </ul>

オプション	説明
<b>-t</b>	トランザクションログファイルのみバックアップする。トランザクションログをデータベースファイルの最新のバックアップコピーに対して適用できるため、このオプションはインクリメンタルバックアップとして使用できる。
<b>-xo filename</b>	トランザクションログをトランケート (削除して再開) する。
<b>target-directory</b>	バックアップファイルのコピー先のディレクトリ。このディレクトリが存在しない場合は、SAP Sybase IQ によって作成される。ただし、親ディレクトリが存在している必要がある。

### 使用法

**dbbackup** ユーティリティを使用すると、他のアプリケーションやユーザがデータベースを使用している最中でもトランザクションログをバックアップできます。バックアップファイルの名前は、データベースファイルの名前と同じです。

**dbbackup** ユーティリティは、データベース名を70文字にトランケートし、トランケートされた名前でターゲットファイルを作成します。SAP Sybase IQ は、セカンダリサーバを同期するときに **dbbackup** を使用します。dbbackup の制限により、データベース名の長さは70文字未満にする必要があります。

ディスク領域が十分にある場合は、**-r** を使用して既存のログファイルを新しい名前前で保存し、新しい空のログを開始します。ディスク領域が限られている場合は、代わりに **-xo** を使用して、既存のログをトランケートします。

終了コードは、0 (正常) または 0 以外 (エラー) です。

SERVER は、**dbbackup** のデフォルトの論理サーバです。

### 例

次の Windows コマンドは、ユーザ ID DBA、パスワード sql として接続し、sample\_server サーバで実行されている iqdemo データベースのトランザクションログを iqbackup ディレクトリにバックアップします。

```
dbbackup -c "eng=sample_server;dbn=iqdemo;uid=DBA;pwd=sql" c:\¥sample
¥iqbackup
```

### 参照：

- dbfhide データベース管理ユーティリティ (9 ページ)
- -m iqsrv16 データベースオプション (228 ページ)

## **dbbackup** データベース管理ユーティリティ

## dbfhide データベース管理ユーティリティ

**dbfhide** ユーティリティは、設定ファイルと初期化ファイルに単純暗号を追加して、各ファイルの内容を隠蔽するために使用する File Hiding ユーティリティです。

### 構文

```
dbfhide original-configuration-file encrypted-configuration-file
```

### パラメータ

次の表に、**dbfhide** ユーティリティの利用可能なオプションを示します。

オプション	説明
<i>original-configuration-file</i>	元のファイルの名前。
<i>encrypted-configuration-file</i>	難読化された新しいファイルの名前。

### 使用法

一部のユーティリティでは、コマンドラインオプションを保存するために設定ファイルが使用されます。これらのオプションに、パスワードを含めることができます。**dbfhide** ユーティリティを使用して、設定ファイル、および SAP Sybase IQ とそのユーティリティで使用する .ini ファイルに単純暗号を追加することによって、ファイルの内容を難読化できます。元のファイルは変更されません。いったんファイルに単純暗号を追加すると、その暗号を削除することはできません。難読化されたファイルを変更するには、元のファイルのコピーを保存しておき、そのファイルを変更して再び難読化します。

### 例

パーソナルデータベースサーバとサンプルデータベースを起動する設定ファイルを作成します。キャッシュを 10MB に設定し、パーソナルサーバのこのインスタンスの名前を “Elora” にします。次のように設定ファイルを作成します。

```
# Configuration file for server Elora -n Elora -c 10M path¥iqdemo.db
```

# で始まる行は、コメントとして処理されます。

ファイルに `sample.txt` という名前を付けます。この設定ファイルを使用してデータベースを起動するには、次のように入力します。

```
start_iq @sample.txt
```

単純暗号を設定に追加します。

```
dbfhide sample.txt encrypted_sample.txt
```

## dbfhide データベース管理ユーティリティ

encrypted\_sample.txt ファイルを使用して、データベースを起動します。

```
start_iq @encrypted_sample.txt
```

設定ファイルの使用の詳細については、『インストールおよび設定ガイド』を参照してください。

asaldap.ini ファイルに単純暗号を追加する場合は、次のコマンドを使用します。

```
dbfhide asaldap.ini encrypted_asaldap.ini
```

### 参照：

- dbbackup データベース管理ユーティリティ (5 ページ)
- -m iqsrv16 データベースオプション (228 ページ)

## .ini ファイルの内容の隠蔽 (dbfhide)

---

**dbfhide** ユーティリティを使用して、.ini ファイルを難読化します。

1. ファイルを別の名前で作成します。

```
rename saldap.ini saldap.ini.org
```

元のファイルのコピーを保存していない場合、いったん難読化したファイルの内容を変更することはできません。

2. File Hiding ユーティリティを使用してファイルを難読化し、難読化されたファイルに必要なファイル名を付けます。

```
dbfhide saldap.ini.org saldap.ini
```

3. ファイルシステム保護またはオペレーティングシステム保護を使用して saldap.ini.org ファイルを保護するか、ファイルを安全な場所に保存します。

saldap.ini ファイルを変更するには、saldap.ini.org ファイルを編集し、手順 2 を繰り返します。

---

**警告！** UNIX での File Hiding ユーティリティ (dbfhide) を使用した .odbc.ini システム情報ファイルへの単純暗号の追加は、使用データソースが SAP Sybase IQ データソースだけの場合にのみ行ってください。その他のデータベースを使用する予定がある場合、.odbc.ini ファイルの内容を難読化すると、その他のドライバが正しく機能しなくなることがあります。

---

dbfhide は、設定ファイルからオプションを読み込む **@data** パラメータを受け付けません。

## dbinfo データベース管理ユーティリティ

**dbinfo** ユーティリティは、データベースカタログストアに関する情報を表示します。**dbinfo** によって返された情報は、IQ メインストアを反映しません。

有効なユーザ ID ならいずれも **dbinfo** を実行できますが、ページの使用状況に関する統計を取得するには **MANAGE ANY DBSPACE** システム権限が必要です。

### 構文

```
dbinfo [ options ]
```

### パラメータ

次の表に、**dbinfo** ユーティリティの利用可能なオプションを示します。

表 2 : **dbinfo** のオプション

オプション	説明
<b>-c</b> "keyword=value; ..."	接続パラメータを指定する。『管理：データベース』の「接続パラメータと通信パラメータリファレンス」を参照。
<b>-o filename</b>	出力メッセージを、指定した名前のファイルに書き込む。
<b>-q</b>	クワイエットモードで動作する (出力メッセージの表示なし)。
<b>-u</b>	ページの使用状況に関する統計情報を出力する。システムテーブルやユーザ定義のテーブルを含む、すべてのカタログストアテーブルの使用状況およびサイズに関する情報を表示する。他のユーザがデータベースに接続していない場合のみ、使用状況に関する統計情報を要求できる。

### 使用法

**dbinfo** ユーティリティは、いつデータベースが作成されたか、保持されているトランザクションログファイルまたはログミラーファイルの名前、カタログストアのページサイズ、インストールされている Java クラスのバージョンなどの情報を表示します。必要に応じて、カタログストアテーブルの使用状況と詳細を含めることもできます。





# dbisql Interactive SQL ユーティリティ

SAP Sybase IQ には、Interactive SQL (**dbisql**) というグラフィカルユーティリティが付属しています。このユーティリティを使用すると、SQL 文の実行、スクリプトの作成、データベースデータの表示を行うことができます。

---

**注意：** 下位互換性を保つために、SAP Sybase IQ には古い Interactive SQL Classic (**dbisqlc**) ユーティリティも含まれています。Interactive SQL Classic ではなく Interactive SQL を使用してください。Interactive SQL Classic は廃止され、SAP Sybase IQ の今後のリリースでは削除される予定です。

---

## Windows の [スタート] メニューから Interactive SQL を起動

Windows の [スタート] メニューから Interactive SQL を起動します。

1. [スタート]>[プログラム]>[Sybase]>[Sybase IQ 16.0]>[Interactive SQL] を選択します。
2. [接続]ウィンドウで、パラメータを指定します。

### 参照：

- Interactive SQL の接続パラメータ (17 ページ)
- Interactive SQL コマンドラインオプション (14 ページ)

## コマンドプロンプトから Interactive SQL を起動

Windows または UNIX で、コマンドプロンプトから Interactive SQL を起動します。

1. 次のいずれかを実行します。
  - コマンドシェルで、dbisql と入力する。
  - [スタート]>[ファイル名を指定して実行] を選択し、dbisql と入力する。
2. [接続] ウィンドウを開き、パラメータを指定します。

### 参照：

- Interactive SQL の接続パラメータ (17 ページ)
- Interactive SQL コマンドラインオプション (14 ページ)

## Interactive SQL ユーティリティの構文

コマンドプロンプトから Interactive SQL を起動します。

```
dbisql [ options ] [ dbisql-command | command-file ]
```

**注意：** Interactive SQL は、@filename パラメータを受け付けません。終了コードは、0 (正常) または 0 以外 (エラー) です。

- システムプロンプトで次のコマンドを入力すると、ユーザ ID DBA とパスワード sql を使用して、現在のデフォルトサーバに対してコマンドファイル mycom.sql が実行されます。コマンドファイル内でエラーが発生した場合は、処理が終了します。

```
dbisql -c "uid=DBA;pwd=sql" -onerror exit mycom.sql
```

- 次のコマンドは、システムプロンプトで 1 行に入力した場合、現在のデフォルトデータベースに 1 人のユーザを追加します。

```
dbisql -c "uid=DBA;pwd=sql" grant connect to joe identified by passwd
```

**参照：**

- Interactive SQL コマンドラインオプション (14 ページ)

## Interactive SQL コマンドラインオプション

コマンドプロンプトから Interactive SQL を起動するときに、オプションを指定します。

表 3 : Interactive SQL オプション

オプション	説明
-c "keyword=value; ..."	<p>接続パラメータを指定する。このオプションを指定しない場合、環境変数 SQLCONNECT が使用される。Interactive SQL が接続できない場合は、接続パラメータを入力できるダイアログボックスが表示される。</p> <p><b>注意：</b> デフォルトのパラメータではなく、Interactive SQL の接続パラメータを常に指定してください。たとえば、サーバ上で複数のデータベースを起動する場合は、データベース名を指定し、サブネットが設定されているネットワークの場合は、通信プロトコルパラメータとホスト番号を指定します。</p>

オプション	説明
<b>-d</b> <i>delimiter</i>	<p>コマンドデリミタを指定する。デリミタを囲む引用符は省略可能だが、コマンドシェル自体によってデリミタが特別な意味に解釈される場合は省略できない。</p> <p>コマンドデリミタは、データベースに保存された設定 (ユーザのための設定、つまり PUBLIC 設定) にかかわらず、Interactive SQL セッションのすべての接続に使用される。</p>
<b>-d1</b>	<p>(最後の文字は英字 L の小文字ではなく数字の 1)。Interactive SQL が実行するすべての文をコマンドウィンドウ (STDOUT) にエコーする。このフィードバックは、SQL スクリプトをデバッグする場合や、Interactive SQL で大きな SQL スクリプトを処理する場合に便利である。</p>
<b>-datasource</b> <i>dsn-name</i>	<p>接続先の ODBC データソースを指定する。このオプションは、SQL Anywhere® 16 JDBC ドライバを使用していなくても使用できる。ただし、接続先のデータソースが TCP/IP を使用するよう設定されていない場合は、SQL Anywhere JDBC ドライバを使用して接続する必要がある。SAP Sybase IQ データソースは、デフォルトで TCP/IP を使用するよう設定される。</p>
<b>-f</b> <i>file name</i>	<p><i>file name</i> というファイルを開く (実行はしない)。ファイル名にスペースが含まれる場合は、引用符で囲む必要がある。それ以外の場合、引用符はオプション。ファイルが存在しない場合、またはファイルではなくディレクトリだった場合、Interactive SQL はエラーメッセージをコンソールに出力し、終了する。ファイル名に完全なドライブとパスの指定が含まれない場合、現在のディレクトリからの相対パスであると見なされる。</p>
<b>-host</b> <i>host name</i>	<p>データベースサーバを実行しているコンピュータのホスト名または IP アドレスを指定する。現在のマシンを表すには <b>localhost</b> という名前を使用できる。</p>

オプション	説明
<b>-nogui</b>	<p>Interactive SQL をコマンドプロンプトモードで実行する。ウィンドウを使用するユーザインタフェースは表示されない。バッチオペレーションに便利なオプション。dbisql-command または command-file のいずれかを指定すると、-nogui が想定される。</p> <p>-nogui モードでは、Interactive SQL はプログラム終了コードを設定して、成功または失敗を示す。Windows オペレーティングシステムの場合、環境変数 ERRORLEVEL にプログラム終了コードが設定される。次の終了コードが返される。</p> <ul style="list-style-type: none"> <li>• 0 – 成功。</li> <li>• 1 – 一般的な失敗。いずれかの段階で、SQL 文または Interactive SQL 文が正常に実行されず、ユーザが SQL 文の実行を停止した。または、Interactive SQL で内部エラーが検出された。</li> <li>• 5 – ユーザが Interactive SQL を終了した。実行時にエラーが発生した場合、エラーを無視するか、Interactive SQL を停止または終了するかを問い合わせるメッセージが表示される。終了を選択した場合は、終了コードとして 5 が返される。エラーが発生し、Interactive SQL オプション ON_ERROR が EXIT に設定されている場合もコード 5 が返される。</li> <li>• 9 – 接続できない。</li> <li>• 255 – 不正なコマンド。コマンドラインに不完全または不正なスイッチが含まれている。</li> </ul> <p><b>注意：</b> -nogui モードでは、-d スイッチを使用してコマンドデリミタを指定した場合でも、[Enter] キーを押すと、コマンドプロンプトで入力したすべての SQL テキストが実行されます。[Enter] キーは、SQL コマンドが完了していることを確認してから押してください。</p>
<b>-onerror</b> (continue   exit)	<p>コマンドファイルから文を読み込んでいる最中にエラーが起こった場合の処理を制御する。このオプションは、on_error 設定に優先する。このオプションは、Interactive SQL をバッチオペレーションに使用する場合に便利。</p>
<b>-port</b> portnumber	<p>データベースサーバが実行されているポート番号を指定する。SAP Sybase IQ のデフォルトのポート番号は 2638。</p>
<b>-q</b>	<p>クワイエットモードで実行する。出力メッセージは表示されない。このオプションは、コマンドまたはコマンドファイルを使用して Interactive SQL を起動したときにのみ役立つ。このオプションを指定しても、エラーメッセージの出力は抑止されない。</p>
<b>-version</b>	<p>Interactive SQL のバージョン番号を表示する。</p>

オプション	説明
-x	コマンドをスキャンするが、実行はしない。このオプションは、長いコマンドファイルの構文エラーをチェックする場合に有用。

**参照：**

- Interactive SQL の接続パラメータ (17 ページ)
- コマンドプロンプトから Interactive SQL を起動 (13 ページ)
- Windows の [スタート] メニューから Interactive SQL を起動 (13 ページ)
- Interactive SQL ユーティリティの構文 (14 ページ)

**クワイエットモードの前提条件の設定**

-q オプション(クワイエットモード)を指定して Interactive SQL を使用し、かつデータ抽出コマンド(主としてオプション TEMP\_EXTRACT\_NAME1 を出力ファイルに設定)がコマンドファイルに含まれている場合は、最初に **[すべての結果セットを表示]** オプションを永続的にオンに設定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場合、データ抽出の出力ファイルは作成されません。

1. Interactive SQL を起動します。
2. [Interactive SQL] ウィンドウで、[ツール]>[オプション] を選択します。[オプション] ウィンドウが表示されます。
3. 左ウィンドウ枠で [SAP Sybase IQ] を選択します。
4. **[結果]** タブをクリックします。
5. [処理中の結果] 領域で、**[すべての結果セットを表示]** を選択します。
6. **[OK]** をクリックします。

**Interactive SQL の接続パラメータ**

データベースが接続されていない場合、SAP Sybase IQ は接続パラメータを要求するために **[接続]** ウィンドウを表示します。**[接続]** ウィンドウに入力した情報はセッション間で維持されません。

**[接続]** ウィンドウで指定する接続パラメータは、データベースサーバで実行されているデータベースの数によって異なります。単一のデータベースだけが実行されており、サーバがローカルマシンで *IQTMP16* 環境変数を使用して起動されている場合、データベースに接続するために必要な入力は、**[ユーザID]** フィールドと**[パスワード]** フィールドのみです。データベースサーバで複数のデータベースが実行

されている場合は、サーバ名やデータベース名などの追加のパラメータを指定する必要があります。

次のいずれかの方法を使用して、[接続] ウィンドウで接続パラメータを入力します。

- 接続アシスタントの使用
- 手動

---

**注意：** `-c` オプションを使用してコマンドラインで接続パラメータを指定することによって、[接続] ウィンドウをスキップできます。

---

[接続] ウィンドウが表示された場合や、情報不足に関するエラーメッセージが表示された場合は、`-host` と `-port` を入力するか、またはその他の不足情報を[詳細] タブで入力します。目的のデータベースがリモートサーバにある場合は、次のように `-host` パラメータと `-port` パラメータをそれぞれ別々の行に入力します。

```
-host fiona  
-port 1870
```

**参照：**

- Interactive SQL コマンドラインオプション (14 ページ)
- コマンドプロンプトから Interactive SQL を起動 (13 ページ)
- Windows の [スタート] メニューから Interactive SQL を起動 (13 ページ)

### 接続パラメータの指定 (接続アシスタント)

[接続] ウィンドウには、データベースへの接続に役立つ接続アシスタントウィザードがあります。接続アシスタントの表示と非表示を切り替えるには、ウィンドウの右上隅にある矢印をクリックします。

1. [接続] ウィンドウの [接続アシスタント] 領域で [次へ] ボタンをクリックします。
2. 画面の指示に従います。

---

**注意：** 接続に ODBC ソースを使用している場合は、接続アシスタントは使用しないでください。接続アシスタントを閉じ、[ID] タブをクリックします。

---

### 接続パラメータの指定 (手動)

[接続] ウィンドウの [ID] タブ、[データベース] タブ、[ネットワーク] タブ、[詳細オプション] タブを使用してデータベースに接続します。

1. [認証] リストで、[データベース] または [Windows 統合ログイン] 認証を指定します。
2. [アクション] を選択します。

表 4 : ODBC データソースを使用した接続

フィールド／ボタン／オプション	詳細
ODBC データソース名	データベース接続用のデータソース (保管された接続パラメータのセット) を選択します。このフィールドは DSN 接続パラメータに相当し、レジストリ内のデータソースを参照している。
ODBC データソースファイル	接続用のデータソースファイルを選択します。

表 5 : このコンピュータで稼働しているデータベースに接続

フィールド／ボタン／オプション	詳細
サーバ名	接続先のデータベースサーバの名前です。リモートサーバの場合は、host name:port number という形式でサーバを指定する。
データベース名	接続先のデータベースの名前です。データベース名が必要になるのは、データベースサーバで複数のデータベースを実行している場合のみ。データベースがサーバ上でまだ実行されていない場合、代わりにデータベースファイルを指定してください。最近使用したデータベースを選択することも、[参照] をクリックしてデータベースファイルを検索することもできる。  <b>注意：</b> 既に実行中のデータベースに接続しようとして、データベース名とデータベースファイルの両方を指定した場合、データベースファイルは無視されます。

表 6 : 別のコンピュータで稼働しているデータベースに接続

フィールド／ボタン／オプション	詳細
ホスト	データベースサーバが実行されているコンピュータの名前。
ポート	データベースサーバが使用しているポートです。
サーバ名	接続先のデータベースサーバの名前です。リモートサーバの場合は、host name:port number という形式でサーバを指定する。
検索	サーバを検索する。

表 7: このコンピュータのデータベースを起動して接続

フィールド/ ボタン/オプション	詳細
データベース ファイル	データベースファイルのフルパスと名前を入力します。このように指定しない場合、ファイルのパスは、データベースサーバの作業ディレクトリの相対パスになる。
暗号化キー	データベースファイルが暗号化されている場合、データベースサーバがデータベースを起動するたびにデータベースサーバの暗号化キーを指定してください。
データベース 名	接続先のデータベースの名前です。データベース名が必要になるのは、データベースサーバで複数のデータベースを実行している場合のみ。データベースがサーバ上でまだ実行されていない場合、代わりにデータベースファイルを指定してください。最近使用したデータベースを選択することも、[参照] をクリックしてデータベースファイルを検索することもできる。  <b>注意:</b> 既に実行中のデータベースに接続しようとして、データベース名とデータベースファイルの両方を指定した場合、データベースファイルは無視されます。
サーバ名	接続先のデータベースサーバの名前です。リモートサーバの場合は、host name:port number という形式でサーバを指定する。
開始行	ローカルデータベースサーバに接続し、独自の起動パラメータを設定する場合にのみ、開始行を入力する。データベースサーバのフルパスを入力してください。
最終切断後に データベース を停止	最後のユーザが切断すると、データベースを自動的に停止します。

表 8: 別のコンピュータのデータベースを起動して接続

フィールド/ ボタン/オプション	詳細
データベース ファイル	データベースファイルのフルパスと名前を入力することが推奨される。このように指定しない場合、ファイルのパスは、データベースサーバの作業ディレクトリの相対パスになる。
暗号化キー	データベースファイルが暗号化されている場合、データベースサーバがデータベースを起動するたびにデータベースサーバの暗号化キーを指定する。



フィールド/ ボタン/ オプション	詳細
データベース名	<p>接続先のデータベースの名前。データベース名が必要になるのは、データベースサーバで複数のデータベースを実行している場合のみ。データベースがサーバ上でまだ実行されていない場合、代わりにデータベースファイルを指定する。最近使用したデータベースを選択することも、[参照]をクリックしてデータベースファイルを検索することもできる。</p> <p><b>注意：</b>既に実行中のデータベースに接続しようとして、データベース名とデータベースファイルの両方を指定した場合、データベースファイルは無視される。</p>
サーバ名	接続先のデータベースサーバの名前。リモートサーバの場合は、 <code>host name:port number</code> という形式でサーバを指定する。
ホスト	データベースサーバが実行されているコンピュータの名前。
ポート	データベースサーバが使用しているポート。
最終切断後に データベースを 停止	最後のユーザが切断すると、データベースを自動的に停止する。

表 9 : 接続文字列を使用して接続

フィールド/ ボタン/ オプション	詳細
パラメータ	接続文字列。

- [接続] をクリックします。

### [接続] ウィンドウのツール

[接続] ウィンドウの下部にある [ツール] をクリックすると、次のオプションにアクセスできます。

表 10 : [接続] ウィンドウのツール

ツール	説明
接続テスト	入力された情報で正しく接続されるかどうかをテストする。
接続文字列をクリップ ボードにコピー	[接続] ウィンドウで指定したオプションから接続文字列を作成し、その文字列をクリップボードにコピーする。
ODBC データソースとして 保存	新しいデータソースを、自身のデータソースのリストに保存する。

## SQL 文とコマンドファイルの実行

**SELECT** 文を実行すると、結果セットが [結果] ウィンドウ枠の [結果] タブに表示されます。デフォルトでは、ロー番号が結果セットの左側に表示されます。

### テキスト補完の使用

テキスト補完は、入力中に完全なキーワードとオブジェクト名を表示できるオプションです。

たとえば、アルファベットの s を入力すると、テキスト補完ウィンドウが表示され、S で始まる入力候補のキーワードやオブジェクト名のリストが示されます。テキスト補完設定は、[オプション] ウィンドウから変更できます。

1. [SQL 文] ウィンドウ枠で、データベースオブジェクト名の最初の文字を入力します。
2. デフォルトでは、入力するとテキスト補完ウィンドウが自動的に表示されます。[Ctrl] キーを押しながら [Space] キーを押すか、[Ctrl] キー、[Shift] キー、[Space] キーを同時に押してウィンドウを開くこともできます。
3. 目的のオブジェクト名が見つからない場合は、[Ctrl] キーを押しながら [A] キーを押して、全データベースオブジェクトのリストを表示します (このリストは、ユーザが設定したフィルタリング オプションに基づいて表示されます。デフォルトでは、データベースオブジェクトがすべて表示されます)。
4. リストからオブジェクト名を選択し、[Enter] キーを押します。

### テキスト補完キーボードショートカット

キーボードショートカットはテキスト補完リストが開いている場合に使用できません。

キー	説明
Ctrl + A	コンテキストには関係なく、一致したリストを表示する。
Ctrl + 二重引用符 (")(または Ctrl + Shift + ')	quoted_identifier オプションの設定に関係なく、引用符で囲んで名前を補完する。
Ctrl + アスタリスク (*) (または Ctrl + Shift + 8)	テーブルの場合、カラムのカンマ区切りのリスト (データ型を含む) を挿入する。ストアードプロシージャの場合、プロシージャ名を挿入し、続けてパラメータ名とデータ型のカンマ区切りのリストを挿入する。

キー	説明
Ctrl + C	テキスト補完リストの内容を変更して、カラムを表示または非表示にする。
Ctrl + F	テキスト補完リストの内容を変更して、SQL 関数を表示または非表示にする。
Ctrl + P	テキスト補完リストの内容を変更して、ストアプロシージャと関数を表示または非表示にする。
Ctrl + プラス記号 (+)(または Ctrl + Shift +=)	テーブルの場合、カラムのカンマ区切りのリストを挿入する。ストアプロシージャの場合、プロシージャ名を挿入し、続けてパラメータ名のカンマ区切りのリストを挿入する。
Ctrl + S	テキスト補完リストの内容を変更して、システムオブジェクトを表示または非表示にする。
Ctrl + Shift + Space または Ctrl + Space	テキスト補完ウィンドウを開く。[Ctrl] キーを押しながら [Space] キーを押してテキスト補完ウィンドウを開くこともできる。
Ctrl + T	テキスト補完リストの内容を変更して、テーブルを表示または非表示にする。
Ctrl + V	テキスト補完リストの内容を変更して、ビューを表示または非表示にする。
Esc	テキストを追加しないでテキスト補完ウィンドウを閉じる。
Tab	選択内容を受け入れてテキスト補完ウィンドウを閉じる。

**参照：**

- Interactive SQL のキーボードショートカット (31 ページ)

**コメントの挿入**

コメントは、SQL 文または文ブロックに説明テキストを付加するために使用しません。

テキストをコメントにできます。データベースサーバは、コメントを実行しません。Interactive SQL では、以下のタイプのコメントがサポートされます。

- -- (二重ハイフン)
- //(二重スラッシュ)
- /\* ... \*/ (スラッシュ - アスタリスク)

1. [SQL 文] ウィンドウ枠内でテキストを選択します。
2. [Ctrl] キーを押しながら[マイナス記号 (-)] キーを押して二重ハイフンのコメントインジケータを追加するか、[Ctrl] キーを押しながら[スラッシュ (/)] キーを押して二重スラッシュのコメントインジケータを追加します。  
テキストが選択されていないと、コメントインジケータは現在の行の先頭に追加されます。

### テーブル、カラム、プロシージャの検索

Interactive SQL でコマンドを入力するときに、現在のデータベースにあるテーブル、カラム、またはプロシージャの名前を検索し、それをカーソル位置に挿入できます。

#### データベース内のテーブル名の検索

データベースに格納されているテーブルの名前を検索し、それをカーソル位置に挿入します。

1. [ツール]>[テーブル名のルックアップ] を選択するか、[F7] キーを押します。
2. テーブルを探し、選択します。
3. [OK] をクリックして、テーブル名を [SQL 文] ウィンドウ枠の現在のカーソル位置に挿入します。

#### データベース内のカラム名の検索

データベースに格納されているカラムの名前を検索し、それをカーソル位置に挿入します。

1. [ツール]>[テーブル名のルックアップ] を選択するか、[F7] キーを押します。
2. カラムを含むテーブルを探し、選択します。

---

**注意：**[テーブル名のルックアップ] ウィンドウでは、検索対象のテーブルの名前の最初の数文字を入力できます。これにより、リストが絞り込まれ、ダイアログには入力したテキストで始まる項目のみが含まれます。

---

3. [カラムを表示] をクリックします。
4. カラムを選択し、[OK] をクリックして、カラム名を [SQL 文] ウィンドウ枠の現在のカーソル位置に挿入します。

#### データベース内のプロシージャ名の検索

データベースに格納されているプロシージャの名前を検索し、それをカーソル位置に挿入します。

1. [ツール]>[プロシージャ名のルックアップ] を選択するか、[F8] キーを押します。

2. プロシージャを探し、選択します。

**注意：**[プロシージャ名のルックアップ] ウィンドウでは、検索対象のプロシージャの名前の最初の数文字を入力できます。これにより、リストが絞り込まれ、ダイアログには入力したテキストで始まる項目のみが含まれます。

3. [OK] をクリックして、プロシージャ名を [SQL 文] ウィンドウ枠の現在のカーソル位置に挿入します。

## SQL 文のインデント

SQL 文のインデントを拡張または縮小したり、インデントされるスペースのデフォルト数を変更したりします。

### SQL 文のインデントの追加または拡張

SQL 文にインデントを追加するか、すでにインデント済みの SQL 文のインデントを拡張します。

1. [SQL 文] ウィンドウ枠で、インデントするテキストを選択します。テキストを選択しないと、インデントは現在の行に適用されます。
2. [Ctrl] キーと [Shift] キーを押しながら、[ピリオド (.)] キーを押します。

### SQL 文のインデントの削除または縮小

SQL 文からインデントを削除するか、すでにインデント済みの SQL 文のインデントを縮小します。

1. [SQL 文] ウィンドウ枠で、インデントを縮小するテキストを選択します。テキストを選択しないと、インデントは現在の行に適用されます。
2. [Ctrl] キーと [Shift] キーを押しながら、[カンマ (,)] キーを押します。

### インデントのスペース数の変更

デフォルトのインデントサイズを変更します。

1. [ツール] > [オプション] を選択します。
2. [エディタ] を選択し、[タブ] タブをクリックします。
3. [インデントサイズ] フィールドに新しい数を入力します。

## [SQL 文] ウィンドウ枠のクリア

[SQL 文] ウィンドウ枠に表示されている SQL コードが不要になった場合には、[SQL 文] ウィンドウ枠をクリアします。

デフォルトでは、[Esc] キーを押すと、[SQL 文] ウィンドウ枠の内容が変更されずに保持されます。[Esc] キーによって [SQL 文] ウィンドウ枠がクリアされるように設定し、開いている結果セットをすべて閉じます。

1. [ツール]>[オプション]>[互換性] を選択します。
2. [Esc キーを押すと、SQL 文がクリアされ、結果セットが閉じます。] を選択します。
3. [Esc] キーを押します。

### すべての SQL 文の実行

複数の SQL 文が [SQL 文] ウィンドウ枠に存在する場合、すべての文を同時に実行できます。

1. [SQL 文] ウィンドウ枠でクエリを入力します。
2. [F5] キーを押すか、[SQL の実行] を選択して、文を実行します。

#### 参照：

- 選択した SQL 文の実行 (26 ページ)
- 一度に 1 つの SQL 文の実行 (26 ページ)
- [文の実行] ツールバーボタンの設定 (27 ページ)

### 選択した SQL 文の実行

複数の SQL 文が [SQL 文] ウィンドウ枠に存在する場合、1 つまたは複数の文を選択して実行できます。

1. [SQL 文] ウィンドウ枠でクエリを入力し、クエリを選択します。
2. [F9] キーを押すか、[選択した SQL の実行] を選択して、文を実行します。

#### 参照：

- すべての SQL 文の実行 (26 ページ)
- 一度に 1 つの SQL 文の実行 (26 ページ)
- [文の実行] ツールバーボタンの設定 (27 ページ)

### 一度に 1 つの SQL 文の実行

SQL 文を個別に実行するには、[シングルステップ] オプションを使用します。この方法は、デバッグ時に便利です。

1. [SQL 文] ウィンドウ枠でクエリを入力します。
2. 実行する文にカーソルを置きます。
3. [SQL]>[シングルステップ] を選択するか、[Shift] キーを押しながら、[F9] キーを押します。  
選択した SQL 文が実行され、次の SQL 文が選択されます。

4. [Shift] キーを押しながら、[F9]キーを押します。  
選択した SQL 文が実行されます。
5. 選択して実行する文がなくなるまで、上記の手順を繰り返します。

**参照：**

- すべての SQL 文の実行 (26 ページ)
- 選択した SQL 文の実行 (26 ページ)
- [文の実行] ツールバーボタンの設定 (27 ページ)

## [文の実行] ツールバーボタンの設定

[文の実行] ボタンを設定して、すべての SQL 文の実行、または選択した文のみの実行のいずれかを行うようにします。

1. [ツール]>[オプション]>[ツールバー] を選択します。
2. 次のいずれかを実行します。
  - [すべての文の実行] を選択します。デフォルトの設定です。
  - [選択の実行] を選択します。

**参照：**

- すべての SQL 文の実行 (26 ページ)
- 選択した SQL 文の実行 (26 ページ)
- 一度に 1 つの SQL 文の実行 (26 ページ)

## SQL 文のキャンセル

キャンセル操作を実行すると、現在の処理が停止し、次のコマンドを求めるプロンプトが表示されます。Interactive SQL ツールバーの [SQL 文の中断] ボタンは、コマンドをキャンセルする機能を提供します。

Interactive SQL ツールバーの [SQL 文の中断] をクリックします。

**参照：**

- ON\_ERROR オプション [Interactive SQL] (74 ページ)

## コマンドファイルの実行

コマンドファイルは SQL 文を含むテキストファイルで、同じ SQL 文を繰り返し実行する場合に便利です。Interactive SQL を使用して、コマンドファイルのオープン、表示、実行、保存を行うことができます。

次のいずれかの方法を使用して、コマンドファイルを実行します。

- Interactive SQL の **READ** 文を使用して、コマンドファイルを実行します。  
次に例を示します。

```
READ temp.sql;
```
- コマンドファイルを [SQL 文] ウィンドウ枠にロードし、そこから直接実行します。
- [ファイル]>[開く] を選択して、コマンドファイルを [SQL 文] ウィンドウ枠にロードします。入力が求められたら、ファイル名を入力します。
- [ファイル]>[スクリプトの実行] を選択して、コマンドファイルをロードせずに実行します。
- コマンドファイルをコマンドライン引数として Interactive SQL に指定します。

### COMMIT 文と ROLLBACK 文の実行

**COMMIT** 文と **ROLLBACK** 文には複数の実行方法があります。

**COMMIT** 文は次の方法で実行できます。

- [SQL]>[コミット] を選択する。
- [Ctrl] キー、[Shift] キー、[C] キーを同時に押す。
- [SQL 文] ウィンドウ枠で Commit と入力して **COMMIT** 文を実行する。

**ROLLBACK** 文は次の方法で実行できます。

- [SQL]>[ロールバック] を選択する。
- [Ctrl] キー、[Shift] キー、[R] キーを同時に押す。
- [SQL 文] ウィンドウ枠で Rollback と入力して **ROLLBACK** 文を実行する。

---

**注意：** **COMMIT** や **ROLLBACK** を [SQL] メニューやキーボードショートカットを使用して実行しても、[SQL 文] ウィンドウ枠の内容は変更されません。ただし、[結果] ウィンドウ枠内の [結果] タブの内容はクリアされます。

---

### Interactive SQL インタフェースのカスタマイズ

Interactive SQL のタブやウィンドウ枠は、[オプション] ウィンドウを使用してカスタマイズできます。

1. Interactive SQL で、[ツール]>[オプション] を選択します。
2. 左側のウィンドウ枠でオプションをクリックし、希望するオプションを指定します。結果が表示される形式の変更、各文の後にコミットを実行するかどうか（または、終了または切断）の指定、更新を確認するかどうかの指定を行うことができます。
3. [OK] をクリックします。



**[Interactive SQL] ウィンドウリファレンス**

[Interactive SQL] ウィンドウを使用して、SQL 文を入力したり、結果やメッセージを表示したりできます。

表 11 : [Interactive SQL] ウィンドウの詳細

ウィンドウ枠	カラム / タブ	説明
SQL 文		データにアクセスしたり、データを変更したりするための SQL 文を入力する場所を提供する。
	[行番号] カラム	<p>左側のカラムはライン番号を示す。これらの行番号を使用して、次を行うことができる。</p> <ul style="list-style-type: none"> <li>行を選択するには、行番号をクリックする。または、カーソルを行に置いて、[Ctrl + カンマ (,)] キーを押す。</li> <li>クリックとドラッグを行って、複数行を選択する。</li> <li>行に対応する SQL 文全体を選択するには、行をダブルクリックする。または、その文にカーソルを置いて、[Ctrl + ピリオド (.)] キーを押す。</li> </ul>
結果		[結果] ウィンドウ枠には、[結果]と[メッセージ]の2つのタブがある。これらのタブは、[結果] ウィンドウ枠の下部に表示される。
	[結果] タブ	[結果] タブには、実行したコマンドの結果が表示される。たとえば、データベースから特定のデータを検索する SQL 文を使用すると、[結果] タブに、上のウィンドウ枠の探索条件と一致するカラムとローが表示される。[結果] タブに表示された結果セットは編集できる。
	[メッセージ] タブ	[メッセージ] タブには、Interactive SQL で実行する SQL 文に関するデータベースサーバのメッセージが表示される。

SAP Sybase IQ データベースのグラフィカルなプランの結果は、別の [プランビューア] ウィンドウに表示されます。

**参照：**

- プランの表示 (Interactive SQL のプランビューア) (45 ページ)

**[データ] メニューのウィンドウリファレンス**

[データ]メニューから利用できるウィンドウを使用して、結果セットをエクスポートしたり、データをインポートしたりできます。

表 12: [データ] メニューのウィンドウ

ウィンドウ	説明
エクスポート	結果セットをエクスポートする機能を提供するエクスポートウィザードを開く。
インポート	ファイルまたはデータベースからデータをインポートする機能を提供するインポートウィザードを開く。

**[ツール] メニューのウィンドウリファレンス**

[ツール]メニューから利用できるウィンドウを使用して、Interactive SQL の設定、クエリに挿入するテーブルやプロシージャの名前の検索、クエリの編集を行うことができます。

表 13: [ツール] メニューのウィンドウ

ウィンドウ	説明
テーブル名のルックアップ	[テーブル名のルックアップ]ウィンドウを使用すると、テーブル名とカラム名を参照し、それらを [SQL 文] ウィンドウ枠に挿入できる。
プロシージャ名のルックアップ	[プロシージャ名のルックアップ]ウィンドウを使用すると、プロシージャ名を参照し、それらを [SQL 文] ウィンドウ枠に挿入できる。
クエリの編集	クエリエディタを使用すると、Interactive SQL でグラフィカルな方法を使用して SELECT 文を作成したり編集したりできる。
プランビューア	プランビューアは、SAP Sybase IQ データベースのグラフィカルなプランを表示するためのグラフィカルなツールである。
オプション	[オプション]ウィンドウを使用すると、Interactive SQL でのコマンド、表示、データのインポートとエクスポート、メッセージの各オプションを設定できる。

**参照：**

- プランの表示 (Interactive SQL のプランビューア) (45 ページ)
- クエリの作成 (クエリエディタ) (43 ページ)

## Interactive SQL のキーボードショートカット

Interactive SQL で利用できるキーボードショートカットについて説明します。

表 14 : Interactive SQL のキーボードショートカット

ファンクションキー	説明
Tab	選択内容を受け入れてテキスト補完ウィンドウを閉じる。
Alt + F4	Interactive SQL を終了する。
Ctrl + A	コンテンツには関係なく、一致したリストを表示する。
Ctrl + C	選択されたローとカラム見出しを [結果] ウィンドウ枠にコピーする。 [SQL 文] ウィンドウ枠では、選択したテキストをクリップボードにコピーする。
Ctrl + End	現在のウィンドウ枠の下部に移動する。
Ctrl + H	実行された SQL 文の履歴を表示する。
Ctrl + Home	現在のウィンドウ枠の先頭に移動する。
Ctrl + N	[Interactive SQL] ウィンドウの内容を消去する。
Ctrl + P	[SQL 文] ウィンドウ枠の内容を印刷する。
Ctrl + Q	クエリエディタを表示する。 クエリエディタは、SQL クエリを作成するために使用できる。クエリの作成が完了したら、[OK] をクリックして、元の [SQL 文] ウィンドウ枠にクエリをエクスポートする。
Ctrl + S	[SQL 文] ウィンドウ枠の内容を保存する。
Ctrl + プラス記号 (+)	対象のアイテムおよびそのパラメータリストを、[SQL 文] ウィンドウ枠に追加する。
Ctrl + アスタリスク (*)	対象のアイテムおよびそのパラメータと型のリストを、[SQL 文] ウィンドウ枠に追加する。
Ctrl + 二重引用符 (")	対象のアイテムを引用符で囲み、[SQL 文] ウィンドウ枠に追加する。
F2	結果セット内で選択した値を編集する。
F5	[SQL 文] ウィンドウ枠内のすべてのテキストを実行する。
F7	[テーブル名のルックアップ] ダイアログを表示する。

ファンクションキー	説明
F8	[プロシージャ名のルックアップ] ダイアログを表示する。
F9	[SQL 文] ウィンドウ枠内の選択されたテキストを実行する。 テキストが選択されていない場合は、すべての文を実行する。
PgDn	現在のペインで 1 ページ下へ移動する。
PgUp	現在のペインで 1 ページ上へ移動する。

以下のキーボードショートカットは、[SQL 文] ウィンドウ枠にフォーカスがあるときにだけ使用できます。

表 15 : [SQL 文] ウィンドウ枠用の Interactive SQL のキーボードショートカット

ファンクションキー	説明
Ctrl + ]	対になるもう一方のカッコにカーソルを移動する。カッコ、中カッコ、角カッコ、およびかぎカッコに使用できる。
Ctrl + Backspace	カーソルの左側にある単語を削除する。
Ctrl + Del	カーソルの右側にある単語を削除する。
Ctrl + G	移動先の行を指定できる [Go To] ダイアログを表示する。
Ctrl + L	現在の行を [SQL 文] ウィンドウ枠から削除し、クリップボードに格納する。
Ctrl + Shift + ]	対になるもう一方のカッコまで選択範囲を広げる。カッコ、大カッコ、角カッコ、およびかぎカッコに使用できる。
Ctrl + Shift + C	<b>COMMIT</b> 文を実行する。
Ctrl + Shift + L	現在の行を削除する。
Ctrl + Shift + R	<b>ROLLBACK</b> 文を実行する。
Ctrl + Shift + U	選択範囲の文字を大文字に変える。
Ctrl + U	選択範囲の文字を小文字に変える。
F3	選択されたテキストと同じテキストが次に現れる場所を見つける。
Home	カーソルを現在行の先頭または現在行の先頭の単語に移動する。
Shift + F3	選択されたテキストと同じテキストが前に現れた場所を見つける。
Shift + Home	現在行のテキストの先頭まで選択範囲を広げる。

## 結果セット

---

実行したコマンドの結果に対処します。

### 複数の結果セットの表示 (UNIX)

デフォルトでは、Interactive SQL は最後に実行した文の最初の結果セットを表示します。UNIX プラットフォームで次の手順を使用して、すべての結果セットを表示します。

1. Interactive SQL をコマンドラインプログラム (-nogui モード) として実行している場合は、次を行います。
  - a) \$HOME ディレクトリに移動し、ファイル `.isqlPreferences11` を見つけます。
  - b) 行 `<entry key="SybaseIQ.showMultipleResultSets">0</entry>` を `<entry key="SybaseIQ.showMultipleResultSets">1</entry>` に変更します。
  - c) 変更した内容を `.isqlPreferences11` に保存します。
2. GUI モードを使用している場合は、次を行います。
  - a) [Interactive SQL] ウィンドウで、[ツール] > [オプション] をクリックします。Interactive SQL の [オプション] ウィンドウが表示されます。
  - b) [SAP Sybase IQ] ページで、[結果] タブを選択します。
  - c) [すべての結果セットを表示] を選択します。
  - d) [OK] をクリックします。

### 複数の結果セットの表示 (Windows)

デフォルトでは、Interactive SQL は最後に実行した文の最初の結果セットを表示します。Windows プラットフォームで次の手順を使用して、すべての結果セットを表示します。

1. [Interactive SQL] ウィンドウで、[ツール] > [オプション] をクリックします。Interactive SQL の [オプション] ウィンドウが表示されます。
2. [SAP Sybase IQ] ページで、[結果] タブを選択します。
3. [すべての結果セットを表示] を選択します。
4. [OK] をクリックします。

## Interactive SQL での結果セットの編集

Interactive SQL でクエリを実行した後では、結果セットをソートしたり編集したりしてデータベースを変更できます。結果セットからローを選択し、他のアプリケーションで使用するためにコピーできます。

Interactive SQL では、ローの編集、挿入、削除を行うことができます。結果セットを編集することは、**UPDATE** 文、**INSERT** 文、**DELETE** 文を実行するのと同じ効果があります。結果セットを編集すると、対応する **INSERT** 文、**UPDATE** 文、**DELETE** 文が Interactive SQL のコマンド履歴に追加されます。

結果セットのローまたは値を編集するには、値を変更したいテーブルまたはカラムに関する適切な権限が必要です。たとえば、ローを削除する場合は、そのローが属するテーブルに対する **DELETE** 権限を持っている必要があります。

次の場合には、結果セットを編集できません。

- プライマリキーを使用してテーブルからカラムを選択しているが、すべてのプライマリキーカラムを選択しているわけではない。
- **JOIN** の結果セットを編集しようとしている (たとえば、複数のテーブルからのデータが結果セットに存在する)。
- 編集が無効にされているテーブルを編集しようとしている。

次の場合は、結果セットの編集は失敗します。

- 権限のないローやカラムを編集しようとしている。
- 無効な値を入力している (たとえば、数値カラムに対する文字列または NULL を許可しないカラムへの NULL)。

編集が失敗すると、エラーを説明する Interactive SQL エラーメッセージが表示され、データベーステーブルの値は変更されません。

### 参照：

- コマンドの呼び出し (41 ページ)

### Interactive SQL の結果セットからのテーブル値の編集

変更するカラムに対して **UPDATE** パーミッションを持っている場合、データベーステーブルの既存のロー内の一部またはすべての値を変更できます。ただし、テーブル編集が無効にされていない必要があります。

1. Interactive SQL 内でクエリを実行します。
2. [結果] タブで、変更する値をクリックします。
3. 値を右クリックして [ローの編集] を選択するか、[F2] キーを押して、結果セットを編集します。

値を含むテーブルセルでカーソルが点滅します。

4. 新しい値を入力します。ローの他の値を編集する場合は、[Tab] キーを押すか、[Shift] キーを押しながら [Tab] キーを押して、他の値に移動します。
5. ローでの編集が完了したら、[Enter] キーを押してデータベースを更新します。  
[Esc] キーを押すことにより、選択した値への変更をキャンセルできます。
6. **COMMIT** 文を実行して、テーブルへの変更を永続的なものにします。

#### 参照：

- テーブル編集の無効化 (35 ページ)

#### テーブル編集の無効化

Interactive SQL の [オプション] ウィンドウを使用して、テーブル編集を無効にできます。

1. [ツール] メニューで、[オプション] を選択し、[SAP Sybase IQ] を選択します。
2. [スクロール可能なテーブル] が選択されていることを確認し、[編集の無効化] を選択します。
3. [OK] をクリックします。
4. クエリを実行します。テーブル編集の変更を有効にするために、新しいクエリを実行する必要があります。

#### 参照：

- Interactive SQL の結果セットからのテーブル値の編集 (34 ページ)

#### Interactive SQL の結果セットからデータベースへのローの挿入

Interactive SQL を使用して、新しいローをテーブルに追加できます。結果セット内のカラム間でタブ移動して、値をローに追加します。新しいローを追加するには、テーブルに対する INSERT パーミッションが必要です。

#### 結果セットへの新しいローの挿入

結果セットを介して新しいブランクローをテーブルに追加します。

1. 結果セットを右クリックして、[ローの追加] を選択します。新しいブランクローが表示されます。ローの最初の値でカーソルが点滅します。
2. 新しい値を入力します。[Tab] キーを押して次のカラムに移動します。

カラムに無効なデータ型の値は入力できません。たとえば、INT データ型を受け付けるカラムに文字列は入力できません。すべてのカラム値を追加するまで、この手順を繰り返します。

3. [Enter] キーを押してデータベースを更新します。

### デフォルト値を持つカラムへの値の挿入

デフォルト値を持つカラムに値を追加する場合、セルエディタのリストには、(DEFAULT) 項目が含まれます。同じく、NULL 値を受け付けるカラムの場合、(NULL) がリストに表示されます。

デフォルト値を挿入する場合は、[(DEFAULT)] を選択します。

カラムが NULL 値を受け付けず、デフォルト値も設定されていない場合は、値を入力する必要があります。

### 結果セットに計算カラムが含まれる場合

結果セットに計算カラムが含まれ、計算カラムの値を指定していない場合、データベースの更新時に値が計算されます。ただし、計算カラムに値を指定すると、データベースは指定した値で更新され、計算カラムの値は計算されません。

### INPUT 文による新しいローの挿入

Interactive SQL の結果セットから新しいローを挿入する方法の代替策として、**PROMPT** 句を指定した **INPUT** 文を使用してローを追加する方法があります。

1. Interactive SQL から、**PROMPT** 句を指定した **INPUT** 文を使用して、ローを追加します。次に例を示します。

```
INPUT INTO Products PROMPT;
```

2. 入力が求められたら、各カラムの値を指定します。

### データベースからのローの削除 (Interactive SQL)

Interactive SQL を使用して、データベースからローを削除することもできます。ローを削除するには、テーブルの DELETE 権限が必要です。

1. 次のいずれかの方法を使用して、削除するローを選択します。
  - a) [Shift] キーを押しながらローをクリックする。
  - b) [Shift] キーを押しながら、上矢印キーまたは下矢印キーを使用する。
2. [Delete] キーを押します。
3. **COMMIT** を実行して、変更を永続的なものにします。



### Interactive SQL の結果セットからのローのコピー

Interactive SQL の結果セットからローを直接コピーし、他のアプリケーションに貼り付けることができます。

1. 目的のローで任意のセルを右クリックし、[データのコピー]>[ロー]を選択します。複数のローをコピーするには、[Ctrl] キーを押しながら対象のローのセルをクリックした後、右クリックして[データのコピー]>[ロー]を選択します。
2. コピーしたローを他のアプリケーションに貼り付けます。

### Interactive SQL の結果セットからのカラムのコピー

Interactive SQL の結果セットからカラムを直接コピーし、他のアプリケーションに貼り付けることができます。

1. 目的のカラムで任意のセルを右クリックし、[Copy Data]>[Columns]を選択します。複数のカラムをコピーするには、[Ctrl] キーを押しながら対象のカラムのセルをクリックした後、右クリックして[Copy Data]>[Columns]を選択します。
2. コピーしたカラムを他のアプリケーションに貼り付けます。

### Interactive SQL の結果セットからのセルのコピー

個々の値のコピー操作では、データのみがクリップボードにコピーされます。カラム見出しはコピーされず、引用も行われません。

1. コピーするセルを右クリックし、[データのコピー]>[セル]を選択します。複数のセルをコピーするには、[Ctrl] キーを押しながらセルをクリックした後、右クリックして[データのコピー]>[セル]を選択します。
2. コピーしたセルを他のアプリケーションに貼り付けます。

### Interactive SQL の結果セットでのカラムのソート

指定したカラムで結果をソートします。

1. [結果] タブでカラム見出しをクリックします。
2. メッセージが表示されたら、残りの結果をフェッチします。

## 結果セットからの SQL 文の生成

結果セット内の選択したローに対する **INSERT** 文、**DELETE** 文、**UPDATE** 文を作成できます。

1. 文を生成するローを選択します。

2. 選択したローを右クリックし、[生成] を選択します。次に、[INSERT 文]、[DELETE 文]、または [UPDATE 文] を選択します。文がクリップボードにコピーされます。

## SQL 文と結果セットの印刷

[SQL 文] ウィンドウ枠の内容とクエリ結果を印刷できます。

1. [SQL 文] ウィンドウ枠でクエリを入力し、必要に応じてクエリを実行します。
2. [Ctrl] キーを押しながら [P] キーを押すか、または [ファイル] > [印刷] を選択します。
3. メッセージが表示されたら、SQL 文と結果のどちらを印刷するかを選択します。

### ヘッダの追加

印刷された SQL 文またはクエリ結果にヘッダを追加できます。

1. [ツール] > [オプション] を選択します。Interactive SQL の [オプション] ウィンドウが表示されます。
2. [エディタ] ページで [印刷] タブをクリックします。
3. [ヘッダ] フィールドで、ヘッダに出力されるテキストを指定します。右矢印をクリックして、ヘッダに含める項目を選択することもできます。

## [お気に入り] リスト

現在のデータベース接続や開いている SQL ファイルを [お気に入り] リストに追加できます。

### お気に入りへの .sql ファイルの追加

頻繁に使用する SQL コマンドファイルを [お気に入り] リストに格納します。[お気に入り] リストはユーザごとに存在し、他のユーザは表示できません。

1. お気に入りに追加する SQL コマンドファイルを開きます。
2. [お気に入り] > [お気に入りに追加] を選択します。
3. [開いたファイル 'filename' の追加] を選択します。[名前] フィールドに、.sql ファイルの名前を入力します。
4. [OK] をクリックします。

### 参照：

- [お気に入り] メニューのウィンドウリファレンス (40 ページ)

## お気に入りへの SQL 文の追加

頻繁に使用する SQL 接続を [お気に入り] リストに保存します。[お気に入り] リストはユーザごとに存在し、他のユーザは表示できません。

1. [SQL 文] ウィンドウ枠で、お気に入りに追加する SQL コマンドを入力します。[SQL 文] ウィンドウ枠内のコンテンツは 16384 文字以下にする必要があります。
2. [お気に入り]-[お気に入りに追加] を選択します。
3. [SQL 文の追加] を選択します。[名前] フィールドにお気に入りの名前を入力します。
4. [OK] をクリックします。

## 接続をお気に入りに追加

頻繁に使用する接続を [お気に入り] リストに格納します。[お気に入り] リストはユーザごとに存在し、他のユーザは表示できません。

1. データベースに接続します。
2. [お気に入り]>[お気に入りに追加] を選択します。
3. [接続パスワードの保存] を選択します。[名前] フィールドに、接続の名前を入力します。
4. [OK] をクリックします。

参照：

- [お気に入り] メニューのウィンドウリファレンス (40 ページ)

## お気に入りの表示

頻繁に使用する SQL コマンドファイルと接続のリストを表示します。

[お気に入り]>[お気に入りを表示] を選択します。

[Interactive SQL] ウィンドウの左側に [お気に入り] ウィンドウ枠が表示されます。

参照：

- [お気に入り] メニューのウィンドウリファレンス (40 ページ)

## お気に入りを開く

お気に入りのリストから、SQL コマンドファイルまたは接続を開きます。

[お気に入り] メニューから、開きたいお気に入りを選択します。

**参照：**

- [お気に入り] メニューのウィンドウリファレンス (40 ページ)

## お気に入りの編集

お気に入りリストを編集します。

1. [お気に入り]>[お気に入りを表示] を選択します。
2. お気に入りを選択し、右クリックしてから [編集] を選択します。
3. ウィンドウの指示に従います。
4. [保存] をクリックします。

## お気に入りのエクスポート

すべてのお気に入りを .fav ファイルにエクスポートします。

1. [お気に入り]>[お気に入りのエクスポート] を選択します。
2. .fav ファイルのファイル名を指定し、[エクスポート] をクリックします。

## お気に入りのインポート

お気に入りを別のコンピュータにインポートするか、インポートファイルをバックアップとして保存します。

1. [お気に入り]>[お気に入りのインポート] を選択します。
2. filename.fav ファイルを参照し、[インポート] をクリックします。

## [お気に入り] メニューのウィンドウリファレンス

[お気に入り] メニューから利用できるウィンドウを使用して、SQL ファイルや接続情報をお気に入りとして管理できます。

表 16 : [お気に入り] メニューのウィンドウ

ウィンドウ	説明
お気に入りに追加	このウィンドウを使用すると、SQL ファイルや接続情報をお気に入りとして保存できる。
お気に入りの整理	このウィンドウを使用すると、お気に入りを保守したり、整理したりできる。
お気に入りの表示	[Interactive SQL] ウィンドウの左側に [お気に入り] ウィンドウを開く。

**参照：**

- お気に入りへの .sql ファイルの追加 (38 ページ)

- 接続をお気に入りに追加 (39 ページ)
- お気に入りの表示 (39 ページ)
- お気に入りを開く (39 ページ)

## コマンドの呼び出し

---

コマンドを実行すると、Interactive SQL によってそのコマンドは履歴リストに自動的に保存されます。このリストは、Interactive SQL のセッション間で保持されます。実行したコマンドをログファイルに記録することもできます。

Interactive SQL では、最大で最新の 50 個のコマンドの記録が維持されます。

コマンドの完全なリストは、[コマンド履歴] ウィンドウで参照できます。最新のコマンドが、リストの下部に表示されます。

### [コマンド履歴] ウィンドウへのアクセス

[コマンド履歴] ウィンドウを開くと、最大で最新の 50 個のコマンドの記録が表示されます。

次のいずれかを実行します。

- [Ctrl] キーを押しながら、[H]キーを押す。
- ツールバーの [過去の SQL 文のリストを開く] ボタンをクリックする。

### コマンドの呼び出し

実行したコマンドをコマンド履歴から呼び出します。

1. [コマンド履歴] ウィンドウを開き、コマンドを選択します。
2. [OK] をクリックします。

---

**注意：** [コマンド履歴] ウィンドウを使用せずにコマンドを呼び出す方法もあります。ツールバーの [前の SQL 文の呼び出し] アイコンと [次の SQL 文の呼び出し] アイコンを使用して、前方または後方のコマンドに移動します。[Alt] キーを押しながら [右矢印] キーを押すか、[Alt] キーを押しながら [左矢印] キーを押して、それぞれ前方または後方のコマンドに移動することもできます。

---

**参照：**

- Interactive SQL での結果セットの編集 (34 ページ)

## [コマンド履歴] ウィンドウからのコマンドのコピー

コマンドを [コマンド履歴] ウィンドウから Interactive SQL の [SQL 文] ウィンドウ 枠にコピーします。

1. [コマンド履歴] ウィンドウを開きます。
2. 1つまたは複数のコマンドを選択し、[Ctrl] キーを押しながら[C] キーを押すか、または [コピー] をクリックします。
3. [OK] をクリックします。  
選択した文が [SQL 文] ウィンドウ枠にコピーされます。

## [コマンド履歴] ウィンドウからのコマンドの保存

コマンド履歴を .sql ファイルとして保存します。

1. [コマンド履歴] ウィンドウを開きます。
2. [履歴を SQL ファイルとして保存] ボタンをクリックするか、[Ctrl] キーを押し ながら [S] キーを押します。
3. [名前を付けて保存] ウィンドウで、ファイルの場所と名前を指定します。コマ ンド履歴ファイルは、.sql の拡張子を持ちます。
4. 作業が終了したら、[保存] をクリックします。

## [コマンド履歴] ウィンドウからのコマンドの削除

コマンド履歴から、1つ、複数、またはすべてのコマンドを削除します。

1. [コマンド履歴] ウィンドウを開きます。
2. 次のいずれかの方法を使用して、コマンドを削除します。
  - 1つまたは複数のコマンドを選択して、[削除] ボタンをクリックする。この 操作は取り消しできません。
  - [履歴のクリア] をクリックして、ウィンドウからすべてのコマンドを削除 する。この操作は取り消しできません。

## コマンドのロギング

実行したコマンドは記録できます。記録されたコマンドはログファイルに格納さ れ、再び使用できます。

1. [SQL]>[ロギングの開始] を選択します。
2. [名前を付けて保存] ウィンドウで、ログファイルの場所と名前を指定します。 たとえば、ファイル名として mylogs.sql を指定します。

3. 作業が終了したら、[保存] をクリックします。
4. [SQL]>[ロギングの停止] を選択します。

## クエリエディタでのクエリの作成

クエリエディタは、Interactive SQL で提供されるツールです。**SELECT** 文を作成するのに役立ちます。

クエリエディタで SQL クエリを作成したり、クエリをインポートして編集したりできます。クエリが完成したら、[OK] をクリックします。クエリが Interactive SQL にエクスポートされ、処理できるようになります。

クエリエディタでクエリを作成する場合に SQL コードを使用する必要はありません。ただし、クエリエディタでは、次の方法で SQL を使用できます。

- Interactive SQL の [SQL 文] ウィンドウ枠でクエリを作成し、クエリエディタにインポートできます。これを行うには、コードを強調表示し、その後クエリエディタを開きます。
- クエリエディタを使用するときにはいつでも、ウィンドウの下部で SQL をクリックすることにより、作成途中のクエリの SQL コードを表示できます。コードを直接編集できます。このとき、フィールドはクエリエディタで自動的に更新されます。

### クエリの作成 (クエリエディタ)

クエリエディタのツールとタブを使用して、SQL クエリを作成します。

1. Interactive SQL からデータベースに接続します。
2. [ツール]>[クエリの編集] を選択します。

Interactive SQL で SQL コードを選択すると、そのコードがクエリエディタに自動的にインポートされます。

3. クエリを作成します。
4. [OK] をクリックして、クエリを [SQL 文] ウィンドウ枠に書き込みます。
5. 提供されているタブを使用して、SQL クエリのコンポーネントを作成します。

表 17: クエリエディタのタブ

タブ	説明
[テーブル] タブ	クエリで使用するテーブルを指定する。

タブ	説明
[ジョイン] タブ	テーブル内のデータを組み合わせるジョイン方法を指定する。クエリに複数のテーブルを含める場合は、テーブル内のデータを組み合わせるジョイン方法を指定する必要がある。[テーブル] タブで追加したテーブルにジョイン方法が指定されていない場合、クエリエディタは1つの方法を提案する。テーブル間に外部キー関係がある場合は、その関係に基づくジョイン条件を生成するか、外積を提案する。クエリを開くと、クエリエディタは指定したとおりのジョイン方法を受け入れる(さらに、指定されていないJOINはKEY JOINのデフォルトとならない)。
[カラム] タブ	結果セットのカラムを指定する。カラムを指定しない場合は、すべてのカラムが表示される。
[INTO] タブ	結果を変数に割り当てる。
[WHERE] タブ	結果セットのローを制限する条件を指定する。
[GROUP BY] タブ	結果セットでローをグループ化する。
[HAVING] タブ	グループ値に基づいて、結果セットのローを制限する。
[ORDER BY] タブ	ローをソートする。

6. 次のツールが使用できます。

表 18 : クエリエディタのツール

ツール	説明
式エディタ	式エディタは、検索条件を作成したり、計算カラムを定義したりするのに使用できる。
抽出テーブル	このウィンドウはメインクエリエディタとほとんど同じである。抽出テーブルやサブクエリを作成できる。

参照：

- プランの表示 (Interactive SQL のプランビューア) (45 ページ)
- [ツール] メニューのウィンドウリファレンス (30 ページ)

## クエリエディタの設定

クエリエディタを設定して、設定をカスタマイズします。

1. [ツール] > [オプション] > [SAP Sybase IQ] を選択します。
2. [クエリエディタ] タブをクリックし、設定を変更します。



## クエリエディタの制限事項

クエリエディタでサポートされていない機能と構文について説明します。

クエリエディタは **SELECT** 文を作成します。ビューを作成するには設計されていません (ただし、Interactive SQL でビューを作成してからクエリエディタで参照することはできます)。**UPDATE** 文または他の非 **SELECT SQL** 文を作成することもできません。作成されるのは、単一の **SELECT** 文のみです。そのため、**SELECT** 文の **UNION** や **INTERSECT** は作成されません。また、クエリエディタでは Transact-SQL® 構文はサポートしていません。

## プランの表示 (Interactive SQL のプランビューア)

プランビューアは、データベースのグラフィカルなプランを表示するためのグラフィカルなツールです。

[プランビューア] ウィンドウは、複数のウィンドウ枠で構成されています。

表 19: プランビューアのウィンドウ枠

ウィンドウ枠	説明
[SQL] ウィンドウ枠	プランを生成する SQL 文を入力するための領域。
[結果] ウィンドウ枠	グラフィカルなプランを表示する。
[詳細] ウィンドウ枠	プランについての詳細をテキストで表示する。

### 参照:

- [Interactive SQL] ウィンドウリファレンス (29 ページ)
- [ツール] メニューのウィンドウリファレンス (30 ページ)
- クエリの作成 (クエリエディタ) (43 ページ)

## プランビューアの起動

プランビューアを起動すると、別のウィンドウが開きます。

1. Interactive SQL を開きます。
2. [ツール] > [プランビューア] を選択するか、[Shift] キーを押しながら [F5] キーを押します。

### 参照:

- グラフィカルなプランの設定 (46 ページ)

- プランの印刷 (47 ページ)
- Interactive SQL でのグラフィカルなプランの表示 (46 ページ)

## Interactive SQL でのグラフィカルなプランの表示

SQL 文に対するクエリオプティマイザの実行プランを Interactive SQL の [プランビューア] で表示できます。

1. [SQL 文] ウィンドウ枠でクエリを入力します。
2. [ツール]>[プランビューア] を選択するか、[Shift] キーを押しながら [F5] キーを押します。  
プランビューアが別のウィンドウに表示されます。指定したクエリが [SQL] ウィンドウ枠に表示されます。
3. [プランの取得] をクリックして、指定したクエリのプランを生成します。
4. [ツール]>[プランビューア] を選択します。
5. [開く] をクリックします。
6. プランファイル (.saplan) を選択し、[開く] をクリックします。

### 参照：

- グラフィカルなプランの設定 (46 ページ)
- プランの印刷 (47 ページ)
- プランビューアの起動 (45 ページ)

## グラフィカルなプランの設定

グラフィカルなプランの実行後、プランの項目の表示形式をカスタマイズできません。

1. プランビューアの左下のウィンドウ枠でプランを右クリックし、[カスタマイズ] を選択します。
2. 設定を変更します。
3. 完了したら、[OK] をクリックします。
4. [プランの取得] をクリックして、変更が反映されたグラフィカルなプランを生成します。

### 参照：

- プランの印刷 (47 ページ)
- プランビューアの起動 (45 ページ)
- Interactive SQL でのグラフィカルなプランの表示 (46 ページ)

## プランの印刷

プランビューアでプランを印刷できます。

1. [ツール]>[プランビューア] を選択します。
2. [印刷] ボタンをクリックするか、プランを右クリックして [印刷] を選択します。

### 参照：

- グラフィカルなプランの設定 (46 ページ)
- プランビューアの起動 (45 ページ)
- Interactive SQL でのグラフィカルなプランの表示 (46 ページ)

## ソース制御の統合

Interactive SQL はサードパーティのソース制御システムとの統合が可能であり、ファイルに関して共通するソース管理操作を Interactive SQL 内から実行できます。

Windows 上では、Interactive SQL は、Microsoft Common Source Code Control API (SCC) をサポートする、Microsoft Visual SourceSafe を含む多くのソース制御製品と統合可能です。Windows や他のオペレーティングシステムで SCC API をサポートしないソース制御製品を使用するには、コマンドラインを指定して、各ソース制御操作を実行します。それらのコマンドの出力が、ログウィンドウに表示されません。

Interactive SQL は、次のタスクをサポートします (タスクがソース制御製品でサポートされている場合)。

- ソース制御プロジェクトのオープン
- 取得
- チェックイン
- チェックアウト
- チェックアウトの取り消し
- バージョンの比較
- ファイル履歴の表示
- ファイルプロパティの表示
- ソース制御マネージャの実行

基になるソース制御プログラムで操作がサポートされていない場合は、対応するメニュー項目が無効になります。たとえば、Visual SourceSafe はこれらの操作をすべてサポートしていますが、カスタム (コマンドライン) ソース制御システムを使

用する場合、ソース制御プロジェクトのオープンやソース制御マネージャの実行はサポートされません。

ソース制御プログラムの操作に慣れた後に、Interactive SQL から使用するようになしてください。

### ソース制御を使用するための Interactive SQL の設定

ファイルのチェックインやチェックアウト、ファイルの異なるバージョンの比較、ファイルの履歴の表示など、ソース制御操作を行う前に、ソース制御を使用するように Interactive SQL を設定する必要があります。

Microsoft SCC API をサポートするソース制御製品が搭載されている Windows コンピュータで Interactive SQL を実行している場合は、その製品を使用することも、カスタム (コマンドライン指向) システムを使用することもできます。

#### Windows での Interactive SQL ソース制御の設定

Microsoft SCC API をサポートするソース制御製品が搭載されている Windows コンピュータで Interactive SQL を実行している場合、ソース制御の統合を有効にします。

1. [ツール]>[オプション] を選択します。
2. 左側のウィンドウ枠で、[ソース制御] をクリックします。
3. [ソース制御の統合を有効にする] をクリックします。
4. [OK] をクリックします。

#### コマンドラインインタフェースを使用する Interactive SQL ソース制御システムの設定

カスタムコマンドライン指向のソース制御製品を搭載しているコンピュータで Interactive SQL を実行している場合は、ソース制御の統合を有効にします。

1. [ツール]>[オプション] をクリックします。
2. 左側のウィンドウ枠で、[ソース制御] をクリックします。
3. [ソース制御の統合を有効にする] をクリックします。
4. [設定] をクリックします。
5. [カスタムソース制御オプション] ウィンドウで、[リセット] をクリックします。
6. リストから使用するソース制御システムを選択し、[OK] をクリックします。
7. 必要に応じて、リストのコマンドを編集します。これを行うには、[ソース制御操作] リストから操作を選択し、対応するコマンドを [コマンドライン] ウィンドウ枠に入力します。

システムのコマンドを [ソース制御操作] リストで定義する場合、プレースホルダ [FILENAME] を使用して、コマンドを実行するとき使用するファイルの名前を表します。

操作に対してコマンドラインを指定しない場合、[ファイル]>[ソース制御] メニューの項目は無効になります。

8. [OK] をクリックし、再び [OK] をクリックします。

## Interactive SQL からのソース制御プロジェクトのオープン

一部のソース制御製品では、ソース制御プロジェクトを開いた後に他のソース制御操作を実行する必要があります。

プロジェクトの正確な定義は、使用するソース制御システムによって異なります。通常、プロジェクトは、ソース制御下の一連のファイルと、ファイルの作業コピーが配置されているローカルファイルシステムのロケーションで構成されます。プロジェクトを開くには、一般的に、ソース制御システムへの認証情報(ユーザー ID とパスワードなど)を提供する必要があります。

使用しているソース制御システムがソース制御プロジェクトのオープンをサポートしている場合、[ファイル]>[ソース制御]>[ソース制御プロジェクトのオープン]メニュー項目が有効になります。[ファイル]メニューからこのオプションを選択すると、プロジェクトを開くためのソース制御固有のウィンドウが開きます。プロジェクトを一度開くと、その後の Interactive SQL セッションでは再び開く必要はありません。プロジェクトは自動的に開きます。

### 参照：

- その他のソース制御操作 (50 ページ)
- Interactive SQL からのファイルのチェックアウト (49 ページ)
- Interactive SQL からのファイルのチェックイン (50 ページ)

## Interactive SQL からのファイルのチェックアウト

Interactive SQL でファイルを開いたら、[ファイル]メニューのコマンドを使用して、ファイルをチェックアウトします。

1. [ファイル]>[開く]を選択し、開くファイルを参照します。

ファイルステータス ([チェックイン済み]、[チェックアウト済み]、[非制御下]) が [Interactive SQL] ウィンドウの下部のステータスバーに表示されます。

2. [ファイル]>[ソース制御]>[チェックアウト]を選択してファイルをチェックアウトします。
3. 使用しているソース制御製品によっては、チェックアウトプロセスの一環としてコメントまたは他のオプションの入力が求められることがあります。

---

**注意：** SCC 準拠のソース制御システムを使用している場合、ステータスは常に正確です。ただし、カスタムソース制御システムを使用している場合、ステータスはファイルが読み取り専用なのかどうかによって異なります。読み取り専用ファイルは、チェックインされているとみなされますが、編集可能なファイルについては、チェックアウトされるか、非制御下となる可能性があります、想定は行われません。

---

**参照：**

- Interactive SQL からのソース制御プロジェクトのオープン (49 ページ)
- その他のソース制御操作 (50 ページ)

**Interactive SQL からのファイルのチェックイン**

ファイルの編集が完了したら、Interactive SQL からファイルをチェックインできます。

1. [ファイル]>[ソース制御]>[チェックイン] を選択します。
2. 入力が求められたら、チェックインコメントを入力します。

**参照：**

- Interactive SQL からのソース制御プロジェクトのオープン (49 ページ)
- その他のソース制御操作 (50 ページ)

**その他のソース制御操作**

ソース制御プロジェクトのオープン、ファイルのチェックイン/チェックアウトだけでなく、Interactive SQL では、他にもいくつかのソース制御操作がサポートされています。使用できる操作は、使用するソース制御システムによって異なります。

これらの操作には、Interactive SQL の [ファイル]>[ソース制御] メニューからアクセスします。

**表 20 : その他のソース制御操作**

アクション	説明
取得	[SQL 文] ウィンドウ枠で現在開いているファイルの最新のコピーを取得する。
チェックアウトの取り消し	ファイルの作業コピーを破棄し、ソース制御アーカイブに存在するファイルのコピーをダウンロードする。
バージョンの比較	開いているファイルの作業コピーとソース制御アーカイブのバージョンを比較する。

アクション	説明
履歴	開いているファイルに実行されたソース制御操作 (通常、チェックイン) のリストを表示する。
プロパティ	開いているファイルに関連付けられているソース制御プロパティのリストを表示する。
ソース制御マネージャの実行	ソース制御システムの管理プログラムを起動する。たとえば、Microsoft Visual SourceSafe を実行している場合、Visual SourceSafe エクスプローラを起動する。

**参照：**

- Interactive SQL からのソース制御プロジェクトのオープン (49 ページ)
- Interactive SQL からのファイルのチェックアウト (49 ページ)
- Interactive SQL からのファイルのチェックイン (50 ページ)

## Interactive SQL のための SQL 文

---

Interactive SQL 文は、Interactive SQL 内でのみ使用できます。

### **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」 (51 ページ)

「標準」 (52 ページ)

「パーミッション」 (52 ページ)

#### **構文**

**CLEAR**

#### **使用法**

(先頭に戻る) (51 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) (51 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) (51 ページ)

なし

## **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」 (52 ページ)

「標準」 (52 ページ)

「パーミッション」 (53 ページ)

### 構文

#### **CONFIGURE**

### 使用法

(先頭に戻る) (52 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) (52 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) (52 ページ)

なし

## 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」 (53 ページ)

「例」 (54 ページ)

「使用法」 (54 ページ)

「標準」 (55 ページ)

「パーミッション」 (56 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) (53 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) (53 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) (53 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** 文の前に置いてもかまいません。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続けます。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 **-l** 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。**-l** 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) (53 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。

## dbisql Interactive SQL ユーティリティ

- 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) (53 ページ)

なし

## 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」 (56 ページ)

「例」 (56 ページ)

「使用法」 (57 ページ)

「標準」 (57 ページ)

「パーミッション」 (57 ページ)

### 構文

```
DISCONNECT [ { connection-name | CURRENT | ALL } ]
```

### パラメータ

(先頭に戻る) (56 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) (56 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) (56 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) (56 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) (56 ページ)

なし

## 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」 (58 ページ)

「例」 (60 ページ)

「使用法」 (60 ページ)

「標準」 (61 ページ)

「パーミッション」 (61 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) (57 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  $\%n$  は改行文字を表し、 $\%¥$  は単一の円記号 (¥) を表します。 $\%xDD$  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) (57 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) (57 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) (57 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) (57 ページ)

なし

## 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：

「例」 (61 ページ)

「使用法」 (62 ページ)

「標準」 (62 ページ)

「パーミッション」 (62 ページ)

### 構文

```
PARAMETERS parameter1, parameter2, ...
```

### 例

(先頭に戻る) (61 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
```

## dbisql Interactive SQL ユーティリティ

```
WHERE DepartmentID = {department_id}
>#{file}.dat;
```

### 使用法

(先頭に戻る) (61 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) (61 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) (61 ページ)

なし

## 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」 (63 ページ)

「例」 (64 ページ)

「標準」 (65 ページ)

「パーミッション」 (65 ページ)

### 構文

```
READ [ ENCODING encoding ] filename [ parameter ] ...
```

## パラメータ

(先頭に戻る) (62 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) (62 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' ]
```

- **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) (62 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) (62 ページ)

なし

## 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」 (65 ページ)

「例」 (65 ページ)

「使用法」 (66 ページ)

「標準」 (66 ページ)

「パーミッション」 (66 ページ)

### 構文

```
SET CONNECTION [connection-name]
```

### パラメータ

(先頭に戻る) (65 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) (65 ページ)

- **例 1** – 次は、Embedded SQL の例です。

```
EXEC SQL SET CONNECTION :conn_name
```

- **例 2 – dbisql** 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) (65 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) (65 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) (65 ページ)

なし

## 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」 (67 ページ)

「使用法」 (67 ページ)

「パーミッション」 (67 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) (66 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) (66 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) (66 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。

## 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」 (68 ページ)

「例」 (68 ページ)

「使用法」 (69 ページ)

「標準」 (69 ページ)

「パーミッション」 (69 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) (68 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。
- **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) (68 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。
- **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 '/s1/sybase/sample_2.db'
```

```
START DATABASE 'c:¥sybase¥sample_2.db'
AS sam2
ON eng1
```

## 使用法

(先頭に戻る) (68 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) (68 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) (68 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」 (70 ページ)

「例」 (70 ページ)

「使用法」 (70 ページ)

「標準」 (70 ページ)

「パーミッション」 (70 ページ)

### 構文

```
START ENGINE AS engine-name [ STARTLINE command-string ]
```

### パラメータ

(先頭に戻る) (69 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) (69 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) (69 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) (69 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) (69 ページ)

なし

## 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」 (71 ページ)

「例」 (71 ページ)

「標準」 (71 ページ)

「パーミッション」 (72 ページ)

### 構文

```
STOP DATABASE database-name
... [ ON engine-name ]
... [ UNCONDITIONALLY ]
```

### パラメータ

(先頭に戻る) (71 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) (71 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) (71 ページ)

- SQL - ISO/ANSI SQL 文法のベンダ拡張。

## dbisql Interactive SQL ユーティリティ

- SAP Sybase Database 製品 - なし。

### パーミッション

(先頭に戻る) (71 ページ)

SERVER OPERATOR システム権限が必要です。

## 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」 (72 ページ)

「例」 (72 ページ)

「標準」 (72 ページ)

「パーミッション」 (73 ページ)

### 構文

```
STOP ENGINE engine-name [ UNCONDITIONALLY ]
```

### パラメータ

(先頭に戻る) (72 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。

### 例

(先頭に戻る) (72 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) (72 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) (72 ページ)

なし

## Interactive SQL 用の SQL オプション

Interactive SQL のオプションは、SAP Sybase IQ SQL のオプションのサブセットです。これらを使用して、データベースに対する Interactive SQL の対応を変更します。

### 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'
```

**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、ユーザー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ー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ー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。

参照：

- SQL 文のキャンセル (27 ページ)

## dblocate データベース管理ユーティリティ

**dblocate** ユーティリティは、TCP/IP ネットワーク上に直接接続されているデータベースを検出して、接続の問題を診断するためのサーバ検出ユーティリティです。

### 構文

```
dblocate [ options ]
```

### パラメータ

次の表に、**dblocate** ユーティリティの利用可能なオプションを示します。

表 21 : **dblocate** のオプション

オプション	説明
<b>-d</b>	検出した各サーバのサーバ名とアドレスをリストし、その後にはそのサーバで実行されているデータベースのカンマで区切られたリストが続く。リストが 160 文字を超える場合、トランケートされ、省略記号 (...) で終わる。
<b>-dn database-name</b>	指定された名前のデータベースを実行しているサーバの名前とアドレスをリストする。リストが 160 文字を超える場合、トランケートされ、省略記号 (...) で終わる。
<b>-dv</b>	検出された各サーバの名前とアドレスを表示する。さらに、サーバで実行されている各データベースが別の行に表示される。リストはトランケートされないため、このオプションは、v オプションの使用によりトランケートされたリストの表示にも使用できる。
<b>-n</b>	出力にコンピュータ名ではなく IP アドレスをリストする。コンピュータ名の検索には時間がかかることがあるため、このオプションによりパフォーマンスが向上する可能性がある。
<b>-o filename</b>	出力メッセージを指定した名前のファイルに書き込む。
<b>-p portnumber</b>	指定された TCP/IP ポート番号を使用しているサーバのみのサーバ名とアドレスを表示する。TCP/IP ポート番号は、1 ~ 65535 の範囲である必要がある。
<b>-q</b>	クワイエットモードで実行する (メッセージなし)。

## dblocate データベース管理ユーティリティ

オプション	説明
-s <i>name</i>	指定されたサーバ名を使用しているサーバのみのサーバ名とアドレスを表示する。このオプションを使用する場合は -ss オプションは使用できない (両方のオプションを使用すると、一致するサーバが見つからない可能性がある)。
-ss <i>substr</i>	サーバ名に指定されたサブストリングが含まれているサーバのみのサーバ名とアドレスを表示する。このオプションを使用する場合は -s オプションは使用できない (両方のオプションを使用すると、一致するサーバが見つからない可能性がある)。
-v	完全なサーバ名を表示する。デフォルトでは、dblocate は 40 バイトを超えるデータベースサーバ名をトランケートする。

### 使用法

**dblocate** は、TCP/IP ネットワークに直接接続して実行されている SQL Anywhere または SAP Sybase IQ データベースサーバを検出します。データベースサーバとそのアドレスのリストを出力します。

ネットワークによって、結果の出力に数秒かかる場合があります。

### 参照：

- start\_iq のデータベースオプション (222 ページ)



## dblog データベース管理ユーティリティ

**dblog** ユーティリティは、データベースに関連付けられたトランザクションログやトランザクションログミラーの名前を表示したり、変更したりするトランザクションログユーティリティです。

**dblog** を使用すると、データベースによるトランザクションログミラーの管理を停止したり、開始したりすることもできます。

SAP Sybase IQ は、データベースのトランザクションログの作成および削除処理を自動的に実行します。データベースの実行時には、トランザクションログを生成する必要があります。トランザクションログがないと、サーバは起動しません。トランザクションログミラーはトランザクションログの重複コピーであり、データベースと並行して管理されます。

---

**注意：** トランザクションログミラーは必須ではありませんが、これを使用してください。SAP Sybase IQ データベースのバックアップを頻繁に行わない場合は特に、この使用を推奨します。

---

### 構文

```
dblog [options] database-file
```

### パラメータ

次の表に、**dblog** ユーティリティの利用可能なオプションを示します。

表 22 : dblog のオプション

オプション	説明
<b>-ek key</b>	強い暗号で保護されたデータベースの暗号化キーをコマンドで直接指定する。強い暗号で保護されたデータベースがある場合は、データベースまたはトランザクションログを使用するために、なんらかの方法で暗号化キーを指定する必要がある。強い暗号で保護されたデータベースの場合、 <b>-ek</b> または <b>-ep</b> を指定する必要がある (両方を同時に指定することはできない)。強い暗号で保護されたデータベースを使用するとき正しいキーを指定しないと、コマンドはエラーになる。

オプション	説明
<b>-ep</b>	暗号化キーを求めるプロンプトが必要であることを指定する。ダイアログボックスが表示されるので、そこに暗号化キーを入力する。暗号化キーがクリアテキストとして表示されないため、セキュリティをさらに強化できる。強い暗号で保護されたデータベースの場合、-ek または -ep を指定する必要がある(両方を同時に指定することはできない)。強い暗号で保護されたデータベースを使用するときに正しいキーを指定しないと、コマンドはエラーになる。
<b>-g n</b>	Replication Server インストールに参加するために Log Transfer Manager を使用する場合はこのオプションを使用する。バックアップの復元後に使用して、世代番号を設定できる。
<b>-il</b>	このデータベース上の Replication Server インストールへの参加に Log Transfer Manager を使用しなくなったが、SQL Remote または MobiLink 同期を引き続き使用する場合はこのオプションを使用する。delete_old_logs オプション用に保存した Log Transfer Manager ログのオフセットをリセットするため、不要になったトランザクションログを削除できる。
<b>-is</b>	このデータベース上の MobiLink 同期の使用を停止したが、Log Transfer Manager または SQL Remote を引き続き使用する場合はこのオプションを使用する。delete_old_logs オプション用に保存した MobiLink ログのオフセットをリセットするため、不要になったトランザクションログを削除できる。
<b>-m <i>mirror-name</i></b>	新しいトランザクションログミラーのファイル名を指定する。データベースがトランザクションログミラーを現在使用していない場合、その使用を開始する。すでにトランザクションログミラーを使用している場合、データベースはトランザクションログミラーとして新しいファイル名を使用するように変更する。
<b>-n</b>	トランザクションログ、およびトランザクションログミラーの使用を停止する。トランザクションログなしでは、データベースはデータの複製に参加することも、データリカバリでトランザクションログを使用することもできない。SQL Remote、Log Transfer Manager、または dbmlsync トランケートオフセットが存在する場合、対応する「無視」オプション (Log Transfer Manager の場合は -il、SQL Remote の場合は -ir、dbmlsync の場合は -is) も指定されていないかぎり、トランザクションログは削除できない。データベースで監査がオンになっている場合は、それを先にオフにしないかぎり、トランザクションログの使用を停止できない。
<b>-o <i>filename</i></b>	出力メッセージを、指定した名前のファイルに書き込む。
<b>-q</b>	クワイエットモードで実行する (メッセージなし)。

オプション	説明
-r	トランザクションログミラーを保持するデータベースの単一のトランザクションログを保持する。
-t log-name	新しいトランザクションログのファイル名を指定する。データベースがトランザクションログを現在使用していない場合、その使用を開始する。すでにトランザクションログを使用している場合、データベースはトランザクションログとして新しいファイル名を使用するように変更する。
-x n	データベースが複写に参加できるように、トランザクションログの現在の相対オフセットをnにリセットする。このオプションは、SQL Remote 統合データベースの再ロードに使用される。
-z n	データベースが複写に参加できるように、トランザクションログの開始オフセットをnにリセットする。このオプションは、SQL Remote 統合データベースの再ロードに使用される。

### 使用法

**dblog** コマンドラインユーティリティを使用すると、データベースに関連付けられたトランザクションログやトランザクションログミラーの名前を表示したり、変更したりできます。トランザクションログミラーの管理を停止したり、開始したりすることもできます。

データベースを作成するときに、トランザクションログの名前を設定します。データベースの実行中は、トランザクションログファイル名を変更できません。

**RESTORE DATABASE** 文を使用してデータベースの移動や名前の変更を行う場合、トランザクションログ以外のすべてのファイルの名前を変更できます。データベースのリストア後のカタログストアファイル(.db ファイル)の配置場所で、元のログファイル名にトランザクションが引き続き書き込まれます。

データベースのすべてのファイルの名前変更または移動を行う場合は、ログファイルに対しても同じ操作を実行してください。ログファイルの移動または名前変更には、**dblog** を使用します。このユーティリティは、以下を指定して **RESTORE DATABASE** を使用した後で実行します。

- 新規データベース名
- **RENAME** オプション

データベースをリストアしていなくても、**dblog** を使用してトランザクションログの名前を変更できます。ただし、次の制約があります。

- SAP Sybase IQ サーバを停止する必要があります。

## **dblog** データベース管理ユーティリティ

- ログ名の変更後、メディア障害からリカバリするために古いログが必要になる場合に備え、次回のデータベースバックアップまで古いログを保管しなければなりません。

**dblog** は、トランザクションログに関する次のような追加の情報を表示します。

- バージョン番号
- レプリケーションで使用するオフセットの開始
- レプリケーションで使用するオフセットの終了
- ページサイズ
- ページの総数
- 空のページ数
- 使用されているログファイルの割合

# dbping データベース管理ユーティリティ

**dbping** ユーティリティは、接続の問題の診断に使用する ping ユーティリティです。

## 構文

```
dbping [options]
```

## パラメータ

次の表に、**dbping** で使用できるオプションを示します。

表 23 : **dbping** のオプション

オプション	説明
<b>-c</b> "keyword=value; ..."	データベース接続パラメータを指定する。『管理：データベース』の「付録：接続パラメータと通信パラメータリファレンス」を参照。接続パラメータを指定しないと、SQLCONNECT 環境変数が設定されている場合に、その接続パラメータが使用されます。
<b>-d</b>	サーバが見つかった場合にデータベースに接続する。 <b>-d</b> オプションを指定しない場合、 <b>dbping</b> は、 <b>-c</b> オプションで指定したサーバが見つかった場合に成功を報告する。 <b>-d</b> オプションを指定した場合、 <b>dbping</b> は、サーバとデータベースの両方に接続した場合にのみ成功を報告する。 たとえば、blair という名前のサーバでデータベース iqdemo を実行している場合、次のコマンドは成功する。 <pre>dbping -c "eng=blair;dbn=iqdemo"</pre> 次のコマンドは失敗し、「Ping database failed -- specified database not found」というメッセージが表示される。 <pre>dbping -d -c "eng=blair;dbn=iqdemo"</pre>
<b>-en</b>	指定したプロパティのいずれかについて NULL が返された場合に、失敗のリターンコードで <b>dbping</b> が終了するように指定する。デフォルトでは、 <b>-pc</b> 、 <b>-pd</b> 、または <b>-ps</b> で指定されたプロパティの値が不明の場合、NULL を出力し、成功のリターンコードで終了する。このオプションは、 <b>-pc</b> 、 <b>-pd</b> 、 <b>-ps</b> でのみ使用できる。

オプション	説明
-library	<p>指定した ODBC ドライバまたはドライバマネージャライブラリを使用する。使用するライブラリを指定する (ファイル拡張子は省く)。このオプションを使用すると ODBC ドライバマネージャは使用されないの、UNIX オペレーティングシステムでは特に便利である。</p> <p>たとえば、次のコマンドは ODBC ドライバを直接ロードする。</p> <pre>dbping -m -c "dsn=IQ16iqdemo" -l dbodbc11</pre> <p><b>dbping</b> を使用すると、UNIX システムで ODBC ドライバマネージャとの接続を確認できる。Interactive SQL やその他のツールとは異なり、<b>dbping</b> では、サードパーティ製のツールと同じようにコンポーネントを明示的にテストできる。次に例を示す。</p> <pre>dbping -m -c "dsn=dsnname" -l /&lt;full path&gt;/libodbc.so</pre> <p>libodbc.so は、サードパーティ製の ODBC ドライバである。</p>
-m	<p>ODBC ドライバマネージャを使用する。それ以外の場合は、Embedded SQL™ を使用して接続する。ODBC を使用して接続を確立する。デフォルトで、<b>dbping</b> は、Embedded SQL インタフェースを使用して接続しようとする。</p>
-o filename	<p>出力メッセージをファイルに記録する。</p>
-pc property, ...	<p>接続時に、指定された接続プロパティを表示する。プロパティは、カンマで区切られたリストで指定する。データベース接続を確立するために必要な接続情報を指定する必要がある。</p>
-pd property, ...	<p>接続時に、指定されたデータベースプロパティを表示する。プロパティは、カンマで区切られたリストで指定する。データベース接続を確立するために必要な接続情報を指定する必要がある。</p>
-ps property, ...	<p>接続時に、指定されたデータベースサーバプロパティを表示する。プロパティは、カンマで区切られたリストで指定する。</p>
-q	<p>クワイエットモードで動作する (メッセージ表示なし)。<b>dbping</b> がエラーになると、メッセージが常に表示される。</p>

オプション	説明
<b>-s</b>	dbping を実行するコンピュータとデータベースサーバを実行中のコンピュータ間のネットワークのパフォーマンスに関する情報を返す。およその接続速度、遅延時間、およびスループットが表示される。サーバ上のデータベースに接続するため接続パラメータを指定するには、通常、 <b>-c</b> オプションが必要。dbping <b>-s</b> は Embedded SQL 接続にのみ使用できる。 <b>-m</b> または <b>-l</b> が指定されている場合、このオプションは無視される。デフォルトでは、dbping <b>-s</b> は、測定する統計ごとに少なくとも 1 秒間要求をループする。接続および切断の繰り返しは、リソースを過剰に消費することを防ぐために、所要時間にかかわらず最大 200 回まで実行される。速度の遅いネットワークでは、各統計の最小数の繰り返しを実行するのに数秒かかることがある。パフォーマンス統計は概算であり、クライアントコンピュータとサーバコンピュータがほとんどアイドル状態にあるときにより正確になる。
<b>-st time</b>	このオプションは、時間の長さを秒で指定すること、および dbping が測定する統計ごとに要求をループすることを除き、 <b>-s</b> と同じである。このオプションでは、 <b>-s</b> よりも正確なタイミング情報を取得できる。
<b>-z</b>	デバッグ情報を表示する。このオプションは、Embedded SQL 接続を試み場合にのみ使用できる。つまり、 <b>-m</b> または <b>-l</b> とともに使用できない。このオプションを指定すると、接続に使用したネットワーク通信プロトコルと他の診断メッセージが表示される。

### 使用法

**dbping** は、接続問題のデバッグに使用します。接続文字列の全体または一部を入力すると、サーバまたはデータベースを見つける (接続する) 試みが成功したかどうかを示すメッセージを返します。

**dbping** は Embedded SQL または ODBC 接続で使用します。dbping は jConnect (TDS) 接続には使用できません。

終了コードは、0 (正常) または 0 以外 (エラー) です。

このユーティリティは、@filename パラメータを受け付けます。

## **dbping** データベース管理ユーティリティ



## dbstop データベース停止ユーティリティ

**dbstop** は、データベースサーバを停止します。dbstop ユーティリティは、コマンドラインでのみ使用できます (UNIX と Windows の両方のプラットフォームで使用可能)。

UNIX では、**dbstop** はネットワーク上のどのノードのサーバでも停止できます。*server-name* のほか、サーバの起動時に指定した接続パラメータもすべて指定する必要があります。正しい接続パラメータを指定しないと、**dbstop** はサーバに接続して停止できません。

**dbstop** コマンドラインオプションは、アクティブな接続があるときにもサーバを停止するかどうかを制御します。

SERVER は、**dbstop** のデフォルトの論理サーバです。

### データベースサーバの停止 (dbstop)

コマンドラインから **dbstop** を実行します。

次の形式でコマンドを発行します。

```
dbstop [ options ] server-name
```

たとえば、サーバ **myserver** 上の **iqdemo** という名前のデータベースを停止するには、次のコマンドを入力します。

```
dbstop -c "uid=DBA;pwd=sql;eng=myserver;dbn=iqdemo"
```

次の例は、データベースが実行中であるかどうかに関係なく、サーバ **myserver** を停止します。

```
dbstop -c "uid=DBA;pwd=sql;eng=myserver;dbn=utility_db"
```

### dbstop オプション

次の表は、**dbstop** ユーティリティの利用可能なオプションを示します。

表 24 : dbstop オプション

スイッチ	説明
@ filename	指定した環境変数または設定ファイルからオプションを読み込む。

## dbstop データベース停止ユーティリティ

スイッチ	説明
server-name	停止する実行中のサーバのサーバ名。サーバ名を指定する場合、接続パラメータも一緒に指定しないようにする。
-c "keyword=value; ..."	<p>ネットワークサーバを停止するには、サーバを停止するパーミッションを持つユーザ ID を接続文字列に指定する。デフォルトでは、ネットワークサーバに対する SERVER OPERATOR システム権限が必要であり、すべてのユーザがパーソナルサーバを停止できる。-gk サーバコマンドラインオプションを使用することによって、デフォルトの動作を変更できる。</p> <p>『管理：データベース』の「付録：接続パラメータと通信パラメータリファレンス」を参照。</p> <p>マルチプレックスサーバを停止するには、そのサーバへの接続を行った論理サーバコンテキストでサーバを停止するパーミッションが必要。デフォルトでは、dbstop は SERVER コンテキストでログインする。そのため、SERVER 論理サーバにログインするパーミッションが必要となる。</p> <p>アクティブな接続がある場合、dbstop は、そのサーバを停止するかどうかを尋ねるプロンプトを表示する。unconditional=true をコマンドラインに指定すると、サーバはアクティブな接続があるときでも、プロンプトを表示しないで停止する。</p> <p>接続文字列内の論理サーバの上書きは避けること。上書きするユーザのログインポリシーでログインリダイレクションが有効になっていると、接続がリダイレクトされたときに、サーバが停止することがある。</p>
-d	データベースサーバは停止しない。接続文字列で指定されたデータベースのみを停止する。
-o filename	出力メッセージを、指定した名前のファイルに記録する。
-q	クワイエットモード - メッセージは表示されない。
-x	アクティブな接続がある場合は停止しない。このオプションを含めると、アクティブな接続がある場合に dbstop によってプロンプトが表示されない。
-y	アクティブな接続がある場合でもプロンプトを表示せずに停止する。

## dbtran データベース管理ユーティリティ

ログ変換ユーティリティ **dbtran** を使用すると、トランザクションログを `.sql` コマンドファイルに変換できます。

### 構文

データベースサーバに対して処理を実行する場合

```
dbtran [ options ] -c { connection-string } -n SQL-file
```

トランザクションログに対して処理を実行する場合

```
dbtran [ options ] [ transaction-log ] [ SQL-file ]
```

### パラメータ

オプション	説明
@data	指定した環境変数または設定ファイルからオプションを読み込む。
-a	コミットされていないトランザクションをトランザクションログに表示するかどうかを制御する。トランザクションログには、トランザクションによって実行された最新の <b>COMMT</b> の前に加えられた変更が記録される。-a を指定しないと、出力ファイルにはコミットされたトランザクションのみが表示される。-a を指定した場合は、トランザクションログで検出された、コミットされていないトランザクションも表示される。
-c "keyword=value; ..."	このユーティリティをデータベースサーバに対して実行する場合に、接続文字列を指定する。
-d	トランザクションを古いものから新しいものへと順に書き出すことを指定する。この機能はデータベースアクティビティの監査目的で使用される。 <b>dbtran</b> の出力をデータベースに対して適用しないこと。

オプション	説明
-ek <i>key</i>	<p>強力に暗号化されているデータベースの暗号化キーを指定する。強い暗号で保護されたデータベースがある場合は、データベースまたはトランザクションログを使用するときに暗号化キーを指定する必要がある。-ek または -ep を指定すること。ただし、両方を指定することはできない。正しい暗号化キーを指定しないとコマンドが失敗する。データベースサーバに対して -c オプションを使用して <b>dbtran</b> を実行している場合は、-ek オプションではなく、接続パラメータを使用してキーを指定すること。たとえば、次のコマンドでは、データベースサーバサンプルからデータベース enc.db に関するトランザクションログが取得され、その出力が log.sql に保存される。</p> <pre>dbtran -n log.sql -c "ENG=sample;DBF=enc.db;UID=DBA;PWD=sql;DBKEY=mykey"</pre>
-ep	<p>暗号化キーを入力するよう要求する。このオプションを指定すると、暗号化キーを入力するためのウィンドウが表示される。クリアテキストでは暗号化キーを見ることができないようにすることで、高いセキュリティが得られる。-ek または -ep のいずれかを指定する。両方を指定することはできない。正しい暗号化キーを指定しないとコマンドが失敗する。-c オプションを使用して <b>dbtran</b> をデータベースサーバに対して実行する場合は、-ep オプションではなく、接続パラメータを使用してキーを指定する。たとえば、次のコマンドでは、データベースサーバサンプルからデータベース enc.db に関するトランザクションログが取得され、その出力が log.sql に保存される。</p> <pre>dbtran -n log.sql -c "ENG=sample;DBF=enc.db;UID=DBA;PWD=sql;DBKEY=mykey"</pre>
-f	最後のチェックポイント以降に完了したトランザクションのみを出力する。
-g	auditing データベースオプションがオンの場合に、監査情報をトランザクションログに追加する。
-ir <i>offset1,offset2</i>	2つの指定オフセット間の部分的トランザクションログを出力する。

オプション	説明
-is <i>source</i> ,...	次の1つ以上のソース(カンマ区切りのリストで指定)の操作によって変更された、ローに対する操作を出力する。 <ul style="list-style-type: none"> <li>• <b>All</b> – すべてのロー。デフォルト設定。</li> <li>• <b>SQLRemote</b> – SQL Remote を使用して変更されたローだけを含む。省略形の "SR" も使用可能。</li> <li>• <b>RepServer</b> – Replication Agent (LTM) と Replication Server を使用して変更されたローだけを含む。省略形の "RS" も使用可能。</li> <li>• <b>Local</b> – レプリケートされないローだけを含む。</li> </ul>
-it <i>owner.table</i> ,...	カンマ区切りのリストに指定したテーブルに対する操作を出力する。各テーブルを <i>owner.table</i> として指定する。
-j <i>date/time</i>	最新のチェックポイントから指定の日付または時刻までの間に発生したトランザクションのみを変換する。ユーザは、日付、時刻、または日時のいずれかを引用符で囲んだものを引数として指定する。時刻を省略した場合、デフォルトは 00:00。日付を省略した場合、デフォルトは現在の日付になる。日付と時刻の有効なフォーマットは、"YYYY/MMM/DD HH:NN" である。
-k	エラーが検出されたときに部分的な .sql ファイルが消去されないようにする。dbtran の実行中にエラーが検出された場合、その時点まで生成された .sql ファイルは、部分的なファイルが使用されないように、通常は消去される。このオプションを指定すると、破損したトランザクションログからトランザクションをサルベージしようとするときに役立つ。
-m	トランザクションログを格納するディレクトリを指定する。このオプションは -n オプションとともに使用する。
-n <i>filename</i>	dbtran をデータベースサーバに対して実行するときに SQL 文を格納する出力ファイルを指定する。
-o <i>filename</i>	指定したファイルに、出力メッセージを書き込む。
-r	コミットされていないトランザクションを削除する。これはデフォルトの動作である。
-rsu <i>username</i> ,...	ユーザ名のカンマで区切られたリストを指定し、デフォルトの Replication Server ユーザ名を上書きする。デフォルトでは、-is オプションは、デフォルトの Replication Server ユーザ名 dbmaint および sa を想定する。

オプション	説明
-s	<b>UPDATE</b> 文の生成方法を制御する。プライマリーまたはユニークインデックスがテーブルにない場合にこのオプションを使用しないと、 <b>dbtran</b> は、ローが重複しているときに非標準の <b>FIRST</b> キーワードの付いた <b>UPDATE</b> 文を生成する。このオプションを使用した場合は、 <b>FIRST</b> キーワードが省かれ、 <b>SQL</b> 標準との互換性が維持される。
-sr	<b>SQL Remote</b> によるリモートサイトへの操作の分散方法について記述するコメントを生成し、出力ファイルに挿入する。
-t	コマンドファイルにトリガが含まれるかどうかを制御する。デフォルトでは、トリガによって実行されたアクションはコマンドファイルに含まれない。コマンドファイルをデータベースに対して実行したときに、対応するトリガがデータベース内にある場合、トリガはアクションを自動的に実行する。コマンドファイルを実行するデータベース内に、対応するトリガが存在しない場合は、トリガアクションを含める必要がある。
-u <i>userid</i> ,...	指定したユーザのトランザクションログだけが出力されるように制限する。
-x <i>userid</i> ,...	指定したユーザ以外のトランザクションログが出力されるように制限する。
-y	確認メッセージを表示することなく、既存のコマンドファイルを置き換える。-q を指定する場合、-y も指定しないと操作は失敗する。
<i>transaction-log</i>	変換するログファイルを指定する。-c または -m オプションとは一緒に使用できない。
<i>SQL-file</i>	変換した情報を含む出力ファイルを指定する。 <i>transaction-log</i> のみで使用する。

### 使用法

**dbtran** は、次のように実行できます。

- データベースサーバに対して実行する — -c オプションに続けて指定した接続文字列によってデータベースサーバに接続し、-n オプションで指定したファイルに出力を格納します。この方法での実行には、**BACKUP DATABASE** システム権限が必要です。たとえば、次のコマンドでは、**iqdemo** というサーバのログ情報が変換され、**iqdemo.sql** という名前のファイルに出力が格納されます。

```
dbtran -c "eng=iqdemo;dbn=iqdemo;dbf=iqdemo.db;uid=DBA;pwd=sql" -n iqdemo.sql
```

- トランザクションログファイルに対して実行する — トランザクションログファイルに対して直接作用します。ユーザがこの文を実行しないように、トランザクションログファイルを一般のアクセスから保護してください。

```
dbtran iqdemo.log iqdemo.sql
```

**dbtran** では、トランザクションログの開始時のログオフセットが表示されます。これを使用して、生成された複数のログファイルの生成順序を判断できます。

**dbtran-c** は、オンライントランザクションログファイル、およびそれと同じディレクトリにあるすべてのオフライントランザクションログファイルの変換を試みます。ディレクトリに複数のデータベースのトランザクションログファイルが格納されている場合は、エラーが発生する場合があります。このエラーを防ぐには、必ず各ディレクトリに1つのデータベースのみのトランザクションログファイルが格納されるようにします。

1つのトランザクションが複数のトランザクションログにまたがる場合があります。トランザクションログファイルに複数のログにまたがるトランザクションが含まれている場合、1つのトランザクションログファイルの変換を実行すると (**dbtran demo.log** など)、複数のログにまたがるトランザクションが失われることがあります。**dbtran** で完全なトランザクションを生成するためには、ディレクトリ内のトランザクションログファイルで **-c** オプションまたは **-m** オプションを使用してください。

終了コードは、0 (正常) または 0 以外 (エラー) です。

このユーティリティは、**@filename** パラメータを受け付けます。

## dbtran データベース管理ユーティリティ



## dbvalid データベース管理ユーティリティ

dbvalid ユーティリティは、カタログストア内のすべてまたは一部の SQL Anywhere テーブルについてインデックスとキーを検証する検証ユーティリティです。

検証ユーティリティは、テーブル全体をスキャンし、テーブルに定義された各インデックスとキーに含まれる各レコードを検索します。デフォルトでは、検証ユーティリティでは高速チェックオプションが使用されます。

**注意：** dbvalid ユーティリティを使用すると、カタログストアテーブルを簡単に検証できます。IQ メインストアテーブルの検証はできません。IQ メインストアテーブルの検証には、**sp\_iqcheckdb** ストアドプロシージャを使用します。

システムコマンドラインレベルで dbvalid ユーティリティにアクセスできます。これは、dbvalid をバッチファイルまたはコマンドファイルに組み込む場合に役立ちます。

### 構文

```
dbvalid [ options ] [ object-name, ... ]
```

### パラメータ

次の表に、dbvalid ユーティリティの利用可能なオプションを示します。

表 25 : dbvalid のオプション

オプション	説明
<i>object-name</i>	検証するテーブルまたは (-i を使用する場合) インデックスの名前。
-c “keyword=value, ...”	データベース接続パラメータを指定する。ユーザ ID は VALIDATE ANY OBJECT システム権限を保持している必要がある。  たとえば、次のコマンドでは、ユーザ DBA、パスワード sql として iqdemo データベースへの接続が行われ、このデータベースが検証される。  dbvalid -c "uid=DBA;pwd=sql;dbf-c:¥sybase¥IQ-16_0¥demo¥iqdemo.db"
-o filename	出力メッセージをファイルに記録する。

オプション	説明
<b>-f</b>	<p>テーブルをフルチェックで検証する。デフォルトの検証チェックに加え、データチェック (<b>-fd</b>) とインデックスチェック (<b>-fi</b>) を実行する。このオプションは、SQL Anywhere <b>VALIDATE TABLE</b> 文の <b>WITH FULL CHECK</b> オプションに相当する。カタログストアの内容によって、このオプションを指定した場合に検証に長時間かかることがある。</p>
<b>-fd</b>	<p>テーブルをデータチェックで検証する。デフォルトの検証チェックに加えて、LONG BINARY、LONG VARCHAR、TEXT、または IMAGE の各データ型をすべて読み込めるかをチェックする。これらのデータ型のエントリーは、複数のページにわたる場合がある。カタログストア内には、次のデータ型がある。</p> <ul style="list-style-type: none"> <li>• Domain – ユーザ定義データ型。</li> <li>• IMAGE – LONG BINARY. のドメイン</li> <li>• TEXT – LONG VARCHAR. のドメイン</li> </ul> <p>このオプションを指定すると、データベースサーバは、各エントリーで使用されるすべてのページをチェックする。これは、SQL Anywhere <b>VALIDATE TABLE</b> 文での <b>WITH DATA CHECK</b> オプションに相当する。カタログストアの内容によって、このオプションを指定した場合に検証に長時間かかることがある。</p>
<b>-fi</b>	<p>テーブルをインデックスチェックで検証する。デフォルトの検証チェックに加え、テーブルの各インデックスを検証する。これは、SQL Anywhere <b>VALIDATE TABLE</b> 文の <b>WITH INDEX CHECK</b> オプションに相当する。カタログストアの内容によって、このオプションを指定した場合に検証に長時間かかることがある。</p>
<b>-fx</b>	<p>テーブルを高速チェックで検証する。デフォルトチェックとデータチェックに加え、テーブル内のロー数とインデックス内のエントリー数が一致するかをチェックする。これは、SQL Anywhere <b>VALIDATE TABLE</b> 文での <b>WITH EXPRESS CHECK</b> オプションに相当する。このオプションは、各ローのインデックス検索を個別に実行しない。</p>

オプション	説明
<b>-i</b>	<p>各 <i>object-name</i> はインデックスである。テーブルではなく、インデックスを検証する。インデックスで参照されるローが実際にテーブルに存在するかを確認する。外部キーインデックスの場合、<b>-i</b> は、対応するローがプライマリテーブルに存在するかどうかを確認する。<i>index-name</i>、ではなく <i>table-name</i> が指定された場合は、プライマリキーインデックスを検証する。この場合、<b>dbvalid</b> にとって、指定された各 <i>object-name</i> 値はテーブルではなくインデックスを表し、次の形式の名前を持つ。</p> <pre>[ [ owner. ] table-name. ] index-name</pre> <p>インデックスを作成するテーブルの所有者であるか、VALIDATE ANY OBJECT システム権限を保持していることが必要。</p>
<b>-q</b>	クワイエットモードで動作する (出力メッセージの表示なし)。
<b>-s</b>	<p>チェックサムを使用してデータベースページを検証する。データベースページがディスク上で変更されたかどうかを調べるために、チェックサムを使用する。チェックサムを有効にしてデータベースを作成した場合、チェックサムを使用してカタログストアを検証できる。チェックサムによる検証では、カタログストアの各ページがディスクから読み込まれ、そのチェックサムが計算される。計算されたチェックサムがページに保存されているチェックサムと一致しない場合は、ページがディスク上で変更されているので、エラーが返される。カタログストアの不正なページの番号が、サーバメッセージウィンドウに表示される。<b>-s</b> は、<b>-i</b>、<b>-t</b>、またはすべての <b>-f</b> オプションで使用できない。</p>
<b>-t</b>	<i>object-name</i> 値のリストは、テーブルのリストである。これはデフォルトの動作である。

### 使用法

**dbvalid** コマンドラインユーティリティを使用すると、カタログストア内のすべてまたは一部のテーブルについてインデックスおよびキーを検証できます。**dbvalid** は、テーブル全体をスキャンし、各ローが適切なインデックス内に存在することを確認します。これは、各カタログストアテーブルで SQL Anywhere **VALIDATE TABLE** 文を実行するのと同様です。

**注意：** **VALIDATE TABLE** は、SAP Sybase IQ ではサポートされていません。IQ メインストアテーブルに対する類似機能が、**sp\_iqcheckdb** で提供されます。

デフォルトでは、検証ユーティリティでは高速チェックオプションが使用されません。しかし、**-f**、**-fd**、**-fi**、**-fn**、または **-i** を指定した場合、高速チェックオプションは使用されません。

## dbvalid データベース管理ユーティリティ

カタログストアテーブルに不整合がある場合、**dbvalid** はエラーをレポートします。エラーがレポートされた場合は、テーブルのすべてのインデックスとキーを破棄して再作成します。テーブルの外部キーを再作成する必要もあります。

**警告！** テーブルやカタログストア全体を検証できるのは、データベースに変更を加えている接続がないときのみです。データベースの変更中にこれを行うと、データベースに実際には不整合がないにもかかわらず、何らかの不整合があることを示す擬似エラーが報告されることがあります。

プログラム終了コード	説明
0	データベースの検証が正常に終了した。
1	ユーティリティは一般的なエラーで終了した。
2	データベース検証のエラー。
7	接続先のデータベースが見つからない (データベース名の不正)。
8	データベースに接続できない (ユーザ ID / パスワードの不正)。
11	接続先のサーバが見つからない (サーバ名の不正)。
12	データベースを起動するための暗号化キーの不正。

### 例

次のコマンドでは、ユーザ DBA、パスワード sql として iqdemo データベースへの接続が行われ、このデータベースのカタログストアが検証されます。

```
dbvalid -c "uid=DBA;pwd=sql;dbf-c:¥sybase¥IQ-16_0¥demo¥iqdemo.db"
```

## iqdscp 設定ユーティリティ

**iqdscp** ユーティリティは、UNIX のみで動作する Open Client と Open Server の設定ユーティリティで、`interfaces` ファイルやディレクトリサービスを設定するために使用できます。

Open Server 15.5 の『Open Client and Open Server Configuration Guide for UNIX』 > 「Using dscp」を参照してください。

## iqdscp 設定ユーティリティ

## iqdsn データベース管理ユーティリティ

**iqdsn** ユーティリティは、SAP Sybase IQ ODBC データソースの作成、変更、削除、記述、リストを行うために使用できるデータベースユーティリティです。

**iqdsn** ユーティリティは、ODBC アドミネストレータの代わりにプラットフォームを問わずに使用できるユーティリティです。

Windows オペレーティングシステムでは、データソースはレジストリ内に保持されます。UNIX オペレーティングシステムでは、データソースは `.odbc.ini` ファイルに保持されます。**iqdsn** ユーティリティは、バッチ処理に使用します。

### 構文

```
iqdsn [ modifier-options ]
{
| -l dsn
| -d dsn
| -g dsn
| -w dsn [details-options;...]
| -cl }
```

### パラメータ

次の表は、**iqdsn** ユーティリティの利用可能なオプションを示します。

表 26 : **iqdsn** の主要オプション

主要オプション	説明
<b>-l</b>	利用できる SAP Sybase IQ ODBC データソースをリストする。リストのフォーマットは、 <b>-b</b> または <b>-v</b> オプションを使用して変更可能。
<b>-d dsn</b>	指定されたデータソースを削除する。 <b>-y</b> を指定すると、確認メッセージが表示されることなく、既存のデータソースが上書きされる。
<b>-g dsn</b>	指定されたデータソース定義をリストする。 <b>-b</b> オプションまたは <b>-v</b> オプションを使用して、出力の形式を変更できる。
<b>-w dsn</b> [ details-options ]	新しいデータソースを作成するか、同じ名前のデータソースがある場合はそれを上書きする。 <b>-y</b> を指定すると、確認メッセージが表示されることなく、既存のデータソースが上書きされる。
<b>-cl</b>	この便利なオプションを指定すると、 <b>iqdsn</b> ユーティリティでサポートされる接続パラメータがリストされる。

表 27 : iqdsn の変更子オプション

変更子オプション	説明
<b>-b</b>	簡易 (Brief) モード。リストを 1 行の接続文字列の形式で出力する。
<b>-or</b>	<p><b>-c</b> オプションとともに指定した場合に、SQL Anywhere 16 - Oracle ODBC ドライバのデータソースを作成する。</p> <p>次に例を示す。</p> <pre>dbdsn -w MyOracleDSN -or -c Userid=DBA;Password=sql;SID=abcd;ArraySize=500;ProcResults=y</pre> <p><b>-or</b> オプションとともに <b>-cl</b> オプションを指定することで、SQL Anywhere 16 Oracle ODBC ドライバの接続パラメータのリストを取得できる。</p>
<b>-q</b>	クワイエット (Quiet) モード。情報バナーを表示しない。
<b>-v</b>	冗長 (Verbose) モード。リストを複数の行にわたる表の形式で出力する。
<b>-va</b>	詳細 (Verbose All) モード。接続パラメータを <b>-v</b> と同じ形式で出力するが、他の隠しパラメータも出力に含まれる。リモートデータアクセスをサポートした UNIX プラットフォーム上でそのようなデータにアクセスするために必要な ODBC ドライバ修飾子や、サードパーティ製のドライバマネージャの ODBC ドライバ修飾子を表示するには、このオプションを使用する。
<b>-y</b>	確認を求めるメッセージを表示することなく、ファイルを自動的に削除または上書きする。

表 28 : iqdsn の詳細オプション

詳細オプション	説明
<b>-c "keyword= value;..."</b>	接続パラメータを接続文字列として指定する。接続文字列は、セミコロンで区切った複数のパラメータで構成される。
<b>-ecencryption type</b>	クライアントアプリケーションとサーバの間で送信されるパケットを暗号化する。「Encryption 接続パラメータ [ENC]」を参照。
<b>-ofilename</b>	指定した名前のファイルに出力メッセージを書き込む。デフォルトでは、メッセージはコンソールに書き込まれる。「LogFile 接続パラメータ [LOG]」を参照。



詳細オプション	説明
<b>-p</b> <i>size</i>	ネットワーク通信の最大パケットサイズをバイト単位で設定する。500より大きく、16000より小さい値を指定する必要がある。デフォルト設定は1460。「CommBufferSize Connection パラメータ [CBSize]」を参照。
<b>-r</b>	複数のレコードフェッチを無効にする。デフォルトでは、データベースサーバが単純なフェッチ要求を受信すると、アプリケーションは追加のローを要求する。 <b>-r</b> オプションを使用して、この動作を無効にできる。「DisableMultiRowFetch 接続パラメータ [DMRF]」を参照。
<b>-tl</b> <i>seconds</i>	クライアント活性タイムアウト時間。不完全な接続を終了する。値は秒単位。デフォルトでは、サーバの設定が使用される (このデフォルトは120秒)。「LivenessTimeout 接続パラメータ [LTO]」を参照。
<b>-z</b>	起動時に通信リンクに関する診断情報を提供する。
<i>server-name</i>	指定されたデータベースサーバに接続する。最初の40文字のみが有効。

### 使用法

**iqdsn** 変更オプションは、主要オプションの前または後に指定できます。この順序が効果を示すのは、接続パラメータ値を複数回指定した場合のみです。このような場合は、最後に指定した値が使用されます。

### 例

この例では、データソース `newdsn` の定義を書き込みますが、データソースがすでに存在していても、確認メッセージを表示しません。

```
iqdsn -y tcpip -w newdsn -c "uid=DBA;pwd=sql" -v
```

1行に1つのデータソース名が表示される形式ですべての既知のユーザデータソースを出力します。

```
iqdsn -l
```

すべてのデータソースとそれらに関連付けられた接続文字列をリストします。

```
iqdsn -l -b
```

ユーザデータソース `MyDSN` の接続文字列をレポートします。

```
iqdsn -g MyDSN
```

データソース `BadDSN` を削除しますが、最初に `BadDSN` の接続パラメータをリストし、確認メッセージを表示します。

```
iqdsn -d BadDSN -v
```

確認メッセージを表示することなく、データソース `BadDSN` を削除します。

## iqdsn データベース管理ユーティリティ

```
iqdsn -d BadDSN -y
```

NewDSN という名前のデータソースをデータベースサーバ MyServer に作成します。

```
iqdsn -w NewDSN -c "uid=DBA;pwd=sql;eng=bar"
```

NewDSN がすでに存在する場合は、上書きするかどうかを尋ねられます。

次の例では、sample データベースサーバに接続します。サーバ名 **sample** で、前に指定した値 MyServer を上書きします。

```
iqdsn -w NewDSN -c "uid=DBA;pwd=sql;eng=MyServer" sample
```

すべての接続パラメータ名とそれらのエイリアスをリストします。

```
iqdsn -cl
```

## Encryption 接続パラメータ [ENC]

トランスポートレイヤセキュリティまたは単純暗号化を使用してクライアントアプリケーションとデータベースサーバ間で送信されるパケットを暗号化します。

### 使用法

**TLS** の場合は、TCP/IP のみ

**NONE** または **SIMPLE** の場合は、特に制限なし

### 値

*String*

### デフォルト値

**NONE**

Encryption の値を設定しない場合は、暗号化はサーバの設定 (デフォルトは暗号化なし) によって制御されます。

### 説明

ネットワークパケットのセキュリティが確実でない場合は、このパラメータを使用できません。暗号化がパフォーマンスに及ぼす影響はごくわずかです。Encryption (**ENC**) 接続パラメータには次の引数を指定できます。

- **None** : 暗号化されていない通信パケットを受け入れます。
- **Simple** : すべてのプラットフォームと 12.6 より前のバージョンの SAP Sybase IQ でサポートされている単純暗号を使って暗号化された通信パケットを受け入れ

ます。単純暗号化では、サーバ認証、RSA 暗号化、またはその他のトランスポートレイヤセキュリティ機能は提供されません。

- **TLS**：RSA 暗号化技術を使用して暗号化された通信パケットを受け入れます。FIPS 認定の RSA 暗号化の場合は、FIPS=Y を指定します。RSA FIPS は別の認定ライブラリを使用しますが、RSA を指定している SAP Sybase IQ サーバと互換性があります。クライアントとサーバ間の TLS 接続は、サポートされているすべてのプラットフォームでサポートされますが、FIPS=Y は LinuxAMD と Windows (32 および 64 ビット) でのみサポートされます。サーバを認証するために、サーバの証明書の値が、次の引数を使用して指定した値に一致することがソフトウェアによって確認されます。
  - **trusted\_certificate**：クライアントがサーバを認証するために使用する証明書ファイルを指定します。
  - **certificate\_company**：組織フィールドの値を指定します。サーバの値とクライアントの値が一致しなければなりません。
  - **certificate\_unit**：組織単位フィールドの値を指定します。サーバの値とクライアントの値が一致しなければなりません。
  - **certificate\_name**：証明書の通称を指定します。サーバの値とクライアントの値が一致しなければなりません。

---

**警告！** サンプルの証明書はテスト目的でのみ使用してください。サンプルの証明書とそこで使用されるパスワードは SAP Sybase IQ ソフトウェアに付属して広く配布されているため、展開後の運用環境でこの証明書を使用した場合、セキュリティは確保されません。独自の証明書を作成して、システムを保護してください。

---

**CONNECTION\_PROPERTY** システム関数を使用することによって、現在の接続の暗号化設定を取得できます。

*例*

- 次の接続文字列フラグメントでは、RSA 暗号化と信頼された証明書のサンプルを使用して、TCP/IP リンクを通じてデータベースサーバ myeng に接続します。

```
"ENG=myeng; LINKS=tcPIP;
Encryption=(FIPS=N;TRUSTED_CERTIFICATE=iq-16_0/samples/
certificates/rsaroot.crt)"
```

**参照：**

- `-ec iqsrv16` データベースサーバオプション (157 ページ)

## LogFile 接続パラメータ [LOG]

---

クライアントのエラーメッセージおよびデバッグメッセージをファイルに送信します。

*使用法*  
すべての場所

*値*  
String

*説明*  
クライアントのエラーメッセージとデバッグメッセージをファイルに保存するには、**LogFile (LOG)** パラメータを使用します。

ファイル名にパスが含まれる場合、そのパスはクライアントアプリケーションの現在の作業ディレクトリを基準とした相対パスになります。

LogFile (LOG) 接続パラメータは接続に対して固有なので、1つのアプリケーションからさまざまな接続に対して異なる LogFile 引数を設定できます。

*例*  
次のコマンドラインフラグメントでは、この接続に関するクライアントメッセージを、そのクライアントの現在の作業ディレクトリ内にある `error.log` ファイルに送信することを指定しています。

```
...  
LogFile=error.log  
...
```

## CommBufferSize 接続パラメータ [CBSIZE]

---

通信パケットの最大サイズをバイト単位で設定します。キロバイト単位で指定するには K を使用します。

*構文*  
{ **CommBufferSize** | **CBSIZE** }=**size**[ **k** ]

*使用法*  
すべての場所

### 指定可能な値

- **size** – 通信パケットの最大サイズを指定します。デフォルト値はバイト単位ですが、k を使用してキロバイトの単位を指定できます。CommBufferSize の最小値は 500 バイトで、最大値は 65535 バイトです。

### デフォルト値

CommBufferSize 値を設定しないと、サーバ側の設定 (デフォルトは 7300 バイト) によって CommBufferSize が制御されます。

### 備考

ネットワーク上のパケットの最大サイズは、プロトコルスタックによって設定されます。CommBufferSize をネットワークで許可されているサイズより大きく設定すると、通信パケットがネットワークソフトウェアによって分割されます。デフォルトのサイズは、標準 Ethernet TCP/IP の最大パケットサイズ (1460 バイト) の倍数です。

パケットサイズを大きくすると、複数のローのフェッチと長いローのフェッチのパフォーマンスが向上しますが、クライアントとサーバのメモリ使用量が増加します。

クライアント側で CommBufferSize の指定がないと、接続ではサーバのバッファサイズが使用されます。クライアント側で CommBufferSize の指定がある場合、接続ではクライアント側の CommBufferSize 値が使用されます。

-p データベースサーバオプションを使用して CommBufferSize を設定すると、CommBufferSize を指定していないすべてのクライアントで -p データベースサーバオプションで指定されたサイズが使用されます。

### 例

次の文では、バッファサイズを 1460 バイトに設定します。

```
...  
CommBufferSize=1460  
...
```

別の方法として、接続ウィンドウの [ネットワーク] タブの [ Buffer size] テキストボックスに値を入力して、このパラメータを設定することもできます。

## DisableMultiRowFetch 接続パラメータ [DMRF]

---

ネットワーク上での複数レコードのフェッチをオフにします。

*使用法*  
すべての場所

*デフォルト値*  
No

デフォルトでは、データベースサーバが単純なフェッチ要求を受信すると、アプリケーションは追加のローを要求します。このパラメータを ON に設定すると、この動作を無効にできます。

**DisableMultiRowFetch** パラメータを ON に設定した場合の動作は、PREFETCH オプションを OFF に設定した場合と同じです。

*例*  
次の接続文字列フラグメントは、プリフェッチを無効にします。

```
DMRF=Yes
```

## LivenessTimeout 接続パラメータ [LTO]

---

接続が利用されなくなった場合の終了方法を制御します。

*使用法*  
TCP/IP 通信プロトコル上のネットワークサーバ  
すべてのプラットフォーム (非スレッド UNIX アプリケーションを除く)

*値*  
*Integer* (秒単位)

*デフォルト値*  
**120**

**LivenessTimeout** 値を設定しない場合、活性タイムアウトはサーバ設定 (デフォルト値は 120 秒) を使って制御されます。

*説明*  
接続が維持されていることを確認するため、クライアント/サーバの TCP/IP 通信プロトコルを介して、定期的に活性パケットが送信されます。クライアントが活

性要求パケットまたは活性応答パケットを検出しない時間が一定の活性タイムアウト期間を超えると、通信が切断されます。

**LivenessTimeout** 値の 3 分の 1 から 3 分の 2 の期間で接続がパケットを送信しない場合に、活性パケットが送信されます。

通信が切断されると、クライアントマシンは接続先のサーバのアドレスを忘れます。クライアントマシンは、次にサーバに接続するときにアドレスを検索し、そのサーバに対するすべての現在の接続を削除します。

サーバへの接続数が 200 を超えると、サーバは指定された **LivenessTimeout** 値に基づいて、それより大きい **LivenessTimeout** 値を自動的に算出します。それにより、サーバでの大量の接続の処理が効率化されます。

別の方法として、[ODBC 設定] ダイアログボックスの [ネットワーク] タブの [LivenessTimeout] テキストボックスに値を入力して、このパラメータを設定することもできます。

#### 例

次の例は、Liveness Timeout 値を 60 秒に設定しています。

```
LTO=60
```





# iqheader データベース管理ユーティリティ

**iqheader** ユーティリティは、ディスクの使用状況を分析したり、マルチプレック スクエリサーバを設定したりするために、どのサーバ (存在する場合) が特定のデ バイス、ファイル、または LUN (Logical Unit Number) を DB 領域として使用してい るのかを判別する、DB 領域ヘッダユーティリティです。

**iqheader** ユーティリティは、SAP Sybase IQ サーバによって現在使用されているデ バイスかどうかにかかわらず、任意のデバイスの設定をレポートします。

このユーザインタフェースは、**iqheader** (Windows では **iqheader.exe**) と呼ばれるス タンドアロンのコンソールアプリケーションです。iqheader ツールはデバイスに SAP Sybase IQ DB 領域ヘッダがあるかどうかを探し、ヘッダ情報をユーザ可読の フォーマットでレポートします。

---

**注意：** LUN は Logical Unit Number の略であり、ホストがディスクアレイ内の各 ディスクドライブ上のデータにアドレッシングしアクセスできるよう SCSI デバイ スを特定するために使用されます。

---

## 構文

```
iqheader [ dbspace_path ]
```

## パラメータ

**iqheader** アプリケーションは 1 つのパラメータを取ります。これは、チェック対 象となるデバイスです。

## 使用法

- **iqheader の使用法** – パラメータなしで呼び出された場合、使用法のまとめが報 告され、0 以外のステータスが返されます。

```
>iqheader  
Usage: iqheader [dbspace_path]
```

- **iqheader エラー** – 指定した対象が DB 領域でない場合、エラーメッセージがレ ポートされ、0 以外のステータスが返されます。

```
>iqheader /dev/null  
Not an IQ file: Error 0
```

- **オペレーティングシステムエラー** – 指定された対象が読み取れないか、または オペレーティングシステムから返されたエラーのためにファイル操作ができな かった場合、ネイティブのオペレーティングシステムエラーが表示され、0 以 外のステータスが返されます。

## iqheader データベース管理ユーティリティ

```
>iqheader /dev/rdisk/c1t32d0s1 <
Open Failed: No such file or directory

>iqheader /dev/rdisk/c1t3d0s1 <
Open Failed: Permission denied
```

- **iqheader 出力** – 有効な DB 領域を指定した場合は、**iqheader** によってその DB 領域の設定がコンソールに出力され、終了ステータスとして 0 が返されます。

表 29 : iqheader 出力

フィールド	説明
File Name	ファイルの名前
Full Path	シンボリックリンク解析後のフルパス
Version	DB 領域ファイルフォーマットのバージョン
File ID	各 DB 領域に割り当てられたユニークな番号
Create Time	DB 領域が作成された時刻
RW Mode	現在の読み書きモード : RW、RO、RW、N/A (Upgraded)
Last RW Mode	前回の DB 領域モード
Size (MB)	DB 領域のサイズ (メガバイト単位)
Reserve (MB)	DB 領域の予約サイズ (メガバイト単位)
Block Size	ブロックのサイズ (バイト単位)
Page Size	ページのサイズ (バイト単位)
First Block	DB 領域にマッピングされる最初のブロック番号
Block Count	実際のディスクブロックにマッピングされるブロックの数
Reserve Blocks	この DB 領域に追加される可能性があるブロックの数
Last Real Block	実際のディスクブロックにマッピングされる最後のブロック番号
Last Mapped Block	DB 領域にマッピングされる最後のブロック番号
OFlag	オンラインステータス (YES/NO)
Create ID	DB 領域が作成されたコミット ID
Alter ID	DB 領域が変更された直近のコミット ID
DBID1	第 1 データベース ID のロケーション

フィールド	説明
DBID2	第2データベースIDのロケーション
DBSpace ID	各DB領域に割り当てられたユニークな識別子
_NextFLAllocLowerBank	DB領域にあらかじめ割り付けた領域の下限
_NextFLAllocUpperBank	DB領域にあらかじめ割り付けた領域の上限
Pre-alter commit ID	システムテーブル ISYSDBFILE と ISYSIQDBSPACE 内の Commit_id
_ReqNumFreeList-Blocks	タイプ F (フリーリストのブロック) のブロックの数

### 例

次に、iqheader の出力の例を示します。

```
File Name: file1.iq
Full Path: /dev/dsk/file1.iq
DBFile Header Info
Version: 2
File ID: 16395
Create Time: 2008-06-02 21:57:00
RW Mode: RO
Last RW Mode: RW
Size (MB): 20
Reserve (MB): 20
Block Size: 8192
Page Size: 131072
First Block: 9408960
Block Count: 2560
Reserve Blocks: 2560
Last Real Block: 9411519
Last Mapped Block: 10454399
OFlags: 1
Create ID: 6905
Alter ID: 6964
DBID1: 0
DBID2: 0
DBSpace ID: 16395
  _NextFLAllocLowerBank: 0
  _NextFLAllocUpperBank: 0
Pre-alter commit ID: 6925
Dropped: NO
```



## iqinit データベース管理ユーティリティ

**iqinit** ユーティリティは、データベースを作成するユーティリティの実行パーミッションをユーザに付与するデータベースを起動します。ユーザは、コンピュータとファイルシステムにアクセスできる必要があります。

**iqinit** は、SQL Anywhere **dbinit** ユーティリティの SAP Sybase IQ バージョンです。**iqinit** によって、データベースを起動することなく、コマンドラインから SAP Sybase IQ データベースまたは SQL Anywhere データベースを作成できます。

- **iqinit** コマンドラインパラメータが指定されていない場合、**iqinit** は SQL Anywhere データベースを作成します。
- **-iqpath** コマンドラインパラメータが指定されていない場合、**iqinit** は SAP Sybase IQ データベースを作成します。

### 構文

```
iqinit [ options ] new-database-file
```

### パラメータ

次の表に、**iqinit** ユーティリティの利用可能なオプションを示します。

表 30 : **iqinit** のオプション

オプション	説明
<i>@data</i>	指定した環境変数または設定ファイルからオプションを読み込む。設定ファイル内のパスワードなどの情報を保護する場合は、File Hiding ユーティリティを使用して、設定ファイルの内容を難読化できる。
<b>-a</b>	CHAR データ型または NCHAR データ型で Unicode 照合アルゴリズム (UCA) を使用している場合 ( <b>-z</b> および <b>-zn</b> を参照)、文字列比較の際に文字のアクセント記号の違いを考慮する (たとえば、e は é より小さいと見なす)。UCA 照合を使用して作成された日本語のデータベースを除き、デフォルトではアクセント記号は無視される (e は é と等しいと見なされる)。すべての基本文字 (アクセントと大文字/小文字の区別を取り除いた文字) が等しい場合は、アクセントが左から右へ比較される。  日本語のデータベースを作成する場合の UCA 照合では、デフォルトでアクセント記号が区別される。つまり、アクセント記号が考慮される。

オプション	説明
-af	<p>CHAR データ型または NCHAR データ型で UCA を使用している場合(-z および -zn を参照)、文字列比較の際に文字のアクセント記号の違いを考慮する(たとえば、e は é より小さいと見なす)。デフォルトではアクセント記号は無視される(e は é と等しいと見なされる)。すべての基本文字(アクセント記号を取り除いた文字)が等しい場合は、フランス語の規則に従ってアクセント記号が右から左へ比較される。</p>
-b	<p>データベースに空白をパディングする。</p> <p>データベースサーバは、すべての文字列を、可変長であり VARCHAR ドメインを使用して格納されている文字列として比較する。これには、固定長の CHAR カラムまたは NCHAR カラムの文字列比較も含まれる。また、データベースへの値の格納時に、SQL Anywhere は値の末尾への空白のパディングやこの削除は行わない。</p> <p>デフォルトでは、データベースサーバは空白を意味のある文字として扱う。したがって、値 'a' (文字 'a' と、後続の 1 つの空白) は、単一文字の文字列 'a' と等価ではない。不等号比較の照合でも、空白は、他の文字と同じように扱われる。</p> <p>空白のパディングが有効である場合 (iqinit -b オプション)、文字列比較のセマンティックは ANSI/ISO SQL 標準にさらに近づく。空白のパディングが有効であると、SQL Anywhere はどのような比較でも末尾の空白を無視する。</p> <p>上に挙げた例では、空白がパディングされたデータベースで 'a' を 'a' に対して等号比較すると、TRUE が返される。空白がパディングされたデータベースでは、固定長文字列がアプリケーションによってフェッチされると、その文字列に空白がパディングされる。このような文字の割り当てが行われたときにアプリケーションが文字列のトランケーション警告を受け取るかどうかは、ansi_blanks 接続オプションによって制御される。</p>
-c	<p>比較や文字列操作をするときに、すべての値で大文字と小文字を区別する。大文字と小文字を区別するデータベースであっても、データベースの識別子については大文字と小文字は区別されない。</p> <p>UCA 照合を使用して作成された日本語のデータベースを除き、デフォルトではすべての比較において大文字と小文字が区別される。日本語のデータベースを作成する場合の UCA 照合では、デフォルトで大文字と小文字が区別される。</p> <p>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る。</p>

オプション	説明
<b>-dba</b> [ <i>DBA-user</i> ][ <i>pwd</i> ]	<p>DBA ユーザ ID とパスワードを指定する。データベースの DBA ユーザに新しい名前を指定すると、ユーザ DBA としてデータベースに接続することはできなくなる。DBA データベースユーザに別のパスワードを指定することもできる。パスワードを指定しないと、デフォルトのパスワード sql が使用される。このオプションを指定しないと、デフォルトのユーザ ID DBA とパスワード sql が作成される。</p> <p>次のコマンドでは、DBA ユーザ名を testuser、パスワードをデフォルトパスワード sql として、データベースが作成される。</p> <pre>iqinit -dba testuser, mydb.db</pre> <p>次のコマンドでは、ユーザ ID DBA (デフォルト) とパスワード mypwd が使用される。</p> <pre>iqinit -dba ,mypwd mydb.db</pre> <p>次のコマンドでは、DBA ユーザが user1 に変更され、パスワード mypwd が使用される。</p> <pre>iqinit -dba user1,mypwd mydb.db</pre>
<b>-dbs</b> <i>size</i> [ <b>k</b>   <b>m</b>   <b>g</b>   <b>p</b> ]	<p>データベース用の領域を事前に割り付ける。データベース用の領域を事前に割り付けると、データベースがあるドライブの空き領域が不足する危険性を小さくすることができる。また、データベースサイズを拡大する操作は時間を要するため、それが必要となる前にデータベースに保存できるデータの量を増やすことがパフォーマンスの向上につながる。</p> <p>デフォルトのサイズはバイト単位。単位をキロバイト、メガバイト、ギガバイト、またはページで指定するには、それぞれ <b>k</b>、<b>m</b>、<b>g</b>、<b>p</b> を使用する。</p>

オプション	説明
<p><b>-ea</b> <i>algorithm</i></p>	<p>データベース暗号化またはテーブル暗号化 (<b>-et</b>) で使用する暗号化アルゴリズムを指定する。単純暗号の場合は、<b>-ea simple</b> と指定する (<b>-ek</b>、<b>-ep</b> はいずれも指定しないこと)。単純暗号はデータベースの難読化に相当し、データベースファイルが不用意に直接アクセスされた場合にデータが表示されないようにすることだけを目的とする。これにより第三者は、ディスクユーティリティを使用してファイルを表示し、データベースのデータを解読することが困難になる。</p> <p>セキュリティを強化するには、128 ビットの場合は AES、256 ビットの場合は AES256 の強力な暗号化を指定する。FIPS 認定の暗号化を使用するには、128 ビットの場合は AES_FIPS、256 ビットの場合は AES256_FIPS をそれぞれ指定する。強力な暗号化を使用するためには、<b>-ek</b> または <b>-ep</b> オプションも指定する必要がある。</p> <p>暗号で保護されていないデータベースを作成するには、<b>-ea none</b> を指定するか、<b>-ea</b> オプションを指定しない (<b>-et</b>、<b>-ep</b>、<b>-ek</b> オプションも指定しない)。<b>-ea</b> オプションを指定しない場合、デフォルトの動作は次のとおり。</p> <ul style="list-style-type: none"> <li>• <b>-ek</b>、<b>-ep</b>、または <b>-et</b> を指定した場合は、<b>-ea none</b></li> <li>• <b>-ek</b> または <b>-ep</b> を指定した (<b>-et</b> とともに、またはこれを伴わずに) 場合は、<b>-ea AES</b></li> <li>• <b>-et</b> を <b>-ek</b> または <b>-ep</b> なしで使用した場合は、<b>-ea simple</b></li> </ul> <p>アルゴリズム名の大文字と小文字は区別されない。</p> <p>ファイル圧縮ユーティリティを使用する場合、暗号で保護されたデータベースは、暗号で保護されていないデータベースほど圧縮できない。</p>
<p><b>-ek</b> <i>key</i></p>	<p>コマンドに暗号化キーを直接指定することによって、強い暗号で保護されたデータベースを作成するように指定する。<b>-ek</b> オプションは、AES アルゴリズム (<b>-ea</b> オプションによって任意で指定) とともに使用する。<b>-ea</b> オプションを指定しないで <b>-ek</b> オプションを指定した場合、デフォルトで AES が使用される。このオプションを <b>-et</b> とともに指定した場合、データベースは暗号化されない。この場合、テーブル暗号化が有効になる。</p>



オプション	説明
<b>-ep</b>	<p>ウィンドウに暗号化キーを入力することによって、強い暗号で保護されたデータベースを作成するように指定する。暗号化キーがクリアテキストで表示されないようにすることで、高いセキュリティが得られる。暗号化キーは、正確に入力されたことを確認するために2回入力する必要がある。キーが一致しない場合は、初期化は失敗する。このオプションを <b>-et</b> とともに指定した場合、データベースは暗号化されない。この場合、テーブル暗号化が有効になる。</p>
<b>-et</b>	<p><b>-ea</b> オプションで指定した暗号化アルゴリズム(およびキー)を使用したテーブル暗号化を有効にする。このオプションは、データベース全体を暗号化しないで、暗号化されたテーブルを作成する場合に使用する。<b>-et</b> を <b>-ek</b> または <b>-ep</b> とともに指定し、<b>-ea</b> は指定しない場合、デフォルトで AES アルゴリズムが使用される。<b>-et</b> のみを指定した場合は、単純暗号が使用される。</p> <p>テーブル暗号化を有効にただけで、テーブルが暗号化されるわけではない。データベースを作成した後で、テーブルを個別に暗号化する必要がある。</p> <p>テーブルの暗号化が有効な場合、暗号化されたテーブルのテーブルページ、関連するインデックスページ、テンポラリファイルのページ、暗号化されたテーブルのトランザクションを含むトランザクションログページが暗号化される。</p> <p>データベース new.db を作成する次の例では、キー abc と暗号化アルゴリズム AES_FIPS を使用したテーブルの強力な暗号化を有効に設定している。</p> <pre>iqinit -et -ek abc -ea AES_FIPS new.db</pre>
<b>-i</b>	<p>jConnect™ for JDBC™ システムオブジェクトをデータから除外する。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するには、jConnect カタログサポートが必要(デフォルトでインストールされる)。このオプションを指定した場合でも、システム情報にアクセスしないかぎり、JDBC を使用できる。jConnect サポートは、<b>ALTER DATABASE</b> 文を使用して後から追加できる。</p>
<b>-iqblksize</b>	I/O 転送ブロックサイズ (バイト単位)。
<b>-iqmsgpath</b>	SAP Sybase IQ メッセージトレースファイルを含むセグメントのパス名。
<b>-iqpath</b>	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。

## iqinit データベース管理ユーティリティ

オプション	説明
<b>-iqpgsize</b>	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)。
<b>-iqreservesize</b>	IQ メインストア用に確保する領域のサイズ (MB 単位)。
<b>-iqsize</b>	<b>-iqpath</b> で指定したローパーティションまたは OS ファイルのサイズ (MB 単位)。
<b>-iqtmppath</b>	テンポラリセグメントファイルのパス名。
<b>-iqtmpreserve-size</b>	IQ テンポラリストア用に確保する領域のサイズ (MB 単位)。
<b>-iqtmpsize</b>	<b>-iqtmppath</b> で指定したローパーティションまたは OS ファイルのサイズ (MB 単位)。
<b>-k</b>	SYSCOLUMNS ビューと SYSINDEXES ビューを作成しない。デフォルトでは、データベース作成機能は、Watcom SQL (このソフトウェアのバージョン 4 以前) で使用可能なシステムテーブルとの互換性を保つために、ビュー SYS.SYSCOLUMNS と SYS.SYSINDEXES を生成する。これらのビューは、Sybase Adaptive Server Enterprise の互換ビュー dbo.syscolumns および dbo.sysindexes と競合する。
<b>-l</b>	使用可能な照合順をリストした後、停止する。データベースは作成されない。照合順を指定するには、 <b>-z</b> オプションを使用する。

オプション	説明
-le	<p>使用可能な文字セットエンコードをリストした後、停止する。データベースは作成されない。各文字セットエンコードは、1つ以上のラベルによって識別される。これらの文字列は、エンコードの識別に使用できる。表示される各テキスト行に、エンコードラベルと、エンコードの識別に使用できる代替ラベルがリストされる。これらのラベルは、いくつかの共通カテゴリのいずれかに該当する。カテゴリには、SA (SQL Anywhere ラベル)、IANA (Internet Assigned Numbers Authority)、MIME (Multipurpose Internet Mail Extensions)、ICU (International Components for Unicode)、JAVA、ASE (Adaptive Server Enterprise) がある。</p> <p>代替ラベルが含まれる文字セットエンコードリストを表示するには、-le+ オプションを指定する。</p> <p>iqinit ユーティリティによる文字セットエンコードのレポート時には、必ず、ラベルのデータベースサーババージョンもレポートされる。たとえば、次のコマンドでは、CHAR データ型の文字セットエンコード windows-1250 がレポートされる。</p> <pre>iqinit -ze cp1250 -z uca test.db</pre>
-m <i>filename</i>	<p>トランザクションログミラーを作成する。トランザクションログミラーは、トランザクションログの同一コピーであり、通常、ユーザデータの保護を強化するために、別のデバイス上で管理される。デフォルトでは、データベースサーバはトランザクションログミラーを使用しない。</p>
-n	<p>トランザクションログを生成しないデータベースを作成する。トランザクションログを生成しないデータベースを作成すると、ディスク領域が節約されるが、コミットごとにチェックポイントが発生するため、パフォーマンスが低下する可能性がある。また、データベースが破損した場合、トランザクションログがないとデータを回復できない。データレプリケーションにはトランザクションログが必要であり、このログによって、メディアまたはシステムの障害発生時におけるデータベース情報のセキュリティが強化される。</p>
-o <i>filename</i>	<p>指定したファイルに出力メッセージを書き込む。</p>

オプション	説明
-p <i>page-size</i>	<p>データベースのページサイズを指定する。データベースのページサイズには、2048、4096、8192、16384、32768 バイトのいずれかを指定できる。デフォルトは 4096 バイト。キロバイト単位で指定するには k を使用する (例: -p 4k)。</p> <p>大規模なデータベースでは、ページサイズを大きくしたほうが有利な場合がある。たとえばテーブルのスキャンでは、一度にページ全体が読み込まれるため、通常、必要な I/O 操作の回数が少なくなる。ただし、ページサイズが大きいと、より多くのメモリが必要となる。ページサイズを選択する際には、パフォーマンステスト (およびテスト全般) の実行が強く推奨される。テストで満足できる結果を得られた最小のページサイズを選択するようにする。ほとんどのアプリケーションでは、16 KB または 32 KB のページサイズは推奨されない。常に十分なデータベースサーバキャッシュが利用可能であることを保証でき、かつメモリとディスク領域のパフォーマンス特性に対するトレードオフを調査済みである場合以外は、運用システムで 16 KB または 32 KB のページサイズは使用すべきでない。多数のデータベースを同じサーバで起動する場合は、適切なページサイズを選択すること。</p>
-pd	<p>このオプションを指定しないと (デフォルト)、権限付きのすべてのシステムプロシージャが invoker の権限で実行される。指定した場合、16.0 より前の権限付きシステムプロシージャは、definer の権限で実行されるに対し、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。</p>
-q	<p>クワイエットモードで実行する (メッセージは表示されない)。</p>

オプション	説明
-s	<p>グローバルチェックサムを追加する (データベースページごとにチェックサムが追加される)。デフォルトでは、このオプションはオン。</p> <p>チェックサムを使用して、データベースページがディスク上で変更されたかどうか判断される。グローバルチェックサムを有効にしてデータベースを作成した場合、各ページがディスクに書き込まれる直前にそのページのチェックサムが計算される。そのページが次にディスクから読み出されるときには、ページのチェックサムが再計算されて、ページに保存されているチェックサムと比較される。チェックサムが異なる場合は、ディスク上のページが変更されているか、破損しており、エラーが発生する。-s オプションの値を指定したかどうかにかかわらず、重要なデータベースページにはデータベースサーバによって必ずチェックサムが追加される。</p> <p>リムーバブルドライブなどのストレージデバイスで実行されるデータベースでは、データベースの破損を速やかに検出できるように、自動的にチェックサムが有効になる。</p> <p>グローバルチェックサムを無効にしてデータベースを作成した場合でも、-wc オプションまたは <b>START DATABASE</b> 文を使用して、ページの書き込み時にチェックサムをページに追記できる。</p>
-t <i>transaction-log-name</i>	<p>トランザクションログファイルの名前を指定する。トランザクションログは、使用アプリケーションを問わず、あらゆるユーザによるすべての変更がデータベースサーバによって記録されるファイルである。トランザクションログはバックアップ/リカバリとデータレプリケーションで重要な役割を果たす。トランザクションログのファイル名にパスがない場合、トランザクションログはデータベースファイルと同じディレクトリに保存される。-t または -n を指定しないで iqinit を実行した場合は、データベースファイルと同じファイル名で拡張子が .log のトランザクションログが作成される。</p>

オプション	説明
<b>-z coll[ collation-tailoring-string</b>	<p>データベースの照合順を指定する。照合順は、文字データ型 (CHAR、VARCHAR、LONG VARCHAR) のソートと比較に使用される。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る。照合は慎重に選択することが重要である。データベースの作成が完了した後で照合を変更するには、データベースをアンロードし、再ロードする必要がある。照合が指定されていない場合、データベースサーバは、オペレーティングシステムの言語と文字セットに基づいて照合を選択する。使用可能な照合順を表示する方法については、<b>-i</b> オプションを参照。</p> <p>文字列のソートや比較を詳細に制御することを目的に、照合の適合化オプション (<i>collation-tailoring-string</i>) を指定することもできる。これらのオプションは、keyword=value ペアの形式で指定する。各ペア全体をカッコで囲んで、照合名に続けて指定する。例：</p> <p><i>collation-tailoring-string</i> での大文字小文字の区別の設定とアクセント記号の区別の設定は、iqinit の大文字小文字の区別とアクセント記号の区別のオプション (<b>-c</b>、<b>-a</b>、および <b>-af</b>) を上書きする (これらの両方を指定した場合)。</p>
<b>-ze encoding</b>	<p>照合のエンコードを指定する。<b>-z</b> で指定するほとんどの照合では、エンコード (文字セット) と順次付けの両方が定義されている。そのような照合については、<b>-ze</b> は指定しない。</p> <p><b>-z</b> で指定した照合が Unicode 照合アルゴリズム (UCA) である場合は、<b>-ze</b> で CHAR データ型エンコードとして UTF-8 またはシングルバイトエンコードを指定できる。</p> <p>デフォルトでは、データベースサーバは UTF-8 を使用する。<b>-ze</b> によってロケール固有のエンコードを指定し、比較と順序付けに UCA を利用できる。</p>

オプション	説明
<b>-zn coll[ collation-tailoring-string</b>	<p>各国の文字データ型 (NCHAR、NVARCHAR、LONG NVARCHAR) のソートと比較に使用する照合順を指定する。照合によって、使用する UTF-8 エンコード (文字セット) に文字の順序付けに関する情報が提供される。このオプションの値は、UCA (デフォルト) か、またはエンコードが 0x7E を超えるすべての文字のバイナリ順序を規定する UTF8BIN。dbicu16 と dbicudt16 の DLL がインストールされていない場合、UTF8BIN がデフォルトの NCHAR 照合になる。</p> <p>文字列のソートや比較を詳細に制御することを目的に、照合の適合理化オプション (<i>collation-tailoring-string</i>) を指定することもできる。これらのオプションは、keyword=value ペアの形式で指定する。各ペア全体をカッコで囲んで、照合名に続けて指定する。例：</p> <pre>iqinit -c -zn UCA(case=LowerFirst) sens.db</pre> <p><i>collation-tailoring-string</i> での大文字小文字の区別の設定とアクセント記号の区別の設定は、iqinit の大文字小文字の区別とアクセント記号の区別のオプション (-c、-a、および -af) を上書きする (これらの両方を指定した場合)。</p>

**例**

次のコマンドでは、bar.iq という名前の SAP Sybase IQ データベースが作成されます。

```
$ iqinit -iqpath bar.iq -iqsize 20M
-iqpgsize 2048
-iqreservesize 10M bar.db
SQL Anywhere Initialization Utility Version 16.0.0.5530
Debug
CHAR collation sequence:
ISO BINENG (CaseSensitivity=Respect)
CHAR character set encoding: ISO_8859-1:1987
NCHAR collation sequence:
UCA (CaseSensitivity=UpperFirst;
AccentSensitivity=Respect;
PunctuationSensitivity=Primary)
NCHAR character set encoding: UTF-8
Creating system tables
Creating system views
Setting option values
Database "bar.db" created successfully
```





## iqocscfg 設定ユーティリティ

**iqocscfg** ユーティリティは、Windows のみで動作する Open Client と Open Server の設定ユーティリティで、環境変数、ディレクトリドライバ、セキュリティドライバを設定するために使用できます。

Open Server 15.5 の Software Developer's Kit 15.5 の『Open Client and Open Server Configuration Guide for Microsoft Windows』 > 「Using ocscfg」を参照してください。



# isql Interactive SQL ユーティリティ

**isql** ユーティリティは、Adaptive Server Enterprise Open Client API を使用する、コマンドライン Interactive SQL ユーティリティです。

**isql** では、ユーザ定義のデータベースオプションは作成できません。独自のデータベースオプションを追加する必要がある場合は、代わりに **dbisql** Interactive SQL ユーティリティを使用してください。

## 構文

```
isql
[-b] [-e] [-F] [-p] [-n] [-v] [-W] [-X] [-Y] [-Q]
  [-a display_charset]
  [-A packet_size]
  [-c cmdend]
  [-D database]
  [-E editor]
  [-h header]
  [-H hostname]
  [-i inputfile]
  [-I interfaces_file]
  [-J client_charset]
  [-K keytab_file]
  [-l login_timeout]
  [-m errorlevel]
  [-o outputfile]
  [-P password]
  [-R remote_server_principal]
  [-s colseparator]
  [-S server_name]
  [-t timeout]
  -U username
  [-V [security_options]]
  [-w columnwidth]
  [-z locale_name]
  [-Z security_mechanism]
  [--conceal]
  [--URP]
```

## パラメータ

次の表に、**isql** ユーティリティの利用可能なオプションを示します。

表 31 : isql のオプション

オプション	説明
-b	テーブルヘッダの出力の表示を無効にする。
-e	入力をエコーする。
-F	FIPS フラガを有効にする。-F パラメータを指定すると、標準以外の SQL コマンドが検出されたときにサーバからメッセージが返される。このオプションを指定しても、SQL 拡張機能は無効にならない。ANSI 以外の SQL コマンドを発行すると、処理が完了する。
-p	パフォーマンス統計を表示する。
-n	-e と組み合わせた使用時に、出力ファイル内のエコーされた入力行から番号とプロンプト記号 (>) を削除する。
-v	isql のバージョン番号と著作権メッセージを印刷し、終了する。
-X	クライアント側のパスワード暗号化によって、サーバへのログイン接続を開始する。-X は、プレーンテキストパスワードによる再接続を行わない、拡張パスワード暗号化接続とパスワード暗号化接続の両方を有効にする。isql (クライアント) は、パスワードの暗号化が必要であることをサーバに指定する。サーバは暗号化キーを返送する。isql はこのキーを使用してパスワードを暗号化する。暗号化されたパスワードが到着すると、サーバはこのキーを使用してパスワードを認証する。  isql がクラッシュした場合は、パスワードを含むコアファイルが作成される。暗号化オプションを使用しなかった場合は、ファイルにパスワードがプレーンテキストで表示される。暗号化オプションを使用した場合、パスワードは判読できない。
-W	拡張パスワード暗号化とパスワード暗号化の両方のネゴシエーションを無効にする。
-Y	連鎖トランザクションを使用するように、Adaptive Server に指示する。
-Q	クライアントに failover プロパティを提供する。

オプション	説明
<b>-a</b> <i>display_charset</i>	<p>文字セットが <b>isql</b> を実行しているマシンの文字セットと異なるターミナルから <b>isql</b> を実行する。<b>-a</b> を <b>-J</b> とともに使用することによって、変換が必要となる文字セット変換ファイル(.x1t ファイル)を指定する。<b>-a</b> を <b>-J</b> なしで使用するのは、クライアントの文字セットがデフォルトの文字セットと同じである場合のみ。</p> <hr/> <p><b>注意：</b> ascii_7 文字セットはすべての文字セットと互換性があります。Adaptive Server の文字セットまたはクライアントの文字セットのどちらかが ascii_7 に設定されている場合は、クライアントとサーバ間で任意の7ビット ASCII 文字を無変更の状態ですべて渡すことができます。その他の文字では変換エラーが発生します。</p>
<b>-A</b> <i>packet_size</i>	<p>この <b>isql</b> セッションで使用するネットワークパケットサイズを指定する。たとえば、次のように指定した場合、この <b>isql</b> セッションのパケットサイズが 4096 バイトに設定される。</p> <pre>isql -A 4096</pre> <ul style="list-style-type: none"> <li>• ネットワークパケットサイズをチェックするには、次のように入力する。  <pre>SELECT * FROM sysprocesses</pre> </li> <li>• network_pktsz という見出しの下に、値が表示される。</li> <li>• size は、<b>default network packet size</b> パラメータ値と <b>maximum network packet size configuration</b> パラメータ値の間の値である必要があり、また、512 の倍数でなければならない。</li> <li>• <b>readtext</b> や <b>writetext</b> など入出力集中型の操作を実行する場合は、デフォルトのパケットサイズよりも大きいパケットサイズを使用する必要がある。</li> <li>• Adaptive Server のパケットサイズを設定または変更しても、リモートプロシージャコールのパケットサイズには影響しない。</li> </ul>
<b>-c</b> <i>cmdend</i>	<p>コマンドターミネータを変更する。デフォルトでは、行に単独で "go" と入力すると、コマンドが終了し、送信される。コマンドターミネータを変更する際には、SQL の予約語および制御文字はいずれも使用しないようにすること。</p>
<b>-D</b> <i>database</i>	<p><b>isql</b> セッションを開始するデータベースを選択する。</p>
<b>-E</b> <i>editor</i>	<p>デフォルトエディタ <b>vi</b> 以外のエディタを指定する。エディタを呼び出すには、<b>isql</b> の行の最初の単語としてエディタ名を入力する。</p>

オプション	説明
<b>-h headers</b>	カラム見出し間に印刷するロー数を指定する。デフォルトでは、クエリ結果セットごとに 1 回だけ見出しが印刷される。
<b>-H hostname</b>	カラム見出し間に印刷するロー数を指定する。デフォルトでは、クエリ結果セットごとに 1 回だけ見出しが印刷される。
<b>-i interfaces_file</b>	Adaptive Server への接続時に検索する interfaces ファイルの名前とロケーションを指定する。 <b>-I</b> を指定しないと、 <b>isql</b> は、SYBASE 環境変数で指定されているディレクトリで interfaces という名前のファイルを検索する。
<b>-J client_charset</b>	<p>クライアントで使用する文字セットを指定する。<b>-J client_charset</b> を指定することによって、クライアントで使用する文字セット <i>client_charset</i> へ、またはこの文字セットからの変換を実行するように Adaptive Server に要求できる。フィルタによって、<i>client_charset</i> と Adaptive Server の文字セットとの間で入力が変換される。</p> <p>引数なしで <b>-J</b> を指定すると、文字セットの変換が NULL に設定される。変換は実行されない。これは、クライアントとサーバで文字セットが同じである場合に使用する。</p> <p><b>-J</b> を省略すると、文字セットがプラットフォームのデフォルトに設定される。デフォルトは、クライアントが使用している文字セットでない場合もある。文字セットおよび関連するフラグの詳細は、『Adaptive Server Enterprise システム管理ガイド 第1巻』の「クライアント/サーバの文字セット変換の設定」を参照。</p>
<b>-K keytab_file</b>	DCE で認証に使用する keytab ファイルのパスを指定する。
<b>-l login_timeout</b>	Adaptive Server への接続時に許容する最大タイムアウト値を指定する。デフォルトは 60 秒。この値は、ログイン試行に対するサーバの応答を <b>isql</b> が待つ時間にのみ影響する。コマンド処理のタイムアウト時間を指定するには、 <b>-t timeout</b> パラメータを使用する。
<b>-m errorlevel</b>	エラーメッセージの表示をカスタマイズする。指定した重大度レベル以上のレベルのエラーについては、メッセージ番号、ステータス、およびエラーレベルが表示される。エラーテキストは表示されない。指定したレベル未満のエラーレベルについては、何も表示されない。
<b>-o outputfile</b>	<b>isql</b> からの出力を格納するオペレーティングシステムファイルの名前を指定する。このパラメータを <b>-o outputfile</b> と指定することは、 <b>&gt; outputfile</b> と指定するのと同じようなものである。

オプション	説明
<b>-P password</b>	Adaptive Server のパスワードを指定する。-P フラグを指定しないと、isql によってパスワードを入力するよう要求される。パスワードが NULL の場合は、パスワードを伴わない -P フラグを使用する。
<b>-R remote_server_principal</b>	セキュリティメカニズムに定義されているサーバのプリンシパル名を指定する。デフォルトでは、サーバのプリンシパル名はサーバのネットワーク名 (-S パラメータまたは DSQUERY 環境変数で指定) と一致する。-R パラメータは、サーバのプリンシパル名とネットワーク名が同じでない場合に使用する。
<b>-s colseparator</b>	カラムの区切り文字をリセットする。この区切り文字は、デフォルトでは空白。オペレーティングシステムに対して特別な意味を持つ文字 (" ", ";", "&", "<", ">" など) を使用する場合は、引用符で囲むか、前に円記号を付ける。
<b>-S server_name</b>	接続先の Adaptive Server 名を指定する。isql はこの名前を interfaces ファイルで検索する。-S を引数なしで指定すると、isql は SYBASE という名前のサーバを検索する。-S を指定しないと、isql は DSQUERY 環境変数で指定されているサーバを検索する。
<b>-t timeout</b>	SQL コマンドがタイムアウトになるまでの秒数を指定する。タイムアウトを指定しないと、コマンドは無期限で実行される。これは、isql 内から発行されるコマンドに影響する。接続時には影響しない。isql へのログインのデフォルトのタイムアウトは 60 秒。
<b>-U username</b>	ログイン名を指定する。ログイン名では、大文字と小文字が区別される。
<b>--URP string</b>	SAP Sybase IQ サーバのログインリダイレクション文字列を指定できる。次の例では、論理サーバ、データベース、ノードタイプ、およびリダイレクション文字列を指定している。 <pre>isql -Usa -P --URP "LS=salogsrv;dbname=dbone;node=writer;redirect=no"</pre> または Adaptive Server にアクセスするクライアントの汎用リモートパスワード <i>remotepassword</i> の設定を有効にする。アプリケーションによって、汎用リモートパスワードが設定される。たとえば、 <b>ctlib</b> では <code>ct_remote_pwd()</code> が使用され、 <b>jConnect</b> では <code>setRemotePassword</code> メソッドが使用される。

オプション	説明
<b>-v</b> <i>security_options</i>	<p>ネットワークベースのユーザ認証を指定する。このオプションでは、ユーザは、ユーティリティの実行前にネットワークセキュリティシステムにログインする必要がある。この場合、ユーザは <b>-U</b> オプションでネットワークユーザ名を指定する必要がある。<b>-P</b> オプションで指定したパスワードは無視される。</p> <p><b>-v</b> の後に、キー文字オプションの <i>security_options</i> 文字列を指定することによって、追加のセキュリティサービスを有効化できる。このようなキー文字には次のものがある。</p> <ul style="list-style-type: none"> <li>• <b>-c</b> - データ機密性サービスを有効にする。</li> <li>• <b>-i</b> - データ整合性サービスを有効にする。</li> <li>• <b>-m</b> - 接続確立のための相互認証を有効にする。</li> <li>• <b>-o</b> - データオリジンスタンプングサービスを有効にする。</li> <li>• <b>-q</b> - 順序不整合の検出を有効にする。</li> <li>• <b>-r</b> - データリプレイの検出を有効にする。</li> </ul>
<b>-w</b> <i>columnwidth</i>	<p>出力の画面幅を設定する。デフォルトは 80 文字。出力行がその最大画面幅に達すると、改行が行われる。</p>
<b>-z</b> <i>locale_name</i>	<p>isql のプロンプトとメッセージを表示する代替言語の正式な名前を指定する。<b>-z</b> を指定しないと、<b>isql</b> はサーバのデフォルト言語を使用する。Adaptive Server への言語の追加は、インストール時、またはその後に <b>langinstall</b> ユーティリティ (Windows では <b>langinst</b>) または <b>sp_addlanguage</b> ストアドプロシージャを使用して実行できる。</p>
<b>-Z</b> <i>security_mechanism</i>	<p>接続で使用するセキュリティメカニズムの名前を指定する。</p> <p>セキュリティメカニズム名は、Sybase インストールディレクトリ内の <i>ini</i> サブディレクトリにある <i>libtcl.cfg</i> 設定ファイルに定義されている。<i>security_mechanism_name</i> を指定しないと、デフォルトのメカニズムが使用される。セキュリティメカニズム名の詳細については、『Open Client and Open Server Configuration Guide』の <i>libtcl.cfg</i> ファイルに関する説明を参照。</p>



オプション	説明
<b>--conceal</b>	<p>isql セッション中、入力内容を非表示にする。<b>--conceal</b> オプションは、パスワードなどの機密情報を入力するときに役立つ。<b>--conceal</b> オプションの構文は次のとおり。</p> <pre>isql --conceal [':?'   'wildcard']</pre> <p>32 バイトの変数 <i>wildcard</i> の最大長は 80 文字。</p> <p><b>注意：</b> バッチモードでは、<b>--conceal</b> は単に無視されます。通知は行われません。</p>

### 使用法

スレッドドライバを使用している場合、**isql\_r** にこの構文を使用します。

IBM プラットフォームでスレッドドライバを使用している場合、**isql** にこの構文を使用します。

isql を使用するには、事前に SYBASE 環境変数を現在のバージョンの Adaptive Server のローケーションに設定する必要があります。

次の表に、対話型 **isql** 内で実行できるコマンドを示します。

表 32 : isql 内で実行できるコマンド

コマンド	説明
<b>:r filename</b>	<p>オペレーティングシステムファイルをコマンドバッファに読み込む。</p> <p>ファイル内にはコマンドターミネータを含めないこと。編集が完了したら、対話形式で行にターミネータを単独で入力する。</p>
<b>:R filename</b>	<p>オペレーティングシステムファイルをコマンドバッファに読み込み、表示する。</p> <p>ファイル内にはコマンドターミネータを含めないこと。編集が完了したら、対話形式で行にターミネータを単独で入力する。</p>
<b>use database_name</b>	現在のデータベースを変更する。
<b>!! os_command</b>	オペレーティングシステムコマンドを実行する。行の先頭に配置する。

コマンド	説明
> <b>file_name</b>	T-SQL コマンドの出力を <i>file_name</i> にリダイレクトする。次の例では、サーババージョンが <i>file_name</i> に挿入される。 <pre>select @@version go &gt; file_name</pre>
>> <b>file_name</b>	T-SQL コマンドの出力を <i>file_name</i> に追加する。次の例では、サーババージョンが <i>file_name</i> に追加される。 <pre>select @@version go &gt;&gt; file_name</pre>
<b>command</b>	T-SQL コマンドの出力を外部コマンドにパイピングする。 次の例では、 <b>sp_who</b> によって生成されるリストで "sa" のすべてのインスタンスが検索される。 <pre>sp_who go   grep sa</pre>
<b>vi</b> (UNIX の場合) <b>edit</b> (Windows の場合)	デフォルトエディタを呼び出す。
<b>reset</b>	クエリバッファをクリアする。
<b>quit</b> または <b>exit</b>	<b>isql</b> を終了する。

ログイン後または **use database** コマンドの発行後に、5701 (「データベースの変更」) サーバメッセージは表示されなくなりました。

以前のバージョンの **isql** とは、エラーメッセージのフォーマットが異なります。このようなメッセージの値に基づくルーチンを実行するスクリプトがある場合は、スクリプトを書き直す必要がある場合があります。

**-x** パラメータを指定した場合は、サーバの機能に応じて、パスワード有効接続が処理されます。

- サーバが拡張パスワード暗号化とパスワード暗号化の両方を処理できる場合は、拡張パスワード暗号化ネゴシエーションが使用されます。
- サーバがパスワード暗号化のみを処理できる場合は、パスワード暗号化ネゴシエーションが使用されます。
- サーバがパスワード暗号化も拡張パスワード暗号化も処理できない場合は、最初の接続試行が失敗し、クライアントはプレーンテキストパスワードを使用して再接続を試みます。

**isql** を対話形式で使用するには、オペレーティングシステムプロンプトでコマンド **isql** (および任意のオプションパラメータ) を指定します。**isql** プログラムは SQL

コマンドを受け入れて、そのコマンドを Adaptive Server に送信します。結果がフォーマットされ、標準出力で出力されます。isql を終了するには、quit または exit を使用します。

コマンドを終了するには、行の先頭にデフォルトのコマンドターミネータ go を入力するか、または -c パラメータを使用する場合は別のコマンドターミネータを入力します。コマンドターミネータに続けて整数を入力することによって、コマンドの実行回数を指定できます。たとえば、次のコマンドを 100 回実行する場合は、次のように入力します。

```
select x = 1
go 100
```

結果は、実行の最後に 1 回表示されます。

コマンドラインで 1 つのオプションを何度も入力した場合、isql は最後の値を使用します。たとえば、次のコマンドを入力した場合、-c の最初の値 "." は 2 番目の値 "send" によって上書きされます。

```
isql -c"." -csend
```

この方法で、設定したあらゆるエイリアスを上書きできます。

現在のクエリバッファでエディタを呼び出すには、行の最初の単語としてエディタ名を入力します。呼び出し可能な優先エディタを定義するには、EDITOR 環境変数でそのエディタを指定します。EDITOR が定義されていない場合、デフォルトは UNIX では vi、Windows では edit です。

たとえば、EDITOR 環境変数が "emacs" に設定されている場合に、isql からこのエディタを呼び出すには、行の最初の単語として "emacs" を入力します。

既存のクエリバッファをクリアするには、行に単独で reset と入力します。保留になっている入力のすべてが isql から破棄されます。行の任意の位置で [CTRL] キーを押しながら [C] キーを押すことによって、現在のクエリをキャンセルし、isql プロンプトに戻ることもできます。

isql で実行するクエリを含むオペレーティングシステムファイルを読み込むには、次のように指定します。

```
isql -U alma -P password < input_file
```

このファイルにはコマンドターミネータが含まれていることが必要です。結果がターミナルに表示されます。クエリを含むオペレーティングシステムファイルを読み込んで、結果を別のファイルに送るには、次のように指定します。

```
isql -U alma -P password < input_file > output_file
```

isql では、float または real データ型の小数点以下 6 桁までが表示され、残りは丸められます。

**isql** によって Adaptive Server に送信される Transact-SQL 文にコメントを含めることができます。コメントを開始するには、`/*` を使用します。終了するには、`*/` を使用します。次に例を示します。

```
select au_lname, au_fname
/*retrieve authors' last and first names*/
from authors, titles, titleauthor
where authors.au_id = titleauthor.au_id
and titles.title_id = titleauthor.title_id
/*this is a three-way join that links authors
**to the books they have written.*/
```

**go** コマンドをコメントアウトする場合は、行の先頭に配置しないようにします。たとえば、**go** コマンドをコメントアウトするには、次のように入力します。

```
/*
**go
*/
```

次のように入力しないでください。

```
/* go */
```

**isql** では、ロケールの環境にかかわらず、日付フォーマットの順序は、月、日付、および年 (mm dd yyyy hh:mmAM (または PM)) として定義されます。このデフォルトの順序を変更するには、**convert** 関数を使用します。

**isql** セッションでは、デフォルトのプロンプトラベルは、デフォルトワイルドカード `:?*` か、または *wildcard* の値です。プロンプトラベルをカスタマイズするには、ワイルドカードの後に 1 語の文字列 (最大長 80 文字) を指定します。複数の語を含むプロンプトラベルを指定すると、最初の語より後の文字は無視されます。

---

**注意：** **isql** セッションで **isql** が `:?*` または *wildcard* の値をワイルドカードとして認識するのは、これらの文字が **isql** の行の先頭に配置されている場合に限られます。

---

### 例

クエリを編集できるテキストファイルに移動します。ファイルを作成して保存すると、**isql** に戻ります。クエリが表示されます。行に単独で `go` と入力すると、クエリが実行されます。

```
isql -Ujoe -Pabracadabra
1> select *
2> from authors
3> where city = "Oakland"
4> vi
```

**reset** と指定すると、クエリバッファがクリアされます。**quit** と指定すると、オペレーティングシステムに戻ります。

```
isql -Ualma
Password:
1> select *
```

```
2> from authors
3> where city = "Oakland"
4> reset
1> quit
```

roman8 文字セットを使用しているサーバに対して Macintosh から isql を実行することを指定します。

```
isql -a mac -J roman8
```

ストア ID 7896 の pubs2 データベースでの出力で、"#" 文字を使用してカラムの区切り文字を作成します。

```
isql -Usa -P -s#
1> use pubs2
2> go
1> select * from sales where stor_id = "7896"
#stor_id#ord_num          #date                      #
#-----#-----#-----#
#7896   #124152          #      Aug 14 1986 12:00AM#
#7896   #234518          #      Feb 14 1991 12:00AM#

(2 rows affected)
```

入力されたパスワードを表示せずに、パスワードを変更します。次の例では、プロンプトラベルとして "old" と "new" が使用されています。

```
$ isql -Uguest -Pguest -Smyase --conceal
sp_password
:?: old
/
:?:? new
-----
old
new
Confirm new
Password correctly set.
(Return status 0)
```

現在のユーザに対してロールをアクティブにします。次の例では、カスタムワイルドカードとプロンプトラベル "role" および "password" が使用されています。

```
$ isql -UmyAccount --conceal '*'Password:
set role
* role
with passwd
** password
on
go

role
password
Confirm password
```



# iqlpp SQL プリプロセッサユーティリティ

SAP Sybase IQ の SQL プリプロセッサユーティリティ **iqlpp** は、入力ファイル (.sql) に記述されている SQL 文を C 言語ソースに変換し、出力ファイル (.c) に出力します。

Embedded SQL は、C と C++ プログラミング言語用のデータベースプログラミングインタフェースです。Embedded SQL は、C または C++ のソースコードに混在した (埋め込まれた) SQL 文から構成されます。これらの SQL 文は、SQL プリプロセッサによって C または C++ のソースコードに変換されます。その後、このコードをコンパイルします。

## 構文

```
iqlpp [ options ] <in filename > [<out filename> ]
```

## パラメータ

次の表に、**iqlpp** ユーティリティの利用可能なオプションを示します。

表 33 : iqlpp のオプション

オプション	説明
<b>-d</b>	データサイズを適切なサイズにする。
<b>-e &lt;level&gt;</b>	<p>不適合な SQL 構文をエラーとして通知する。</p> <p>&lt;level&gt; に使用できる値を次に示す。</p> <ul style="list-style-type: none"> <li>• <b>c03</b> – コア SQL/2003 構文でない構文を通知する。</li> <li>• <b>p03</b> – 上級 SQL/2003 構文でない構文を通知する。</li> <li>• <b>c99</b> – コア SQL/1999 構文でない構文を通知する。</li> <li>• <b>p99</b> – 上級 SQL/1999 構文でない構文を通知する。</li> <li>• <b>e92</b> – 初級レベル SQL/1992 構文でない構文を通知する。</li> <li>• <b>i92</b> – 中級レベル SQL/1992 構文でない構文を通知する。</li> <li>• <b>f92</b> – 上級 SQL/1992 構文でない構文を通知する。</li> <li>• <b>t</b> – 上級 SQL/1992 構文でない構文を通知する。</li> <li>• <b>u</b> – 標準ではないホスト変数型を通知する。</li> </ul> <p>以前のバージョンとの互換性を維持するために、<b>e</b> (初級レベル SQL92 用)、<b>i</b> (中級レベル SQL92 用)、<b>f</b> (上級 SQL92 用)、<b>w</b> (サポートされているすべての構文を許可) もサポートされている。</p>

オプション	説明
<b>-h</b> <width>	出力する行の長さの最大値を制限する。
<b>-k</b>	SQLCODE のユーザ宣言をインクルードする。
<b>-n</b>	行番号。
<b>-o</b> <O/S spec>	ターゲットオペレーティングシステム (WINDOWS、WINNT、または UNIX) を指定する。
<b>-q</b>	クワイエットモード (バナー表示なし)。
<b>-r-</b>	再入可能コードを生成する。
<b>-s</b> <len>	コンパイラに与える文字列定数の長さの最大値を指定する。
<b>-w</b> <level>	<p>不適合な SQL 構文を警告として通知する。</p> <p>&lt;level&gt; に使用できる値を次に示す。</p> <ul style="list-style-type: none"> <li>• <b>c03</b> – コア SQL/2003 構文でない構文を通知する。</li> <li>• <b>p03</b> – 上級 SQL/2003 構文でない構文を通知する。</li> <li>• <b>c99</b> – コア SQL/1999 構文でない構文を通知する。</li> <li>• <b>p99</b> – 上級 SQL/1999 構文でない構文を通知する。</li> <li>• <b>e92</b> – 初級レベル SQL/1992 構文でない構文を通知する。</li> <li>• <b>i92</b> – 中級レベル SQL/1992 構文でない構文を通知する。</li> <li>• <b>f92</b> – 上級 SQL/1992 構文でない構文を通知する。</li> <li>• <b>t</b> – 上級 SQL/1992 構文でない構文を通知する。</li> <li>• <b>u</b> – 標準ではないホスト変数型を通知する。</li> </ul>
<b>-x</b>	複数バイト SQL 文字列をエスケープシーケンスに変更する。
<b>-z</b> <cs>	照合順を指定する。推奨される照合順のリストについては、コマンドプロンプトで <b>iqinit -l</b> と入力する。

### 使用法

SQL プリプロセッサは、コンパイラの実行前に、Embedded SQL を含んだ C または C++ プログラムを処理します。iqsqlpp は、入力ファイル *sql-filename* に記述されている SQL 文を C 言語ソースに変換し、*output-filename* に出力します。Embedded SQL を含んだソースプログラムの拡張子は通常 *.sql* です。デフォルトの出力ファイル名は、*sql-filename* に拡張子 *.c* を付けたものになります。*sql-filename* に拡張子 *.c* が付いている場合は、デフォルトの出力ファイル名拡張子は *.cc* になります。



## オプション

- **-d** – データサイズを適切なサイズにします。データ領域のサイズを小さくするコードを生成します。データ構造体を再利用し、実行時に初期化してから使用します。これはコードサイズを増加させます。
- **e <flag>** – このオプションは、指定された SQL92 セットの一部ではない Embedded SQL をエラーとして通知します。

設定できる <flag> の値とその意味を次に示します。

- **e** – 初級レベル SQL92 構文でない構文を通知する。
- **i** – 中級レベル SQL92 構文でない構文を通知する。
- **f** – 上級レベル SQL92 構文でない構文を通知する。
- **t** – 標準ではないホスト変数型を通知する。
- **w** – サポートされている構文をすべて許可する。
- **u** – UltraLite でサポートされていない構文を通知する。
- **c99** – コア SQL/1999 構文でない構文を通知する。
- **c03** – コア SQL/2003 構文でない構文を通知する。
- **p99** – 上級 SQL/1999 構文でない構文を通知する。
- **p03** – 上級 SQL/2003 構文でない構文を通知する。
- **-h width** – **iqlpp** によって出力される行の最大長を *width* の値に制限します。行の内容が次の行に続くことを表す文字は円記号 (¥) です。また、*width* に指定できる最小値は 10 です。
- **-k** – コンパイルされるプログラムが **SQLCODE** のユーザ宣言をインクルードすることをプリプロセッサに通知します。
- **-n** – C ファイルに行番号情報を生成します。これは、生成された C コード内の適切な場所にある *#line* 指令で構成されます。使用しているコンパイラが *#line* 指令をサポートしている場合、このオプションを使用すると、コンパイラは SQC ファイル (Embedded SQL が含まれるファイル) 中の行番号を使用してその場所のエラーをレポートします。これは、SQL プリプロセッサによって生成された C ファイル中の行番号を使用して、その場所のエラーをレポートするのは対照的です。また、ソースレベルデバッグも、*#line* 指令を間接的に使用します。このため、SQC ソースファイルを表示しながらデバッグできます。
- **-o** – <O/S spec> ターゲットオペレーティングシステムを指定します。このオプションは、プログラムを実行するオペレーティングシステムと一致する必要があります。プログラム内に、特殊記号への参照が生成されます。この記号はインタフェースライブラリで定義されます。適切でないオペレーティングシステムを指定したり、適切でないライブラリを使用したりすると、リンカがエ

ラーを検知します。サポートされているオペレーティングシステムは次のとおりです。

- **WINDOWS** – Microsoft Windows
- **UNIX** – このオプションは、32 ビットの UNIX アプリケーションを作成する場合に使用します。
- **UNIX64** – このオプションは、64 ビットの UNIX アプリケーションを作成する場合に使用します。
- **-q** – クワイエットモードで作動します。バナーを表示しません。
- **-r** – 再入可能コードを生成します。
- **-s <len>** – プリプロセッサが C ファイルに出力する文字列の最大サイズを設定します。この値より長い文字列は、文字のリスト (“a,” “b,” “c,” など) を使用して初期化されます。ほとんどの C コンパイラには、処理できる文字列リテラルのサイズに制限があります。このオプションを使用して上限を設定します。デフォルト値は 500 です。
- **-w <flag>** – このオプションは、指定された SQL92 セットの一部ではない Embedded SQL を警告として通知します。

設定できる <flag> の値とその意味を次に示します。

- **e** – 初級レベル SQL92 構文でない構文を通知する。
- **i** – 中級レベル SQL92 構文でない構文を通知する。
- **f** – 上級レベル SQL92 構文でない構文を通知する。
- **t** – 標準ではないホスト変数型を通知する。
- **u** – UltraLite でサポートされていない構文を通知する。
- **w** – サポートされている構文をすべて許可する。
- **c03** – コア SQL/2003 構文でない構文を通知する。
- **c99** – コア SQL/1999 構文でない構文を通知する。
- **p99** – 上級 SQL/1999 構文でない構文を通知する。
- **p03** – 上級 SQL/2003 構文でない構文を通知する。
- **-x** – 複数バイト文字列をエスケープシーケンスに変更して、コンパイラを通過できるようにします。
- **-z <cs>** – 照合順を指定します。推奨される照合順のリストについては、コマンドプロンプトで **iqinit -l** と入力してください。

照合順は、プリプロセッサにプログラムのソースコードで使用されている文字を理解させるために使用します。たとえば、識別子に使用できるアルファベット文字の識別などに使用されます。**-z** が指定されていない場合、プリプロセッ

サは、オペレーティングシステム、IQLANG 環境変数、IQCHARSET 環境変数に基づいて、使用する合理的な照合順を決定しようとします。

## **iqsqlpp** SQL プリプロセッサユーティリティ

# start\_iq データベースサーバ起動ユーティリティ

データベース起動ユーティリティ **start\_iq** は、SAP Sybase IQ ネットワークデータベースサーバを起動します。

SAP Sybase IQ の UNIX バージョンにはスクリプト **start\_iq**、Windows バージョンにはスクリプト **start\_iq.exe** がそれぞれ含まれています。これらのスクリプトは、現在の環境が正しく設定されているか検証し、推奨されるデフォルト値にあらかじめ設定されている必須のスイッチ (および追加設定したスイッチ) を使用してサーバを起動します。また、**start\_iq** ユーティリティは、一部のパラメータを受け取り、その他のパラメータを計算します。使用しているオペレーティングシステムに固有のスイッチの詳細については、『インストールおよび設定ガイド』を参照してください。

---

**注意：** SAP Control Center は、サーバの起動を容易化するグラフィカルインタフェースを備えています。マルチプレックスサーバは、Sybase Control Center から起動することをおすすめします。

---

## start\_iq ユーティリティの起動

---

Windows または UNIX のコマンドラインから、**start\_iq** データベース起動ユーティリティを起動します。

1. 次の形式でコマンドを発行します。

```
start_iq [ server-options ] [ database-file  
[ database-options ], ...]
```

2. [server-options] で **-n** スイッチを指定すると、間違ったサーバに誤って接続することを回避できます。

## すべての start\_iq スイッチのリスト

---

*server-options*、*database-file*、*database-options* の各パラメータの利用可能なすべてのスイッチをリスト表示できます。  
次のコマンドを発行します。

```
start_iq -?
```

## start\_iq のサーバオプション

---

start\_iq *server-options* パラメータで利用可能なスイッチについて説明します。

参照：

- データベースサーバの命名制限 (189 ページ)

## @data iqsrv16 データベースサーバオプション

指定された環境変数または設定ファイルからオプションを読み込みます。

構文

```
iqsrv16 @data ...
```

適用対象

すべてのオペレーティングシステムとデータベースサーバ。言語選択ユーティリティ (dblang)、証明書の作成ユーティリティ (createcert)、証明書ビューアユーティリティ (viewcert)、Microsoft ActiveSync プロバイダインストールユーティリティ (mlasinst)、ファイル非表示ユーティリティ (dbfhide) を除くすべてのデータベースユーティリティでサポートされます。

言語選択ユーティリティ (dblang)

証明書作成ユーティリティ (createcert)

証明書ビューアユーティリティ (viewcert)

Microsoft ActiveSync プロバイダインストールユーティリティ (mlasinst)

ファイル難読化ユーティリティ (dbfhide)

備考

このオプションを使用して、指定された環境変数または設定ファイルからコマンドラインオプションを読み出します。指定された名前と同じ名前の環境変数と設定ファイルが両方存在する場合は、環境変数が使用されます。

設定ファイルには、改行を含めたり、あらゆるオプションの設定を格納したりできます。

設定ファイルの情報を (パスワードが含まれるなどの理由で) 保護する場合は、ファイル難読化ユーティリティ (dbfhide) を使用して、設定ファイルの内容を難読化してください。

@data パラメータはコマンドの任意の位置に指定でき、ファイルに含まれるパラメータがその位置に挿入されます。複数のファイルを指定可能で、ファイル指定子をコマンドラインオプションで使用できます。

## 例

次の設定ファイルには、myserver という名前のサーバをキャッシュサイズ 4 MB で起動し、サンプルデータベースをロードするオプションのセットが含まれています。

```
-c 4096
-n myservers
"c:¥mydatabase.db"
```

この設定ファイルを c:¥config.txt として保存すると、コマンドで次のように使用できます。

```
iqsrv16 @c:¥config.txt
```

次の設定ファイルにはコメントが含まれています。

```
#This is the server name:
-n MyServer
#These are the protocols:
-x tcpip
#This is the database file
my.db
```

次の文は、データベースサーバをキャッシュサイズ 4 MB で起動し、サンプルデータベースをロードするオプションを格納する環境変数を設定します。

```
SET envvar=-c 4096 "c:¥mydatabase.db";
```

次のコマンドは、envvar という環境変数を使用してデータベースサーバを起動します。

```
iqsrv16 @envvar
```

## @envvar iqsrv16 サーバオプション

指定した環境変数からコマンドラインスイッチを読み込みます。

### 構文

```
@envvar
```

### 備考

環境変数には、あらゆるコマンドラインスイッチのセットを格納できます。たとえば、次に示した Windows 文の 1 行目では、コマンドラインスイッチのセットを含んだ環境変数を設定し、サンプルデータベースをロードします。2 行目の文で、データベースサーバを起動します。

```
set envvar= -gp 4096 -gm 15
c:¥sybase¥IQ-16_0¥demo¥start_iq -n myservers @envvar iqdemo.db
```

## start\_iq データベースサーバ起動ユーティリティ

---

**注意：** @ コマンドラインスイッチの値をファイルと環境変数の両方で指定すると、予測できない結果となります。これらの方法のいずれか 1 つだけを使って、特定の @ コマンドラインスイッチを設定してください。

---

### **@filename iqsrv16 サーバオプション**

設定ファイルからスイッチを読み込みます。

構文

@filename

備考

ファイルには、改行を含めたり、あらゆるコマンドラインスイッチのセットを格納したりできます。たとえば、次の Windows コマンドファイルに含まれている iqdemo という一連のサーバコマンドラインスイッチでは、使用可能な接続数が 10 に設定され、カタログページの最大サイズが 4096 バイトに設定され、iqdemo というデータベースが起動されます。

```
# iqdemo.cfg
# -----
# Default startup parameters for the IQ demo database
# -----
-n iqdemo
-x tcpip{port=2638}
# The following parameters are also found in the configuration file
# %IQDIR%\scripts\default.cfg. Any parameters not specified below
# and not in the start up parameter list, will be added by start_iq
# using default.cfg as a guide.

-c 48m
-gc 20
-gd all
-gl all
-gm 10
-gp 4096
-iqmc 32
-iqtc 24
```

この設定ファイルが c:\%config.txt として保存されている場合、このファイルをコマンドラインで次のように使用できます。

```
start_iq @c:\%config.txt
```



## **-c iqsrv16** サーバオプション

カタログストアページとその他のサーバ情報のキャッシュに使うために予約する初期メモリを設定します。

### 構文

```
-c cache-size [ k | m | g | p ]
```

### 備考

データベースページをキャッシュに入れるためのメモリがキャッシュ内に別途確保されている場合は、そのメモリがデータベースサーバで使用されます。10000 より小さいキャッシュサイズはKB単位とみなします (1K = 1024 バイト)。10000 以上のキャッシュサイズはバイト単位とみなします。キャッシュサイズ nK, nM、または nP (1M = 1024 KB) も設定できます。ここで、P は物理システムメモリの割合です。

default.cfg ファイルの **-c** と **start\_iq** のデフォルト値は、Windows プラットフォームでは 32MB (**-c 32M**)、UNIX プラットフォームでは 48MB (**-c 48M**) です。このデフォルトを使用するか、またはより大きな値に **-c** を設定します。

P の代わりに % も使用できますが、UNIX 以外のオペレーティングシステムでは、ほとんどの場合 % を環境変数のエスケープ文字として使用するため、% 文字をエスケープする必要があります。たとえば、システムの物理メモリの 20 パーセントを使用するには、次のように指定します。

```
start_iq -c 20%% ...
```

**-c** を、**-ch** または **-cl** と同じ設定ファイルまたはコマンドラインで使用しないでください。関連情報については、**-ch cache-size** オプションおよび **-ca 0** オプションを参照してください。

## **-al iqsrv16** サーバオプション

標準認証を使用して、特定数のユーザのみに LDAPUA の LOGIN\_MODE を拡張します。

### 構文

```
-al "user1;user2;user3" server_name.cfg database-name.db
```

### 備考

- 最大 5 つのユーザ ID をセミコロンで区切って指定できます。各ユーザ ID は二重引用符で囲みます。

## start\_iq データベースサーバ起動ユーティリティ

- サーバレベルでの実行時には、**-al** スイッチはサーバの次の再起動時まで有効です。

## **-ca iqsrv16** データベースサーバオプション

静的キャッシュサイズを強制的に適用します。

### 構文

```
iqsrv16 -ca 0 ...
```

### 適用対象

Windows、UNIX

### 備考

-ca 0 オプションを指定すると、自動キャッシュサイズチューニングに無効にすることができます。-ca 0 オプションの設定がない場合、データベースサーバは自動的にキャッシュサイズを増加します。このオプションを指定した場合、動的メモリが足りなくなったことを示すエラーがデータベースサーバで発生しても、キャッシュサイズは引き続き調整されます。

このサーバオプションは、-ca 0 という形式でのみ使用してください。

### 例

次の例は、総物理メモリ量の 40% の静的キャッシュがある myserver というデータベースサーバを起動し、サンプルデータベースをロードします。データベースサーバでは、キャッシュサイズは自動的に調整されません。

```
start_iq @iqdemo.cfg iqdemo.db
```

次の例は、Windows の場合です。

```
start_iq @"%  
    ALLUSERSPROFILE%¥SybaseIQ¥demo¥iqdemo.cfg" -c 40P  
    -ca  
    0 -n myserver "%  
    ALLUSERSPROFILE%¥SybaseIQ¥demo¥iqdemo.db"
```

次の例は、Unix の場合です。

```
start_iq  
    @$IQDIR16/demo/iqdemo.cfg -c 40P -ca 0  
    $IQDIR16/demo/iqdemo.db
```

## -cc iqsrv16 データベースサーバオプション

次にデータベースが起動されるときに、キャッシュウォーミングに使用するデータベースページに関する情報を収集します。

### 構文

```
iqsrv16 -cc{ + | - } ...
```

### 適用対象

すべてのオペレーティングシステムとデータベースサーバ

### 備考

デフォルトでは、ページ収集はオンになっています。収集がオンになると、データベースサーバは要求された各データベースページを追跡し続けます。ページの収集は、最大ページ数が収集されるか、データベースが停止されるか、または収集率が最小値を下回った場合に終了します。収集する最大ページ数を設定したり、収集率の値を指定したりすることはできません(この値は、キャッシュサイズとデータベースサイズに基づいています)。いったん収集が停止すると、要求されたページに関する情報はデータベースに記録されるので、それらのページは次回データベースが `-cr` オプションで開始されるときに、キャッシュの準備に使用できます。参照されたページの収集は、デフォルトではオンになっています。

## -ch iqsrv16 サーバオプション

カタログストアのキャッシュサイズの上限(バイト数)を設定します。

### 構文

```
-ch size[ k| m| g| p]
```

### 備考

デフォルトでは、上限値の概算は、256MB とマシンの物理メモリの 90% のうち、いずれか低い方になります。

`-c` オプションと同様に、**K**、**M**、**P** の単位文字を使ってキャッシュサイズを指定します。キャッシュサイズ引数の意味と使用法、および **K**、**M**、**P** の単位文字の詳細については、「`-c cache-size`」を参照してください。

大量の解析を必要とする特定のクエリに対応する場合など、標準のカタログキャッシュサイズでは小さすぎる場合があります。このような場合は、`-cl` と `-ch` を設定すると有効なことがあります。たとえば、32 ビットプラットフォームでは次の設定を試してください。

```
-cl 128M  
-ch 512M
```

---

**警告！** カタログストアのキャッシュサイズを明示的に制御するには、設定ファイル (.cfg) またはサーバ起動時の UNIX コマンドラインで、次のいずれかを実行する必要があります。ただし、両方は実行しないでください。

- **-c** パラメータを設定する。
- **-ch** パラメータと **-cl** パラメータを使用して、カタログストアのキャッシュサイズの特定の上限と下限を設定する。

上記のパラメータをこれ以外の組み合わせで指定すると、予期しない結果が生じることがあります。

---

### 参照：

- **-c** iqsrv16 サーバオプション (149 ページ)
- **-cl** iqsrv16 データベースサーバオプション (152 ページ)

## **-cl iqsrv16** データベースサーバオプション

動的キャッシュサイズ変更機能に対して最小キャッシュサイズを設定します。

### 構文

```
iqsrv16 -cl { size[ k | m | g | p ] } ...
```

### デフォルト

Windows では 2 MB

UNIX では 8 MB

### 適用対象

Windows、UNIX

### 備考

このオプションは、キャッシュの下限値を設定します。**-c** を指定し、**-cl** を指定しないと、最小キャッシュサイズは初期キャッシュサイズ (**-c** 設定) に設定されます。**-c** と **-cl** をどちらも指定しないと、必要に応じてキャッシュを縮小できるように、最小キャッシュサイズは低い定数値に設定されます。Windows プラットフォームでは、この値は 2 MB になります。

*size* には、メモリ容量をバイト単位で指定します。単位をキロバイト、メガバイト、またはギガバイトで指定するには、それぞれ *k*、*m*、または *g* を使用してください。

単位 *p* は、物理システムメモリと、サポートされる最大キャッシュサイズのうち、いずれか小さい方のパーセンテージを表します。サポートされる最大キャッシュサイズは、オペレーティングシステムによって異なります。次に例を示します。

- 2.5 GB – Windows 32 ビット Advanced Server、Enterprise Server、Datacenter Server
- 3.5 GB – Windows x64 Edition 上の 32 ビットデータベースサーバ
- 1.5 GB – その他すべての 32 ビットシステム
- 64 ビットのデータベースサーバの場合、キャッシュサイズは無制限と見なされる

p を使用すると、引数はパーセンテージを表します。p の代わりに % も使用できますが、Windows ではほとんどの場合 % を環境変数のエスケープ文字として使用するため、% 文字をエスケープする必要があります。

たとえば、初期キャッシュサイズを物理システムメモリの 50% に設定するには、次のコマンドを実行できます。

```
iqsrv16 -cl 50% ...
```

---

**注意：**初期キャッシュサイズまたは最小キャッシュサイズを最大キャッシュサイズの 8 分の 1 より小さく設定しようとする、サポートされるキャッシュサイズに合わせて自動的にキャッシュサイズが増やされます。

---

### 例

次の例は、silver という名前のデータベースサーバを最小キャッシュサイズ 5 MB で起動し、データベースファイル example.db をロードします。

```
iqsrv16 -cl 5m -n silver "c:¥example.db"
```

## -cp iqsrv16 データベースサーバオプション

クラスを検索する追加の一連のディレクトリまたは JAR ファイルを指定します。

### 構文

```
iqsrv16 -cp location[ ;location ... ] ...
```

### 適用対象

すべてのオペレーティングシステムとデータベースサーバ。

### 備考

データベース内の Java で使用しているすべてのクラスと JAR ファイルはデータベース内にインストールすることをおすすめします。クラスと JAR ファイルをデータベース内に保存すると、データベースを簡単に別のコンピュータやオペレーティングシステムに移動できます。また、クラスと JAR ファイルをデータベース内にインストールすると、データベースサーバのクラスローダでクラスとリソースをデータベースからフェッチできるという利点もあります。その結果、データベース内の Java を使用している各接続で、これらのクラスの独自のインスタンスと、クラス内の静的変数の独自のコピーを使用できます。

## start\_iq データベースサーバ起動ユーティリティ

ただし、クラスまたは JAR ファイルをシステムクラスローダによってロードする必要がある場合は、`java_class_path` データベースオプションまたは `-cp` データベースサーバオプションで指定できます。どちらのオプションを指定しても、Java VM の起動用にデータベースサーバによって構築されるクラスパスにクラスと JAR ファイルが追加されます。`java_class_path` データベースオプションは、サーバが複数のデータベースを実行していて、それぞれのデータベースにシステムクラスローダによってロードする必要のある異なるセットの JAR およびディレクトリがあるときに便利です。`-cp` データベースサーバオプションは、サーバ上のすべてのデータベースが同一のクラスまたは JAR ファイルを必要とするときに便利です。

### **-cr iqsrv16 データベースサーバオプション**

データベースが最後に実行されたときに収集した情報を使用して、キャッシュとデータベースページを再ロード (準備) します。

#### 構文

```
iqsrv16 -cr{ + | - } ...
```

#### 適用対象

すべてのオペレーティングシステムとデータベースサーバ

#### 備考

データベースが最後に起動したときに参照されたページを使用して、データベースサーバにキャッシュを準備するよう指示できます (ページ収集は `-cc` オプションを使用してオンにします)。キャッシュウォーミングは、デフォルトではオンになっています。データベースが起動されると、サーバはデータベースをチェックして、データベースが最後に起動されたときに要求されたページの収集があるかどうかを確認します。データベースにこの情報が含まれている場合、前に参照したページがキャッシュにロードされます。

データベースが最後に起動されたときに参照されたページのキャッシュを準備することで、同じクエリまたは似たようなクエリがデータベースの起動のたびに実行されるような場合に、パフォーマンスを改善できます。

### **-cs iqsrv16 データベースサーバオプション**

動的キャッシュサイズ決定に関する統計をデータベースサーバメッセージウィンドウに表示します。

#### 構文

```
iqsrv16 -cs ...
```

#### 適用対象

Windows、UNIX

### 備考

トラブルシューティング目的で、キャッシュサイズの調整方法を決めるためにデータベースサーバで使用される統計が、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。

## **-cv iqsrv16 データベースサーバオプション**

データベースサーバメッセージウィンドウでのキャッシュウォーミングに関するメッセージの表示を制御します。

### 構文

```
iqsrv16 -cv{ + | - } ...
```

### デフォルト

キャッシュウォーミングメッセージは表示されません。

### 適用対象

すべてのオペレーティングシステムとデータベースサーバ

### 備考

-cv+ が指定されると、次のいずれかのキャッシュウォーミングアクティビティが発生した場合に、データベースサーバメッセージウィンドウにメッセージが表示されます。

- 要求されたページの収集が開始または停止する (-cc サーバオプションで制御)
- ページの再ロードが開始または停止する (-cr サーバオプションで制御)

### 例

次のコマンドは、データベースページの収集とページのロードをオンにして mydatabase.db というデータベースを起動し、これらのアクティビティに関するメッセージをデータベースサーバメッセージウィンドウに表示します。

```
iqsrv16 -cc+ -cr+ -cv+ mydatabase.db
```

## **-dt iqsrv16 データベースサーバオプション**

テンポラリファイルを保存するディレクトリを指定します。

### 構文

```
iqsrv16 -dt temp-file-dir ...
```

### 適用対象

すべてのサーバおよびオペレーティングシステム (UNIX 上の共有メモリ接続を除く)

### 備考

データベースサーバでは、次の2種類のテンポラリファイルが作成されます。

- **データベースサーバ関連のテンポラリファイル (すべてのプラットフォームで作成)** -dt オプションでは、データベースサーバ関連のテンポラリファイルを保存するディレクトリを指定できます。このオプションを指定しないでデータベースサーバを起動すると、データベースサーバは以下の環境変数をこの通りの順序で調べ、テンポラリファイルの保存先とするディレクトリを決定します。

Windows/UNIX
1. IQTMP16
2. TMP
3. TMPDIR
4. TEMP

これらの環境変数がいずれも定義されていない場合、テンポラリファイルは、Windows の場合は現在のディレクトリ、UNIX の場合は /tmp ディレクトリに作成されます。

- **通信関連のテンポラリファイル (クライアントとデータベースサーバの両方に対して UNIX のみで作成)** -UNIX では、通信関連のテンポラリファイルは -dt で指定されたディレクトリには保存されません。その代わりに、データベースサーバは以下の環境変数をこのままの順序で調べ、テンポラリファイルの保存先にするディレクトリを決定します。

Unix
1. IQTMP16
2. TMP
3. TMPDIR
4. TEMP

これらの環境変数がいずれも定義されていない場合、テンポラリファイルは UNIX では /tmp ディレクトリに作成されます。

UNIX では、共有メモリを介して接続する場合、クライアントとデータベースサーバの両方で IQTMP16 を同じ値に設定する必要があります。

### 例

データベースサーバ関連のテンポラリファイルの場所を特定するには、DB\_PROPERTY システム関数で TempFileName プロパティを使用します。

```
SELECT DB_PROPERTY ( 'TempFileName' );
```



## **-ec iqsrv16** データベースサーバオプション

トランスポートレイヤセキュリティまたは暗号化を使用して、すべてのクライアントとの間で転送されるすべての Command Sequence 通信プロトコルパケット (DBLib、ODBC、OLE DB) を暗号化します。TDS パケットは暗号化されません。

### 構文

```
iqsrv16 -ec encryption-options ...
```

```
encryption-options :
```

```
{ NONE |
  SIMPLE |
  TLS ( [ FIPS={ Y | N }; ]
  IDENTITY=server-identity-filename;
  IDENTITY_PASSWORD=password ) }, ...
```

### 指定可能な値

- **NONE** – 暗号化されない接続を受け入れます。
- **SIMPLE** – 単純暗号化された接続を受け入れます。このタイプの暗号化は、すべてのプラットフォームで、また以前のバージョンのデータベースサーバとクライアントでサポートされます。単純暗号化では、サーバ認証、RSA 暗号化、またはその他のトランスポートレイヤセキュリティ機能は提供されません。
- **TLS** – RSA 暗号化で暗号化された接続を受け入れます。TLS パラメータは次の引数を受け取ります。
  - **FIPS** – FIPS 認定の RSA 暗号化の場合は、FIPS=Y を指定します。RSA FIPS 認定の暗号化は別の認定ライブラリを使用しますが、9.0.2 以降で RSA を指定しているクライアントと互換性があります。  
FIPS 認定コンポーネントがサポートされているプラットフォームのリストについては、<http://www.sybase.com/detail?id=1061806> を参照してください。  
アルゴリズムは、証明書を作成するときに使用される暗号化と一致する必要があります。
  - **server-identity-filename** – サーバ ID 証明書のパスとファイル名を指定します。FIPS 認定の RSA 暗号化を使用している場合は、RSA アルゴリズムを使用して証明書を生成する必要があります。
  - **password** – サーバのプライベートキーのパスワードを指定します。このパスワードは、サーバ証明書を作成するときに指定します。

### 適用対象

NONE と SIMPLE は、すべてのサーバとオペレーティングシステムに適用されません。

## start\_iq データベースサーバ起動ユーティリティ

TLS は、すべてのサーバとオペレーティングシステムに適用されます。

FIPS 認定の暗号化サポートの詳細については、<http://www.sybase.com/detail?id=1061806> を参照してください。

### 備考

このオプションは、トランスポートレイヤセキュリティを使用してクライアントアプリケーションとデータベースサーバ間の通信パケットを安全化する場合に使用します。

-ec オプションを指定すると、データベースサーバは指定された暗号化タイプによって暗号化される接続のみ受け入れます。カンマ区切りリストで、少なくとも1つのサポートされているパラメータを指定してください。TDS プロトコルを介した接続は、jConnect を使用する Java アプリケーションを含みますが、-ec オプションの使用に関係なく常に受け入れられ、暗号化されることはありません。この TDS プロトコルオプションを NO に設定すると、これらの暗号化されていない TDS 接続は禁止されます。

デフォルトでは、通信パケットは暗号化されないため、セキュリティに潜在的なリスクがあります。ネットワークパケットのセキュリティが心配な場合は、-ec オプションを使用します。暗号化がパフォーマンスに及ぼす影響はごくわずかです。

データベースサーバが単純暗号化を受け入れ、暗号化されない接続を受け入れない場合、暗号化を使用しない TDS 接続以外の接続では、単純暗号化が使用されません。

-ec SIMPLE を指定してデータベースサーバを起動すると、データベースサーバは単純暗号化を使用した接続だけを受け入れます。TLS 接続 (RSA 暗号化、RSA FIPS 認定暗号化) は失敗し、暗号化を要求しない接続では単純暗号化が使用されます。

データベースサーバで TCP/IP 上の暗号化された接続を受け入れ、さらに共有メモリを介してローカルコンピュータのデータベースへも接続できるようにする場合は、データベースサーバの起動時に -ec オプションとともに -es オプションを指定できます。

dbrsa16.dll ファイルには、暗号化と復号化に使用される RSA コードが含まれています。dbfips16.dll ファイルには、FIPS 認定の RSA アルゴリズムのコードが含まれています。データベースサーバに接続するときに、適切なファイルが見つからなかったり、エラーが発生したりすると、データベースサーバメッセージウィンドウにメッセージが表示されます。指定されたタイプの暗号化を開始できない場合、サーバは起動しません。

クライアントとサーバで暗号化の設定が一致していることが必要です。設定が異なっていると、次の場合を除き、接続は失敗します。

- データベースサーバに対して `-ec SIMPLE` を指定し、`-ec NONE` を指定しなかった場合、暗号化を要求しない接続は許可され、自動的に単純暗号化が使用されます。
- データベースサーバ側で `RSA` を指定し、クライアント側で `FIPS` 認定の暗号化を指定している場合、またはその逆の場合には、接続は成功します。この場合、`Encryption` 接続プロパティはデータベースサーバ側で指定された値を返します。

---

**注意：** 強力な暗号化テクノロジーはすべて、輸出規制対象品目です。

---

### 例

次の例は、暗号化されない接続と単純暗号化を使用する接続を許可します。

```
iqsrv16 -ec NONE,SIMPLE -x tcpip c:¥mydemo.db
```

次の例は、`RSA` サーバ証明書 `rsaserver.id` を使用するデータベースサーバを起動します。

```
iqsrv16 -ec TLS (IDENTITY=rsaserver.id;IDENTITY_PASSWORD=test) -x  
tcpip c:¥mydemo.db
```

次の例は、`FIPS` 認定の `RSA` サーバ証明書 `rsaserver.id` を使用するデータベースサーバを起動します。

```
iqsrv16 -ec TLS (FIPS=Y;IDENTITY=rsaserver.id;IDENTITY_PASSWORD=test)  
-x tcpip c:¥mydemo.db
```

## -ep iqsrv16 データベースサーバオプション

強力に暗号化されたデータベースを起動するときに、暗号化キーをユーザに要求します。

### 構文

```
iqsrv16 -ep ...
```

### 適用対象

すべてのオペレーティングシステムとデータベースサーバ。

### 備考

`-ep` オプションを指定すると、データベースサーバは、コマンドラインから起動されるときに暗号化キーの入力が必要なデータベースについて、ユーザが暗号化キーを入力するためのウィンドウを表示します。このサーバオプションでは、クリアテキストで暗号化キーを見ることができないようにすることで、高いセキュリティが得られます。

データベースサーバでこのオプションを指定した場合、次の条件がすべて該当する場合にユーザは暗号化キーの入力を要求されます。

## start\_iq データベースサーバ起動ユーティリティ

- -ep オプションが指定されている
- サーバが起動中である
- データベースを起動するキーが必要である
- データベースサーバが Windows サービスではない、またはデスクトップとの対話オプションがオンになった Windows サービスである
- サーバがデーモンではない (UNIX)。

クライアントアプリケーションとデータベースサーバ間の通信パケットの安全性を保つため、-ec サーバオプションとトランスポートレイヤセキュリティを使用します。

### 例

myencrypted.db データベースが起動すると、ユーザは暗号化キーの入力を要求されます。

```
iqsrv16 -ep -x tcpip myencrypted.db
```

## -es iqsrv16 データベースサーバオプション

共有メモリを経由した暗号化されていない接続を許可します。

### 構文

```
iqsrv16 -ec encryption-options -es ...
```

### 適用対象

すべてのサーバとオペレーティングシステム。

### 備考

このオプションは、-ec オプションとともに指定された場合のみ有効です。-es オプションは、共有メモリを経由した、暗号化されていない接続を許可するようにデータベースサーバに指定します。TCP/IP を介した接続では、-ec オプションで指定された暗号化タイプを使用する必要があります。このオプションは、リモートクライアントからのデータベースアクセスには暗号化された接続を使用し、パフォーマンス上の理由から、ローカルコンピュータからのデータベースアクセスには暗号化されていない接続を使用できるようにしたい場合に便利です。

### 例

次の例は、単純暗号化を使用する接続と、共有メモリを経由した暗号化されない接続を許可します。

```
iqsrv16 -ec SIMPLE -es -x tcpip c:¥mydemo.db
```

## **-fips iqsrv16 データベースサーバオプション**

強力なデータベースと通信の暗号化に FIPS 認定のアルゴリズムのみを使用することを要求できます。

### 構文

```
iqsrv16 -fips ...
```

### 適用対象

Windows、UNIX、Linux

### 備考

このオプションを指定すると、すべてのデータベースサーバ暗号化で FIPS 認定のアルゴリズムが使用されます。このオプションは、データベースの強力な暗号化、クライアント/サーバのトランスポートレイヤセキュリティ、Web サービスのトランスポートレイヤセキュリティに適用されます。`-fips` オプションが指定されているときには、暗号化されていない接続とデータベースは使用できますが、単純暗号化は使用できません。

---

**注意：**強力な暗号化テクノロジーはすべて、輸出規制対象品目です。

---

データベースの強力な暗号化では、`-fips` オプションを指定すると、CREATE DATABASE 文の ALGORITHM 句で FIPS 認定相当の AES と AES256 が指定されていても、新しいデータベースでそれが使用されます。

`-fips` を指定してデータベースサーバを起動した場合、AES、AES256、AES\_FIPS、AES256\_FIPS の暗号化方式で暗号化されたデータベースを実行できますが、単純暗号化方式で暗号化されたデータベースは実行できません。暗号化されていないデータベースはサーバで開始できます。

AES\_FIPS または AES256\_FIPS で暗号化したデータベースを実行するために使用するコンピュータには、SAP Sybase IQ セキュリティオプションをインストールしてください。

トランスポートレイヤセキュリティでは、`-fips` オプションを指定すると、RSA が指定されていても、サーバは FIPS 認定の RSA アルゴリズムを使用します。

Web サービス用のトランスポートレイヤセキュリティでは、`-fips` オプションを指定すると、HTTPS が指定されていてもサーバは FIPS 認定の HTTPS FIPS を使用します。

`-fips` を指定すると、ENCRYPT 関数と HASH 関数では FIPS 認定の RSA 暗号化アルゴリズムが使用され、パスワードハッシングではアルゴリズムとして SHA-256 ではなく SHA-256 FIPS が使用されます。

## -ga iqsrv16 データベースサーバオプション

最後の非 HTTP クライアント接続を切断した後、データベースをアンロードします。

### 構文

```
iqsrv16 -ga ...
```

### 適用対象

すべてのオペレーティングシステム

### 備考

ネットワークサーバでこのオプションを指定すると、最後の非 HTTP クライアント接続が切断された後に各データベースがアンロードされます。最後の非 HTTP クライアント接続が切断された後に各データベースをアンロードし、最後のデータベースが停止したときにはサーバも停止します。

データベースへの唯一の接続が HTTP 接続で、自動的に停止するようにデータベースが設定されている場合、HTTP 接続が切断したときにデータベースはアンロードされません。また、-ga オプションを指定したデータベースに HTTP 接続と Command Sequence 接続または TDS 接続がある場合は、最後の Command Sequence 接続または TDS 接続が切断したときにデータベースが自動的に停止します。このときに HTTP 接続がまだあった場合は切断されます。

## -gb iqsrv16 データベースサーバオプション

サーバプロセスの優先度クラスを設定します。

### Windows 構文

```
iqsrv16 -gb { idle | normal | high | maximum } ...
```

### UNIX 構文

```
iqsrv16 -gb level ...
```

### 指定可能な値

- UNIX – UNIX の場合、*level* は -20 ~ 19 の整数です。UNIX のデフォルト値は、親プロセスの *nice* 値と同じです。*level* 値を下げると、より適切なスケジューリング優先度になります。-gb オプションには、*nice* 値の設定に関するすべての制限が適用されます。たとえば、ほとんどの UNIX プラットフォームの場合、プロセスの優先度レベルを下げる (たとえば、0 から -1 に変更する) ことができるのは root ユーザだけです。

- **Windows** – Windows の場合、一般的な設定として使用されるのは、normal または high です。値 idle は万全を期すために提供されています。値 maximum はコンピュータの実行を妨げる場合があります。

#### 適用対象

Windows、UNIX

## -gc iqsrv16 データベースサーバオプション

チェックポイント間の最大間隔を設定します。

#### 構文

```
iqsrv16 -gc minutes ...
```

#### デフォルト

60 分

#### 指定可能な値

- **minutes** – デフォルト値は、checkpoint\_time データベースオプションの設定値 (デフォルトは 60 分) です。値 0 を入力すると、デフォルト値の 60 が使用されます。

#### 適用対象

すべてのオペレーティングシステムとデータベースサーバ

#### 備考

このオプションは、各データベースでチェックポイントを行わずに、データベースサーバを実行する期待最大時間を分数で設定するために使用します。

通常、チェックポイントは、指定する時間より頻繁に発生します。

## -gd iqsrv16 データベースサーバオプション

実行中のデータベースサーバ上でデータベースを起動または停止するために必要な権限を設定します。

#### 構文

```
iqsrv16 -gd { DBA | all | none } ...
```

#### 指定可能な値

- **DBA** – SERVER OPERATOR システム権限を持つユーザのみがデータベースを起動または停止できます。

## start\_iq データベースサーバ起動ユーティリティ

- **all** – すべてのユーザがデータベースを起動または停止できます。リモートクライアントからアクセスできるネットワークサーバへの設定はおすすめしません。
- **none** – データベースサーバ自体が起動しているか停止している場合を除き、データベースの開始と停止は許可されません。

### デフォルト

デフォルトの設定は、ネットワークデータベースサーバの DBA です。大文字と小文字の両方の構文を使用できます。

### 適用対象

すべてのオペレーティングシステムとデータベースサーバ。

### 備考

このオプションは、ユーザが新しいデータベースファイルをデータベースサーバにロードするために、または実行中のデータベースサーバでデータベースを停止するために必要な権限のレベルを指定します。

オプションを DBA に設定した場合、クライアントアプリケーションは、データベースを開始または停止させるために同一のサーバ上で動作している別のデータベースへの既存の接続を使用する必要があります。DatabaseFile 接続パラメータを使用して、まだ実行中でないデータベースを起動することはできません。

次のように StartDBPermission サーバプロパティを使用して -gd オプションの設定を取得できます。

```
SELECT PROPERTY ( 'StartDBPermission' );
```

データベースサーバを停止するための権限は、-gk オプションで指定します。

### 例

ネットワークデータベースサーバで -gd オプションを使用する手順は、次のとおりです。

1. ネットワークデータベースサーバを起動します。

```
iqsrv16 -su mypwd -gd DBA -n my_server
```

2. Interactive SQL からユーティリティデータベースに接続します。

```
dbisql -c "UID=DBA;PWD=mypwd;DBN=utility_db"
```

3. データベースを起動します。

```
START DATABASE 'demo.db';
```

4. データベースに接続します。

```
CONNECT USING 'DBN=demo;UID=DBA;PWD=sql';
```



## -ge iqsrv16 データベースサーバオプション

外部関数のスタックサイズを設定します。

### 構文

```
iqsrv16 -ge integer ...
```

### デフォルト

32 KB

### 適用対象

Windows

### 備考

外部関数を実行するスレッドのスタックサイズをバイト数で設定します。

## -gf iqsrv16 データベースサーバオプション

サーバ上でトリガを起動不可にします。

### 構文

```
iqsrv16 -gf ...
```

### 適用対象

すべてのオペレーティングシステムとデータベースサーバ。

### 備考

-gf サーバオプションは、参照整合性トリガ (カスケード更新および削除など) を含めて、トリガの起動を無効にするようサーバに指示します。

## -gk iqsrv16 データベースサーバオプション

データベースサーバの停止に必要な権限を設定します。

### 構文

```
iqsrv16 -gk { DBA | all | none } ...
```

### 指定可能な値

- **DBA** - SERVER OPERATOR システム権限を持つユーザのみがデータベースサーバを停止できます。これはネットワークサーバのデフォルトです。
- **all** - データベースサーバを停止するのに権限が必要ありません。
- **none** - データベースサーバを停止できません。

## start\_iq データベースサーバ起動ユーティリティ

### 適用対象

すべてのオペレーティングシステムとデータベースサーバ。

### 備考

-gd データベースサーバオプションは、dbstop ユーティリティに適用されるほか、次の文にも適用されます。

- ALTER DATABASE *dbname* FORCE START 文
- STOP DATABASE 文

## -gl iqsrv16 サーバオプション

LOAD TABLE を使用してデータをロードするためのパーミッションを設定します。

### 構文

**-gl level**

### 備考

LOAD TABLE 文はデータベースサーバマシンからファイルを読み取ります。このような文を使用したファイルシステムアクセスを制御するために、-gl コマンドラインスイッチを使用して、このような文の使用に必要なデータベースパーミッションレベルを調整できます。levelは次のいずれかです。

- DBA - LOAD ANY TABLE システム権限、ALTER ANY TABLE システム権限、または ALTER ANY OBJECT システム権限を持つユーザのみがデータをロードできます。
- ALL - すべてのユーザがデータをロードできます。
- NONE - データのロードはできません。

オプションには大文字の構文も小文字の構文も使用できます。

デフォルト設定は、start\_iq を使用して起動されたサーバの場合は all で、その他の場合は dba です。以前のバージョンとの一貫性を維持するために、すべてのシステムで all 値を使用してください。iqdemo.cfg および default.cfg 設定ファイルでは、all 設定が使用されています。

## -gm iqsrv16 データベースサーバオプション

データベースサーバに対する同時接続の数を制限します。

### 構文

```
iqsrv16 -gm integer ...
```

### デフォルト

パーソナルサーバのデフォルト値は 10 です。ネットワークデータベースサーバのデフォルト値は 32766 ですが、この数値は、操作時にサーバで使用される内部テンポラリー接続によって削減されます。

### 適用対象

すべてのオペレーティングシステムとデータベースサーバ。

### 備考

サーバの接続制限を定義します。ライセンス契約に許可されている数より大きい値、もしくはメモリ制約を超えた値をここで設定した場合、その値は無効です。通常は、コンピュータリソースの制約から、ネットワークサーバへの接続の最大数はデフォルト値より小さくなります。

DROP CONNECTION システム権限を持つユーザがデータベースサーバに接続して他の接続を削除できるように、データサーバは、接続制限より 1 つ余分に DBA 接続を許可します。

## -gn iqsrv16 サーバオプション

複数のユーザが使用する際の接続の数と、カタログストアで使用される実行スレッドの数を設定します。

### 構文

**-gn** *integer*

### 備考

このパラメータは、すべてのオペレーティングシステムとサーバに適用されます。各接続では、1 つの要求に対して 1 つのスレッドを使用します。要求を処理すると、スレッドは他の接続が使用できるようプールに戻ります。接続は同時に複数の要求を処理できないので、接続が同時に複数のスレッドを使用することはありません。

この規則の例外となるのは、Java アプリケーションがスレッドを使用する場合です。Java アプリケーションでは、各スレッドがデータベースサーバで使われる実行スレッドとなります。

Windows を使用している場合は、**start\_iq** でこのパラメータを指定します。パラメータの値を計算するには、次の式を使用します。

```
gn_value >= gm_value * 1.5
```

**-gn** は、**-gm** の値の少なくとも 1.5 倍の値に設定します。最小値は 25 です。スレッドの総数は、プラットフォーム固有の最大スレッド数を超えないようにします。詳細については、「**-iqmt num**」を参照してください。

**start\_iq** データベースサーバ起動ユーティリティ

参照：

- `-iqmt iqsrv16` サーバオプション (177 ページ)

## **-gp iqsrv16** データベースサーバオプション

許可される最大データベースページサイズを設定します。

構文

```
iqsrv16 -gp { 2048 | 4096 | 8192 | 16384 | 32768 } ...
```

デフォルト

4096 (データベースをロードせずにデータベースサーバを起動した場合)

適用対象

すべてのオペレーティングシステムとデータベースサーバ。

備考

サーバのページサイズよりも大きいページサイズのデータベースファイルはロードできません。このオプションは、サーバのページサイズをバイト数で明示的に設定します。

デフォルトでは、サーバページサイズは、コマンドラインで指定されたデータベースの最大ページサイズと同じ大きさです。

いずれのプラットフォームでも、このオプションを指定せずにデータベースがロードされていない状態でサーバを起動した場合、デフォルト値は 4096 になります。

## **-gr iqsrv16** サーバオプション

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。

構文

```
-gr num
```

## **-gss iqsrv16** サーバオプション

サーバ接続要求を実行するサーバ実行スレッドのスタックサイズをある程度設定します。

構文

```
-gss { integer | integerK | integerM }
```

### 備考

SAP Sybase IQ は、これらのサーバスレッドのスタックサイズを、次の式を使用して計算します。(-gss + -iqtss)。 「-iqtss」を参照してください。

Windows の場合、-gss のデフォルト値は 4MB です。スタックサイズの最大値は 256MB です。

### 参照：

- -iqtss iqsrv16 サーバオプション (182 ページ)

## -gt iqsrv16 データベースサーバオプション

使用できる物理プロセッサの最大数を設定します (ライセンスされたプロセッサの数を上限とする)。このオプションは、マルチプロセッサシステムでのみ役立ちます。

### 構文

```
iqsrv16 -gt num-processors ...
```

### 指定可能な値

- **num-processors** - 1 と、次に示す項目の最小値との間の整数で値を設定できます。

コンピュータ上の物理プロセッサの数

サーバがライセンスを受けている CPU の最大数 (CPU がライセンスされている場合)

-gt オプションに範囲外の値を指定すると、下限または上限の値が設定されます。

### 備考

パーシットライセンスでは、ネットワークデータベースサーバは、コンピュータで使用可能なすべての CPU を使用します。CPU ベースのライセンスの場合、ネットワークデータベースサーバはライセンスを受けたプロセッサ数のみ使用可能です。ネットワークデータベースサーバが使用できる CPU 数は、SAP Sybase IQ のエディションによって制限される場合もあります。

-gt オプションの値を指定すると、データベースサーバはアフィニティマスクを調整し (ハードウェアプラットフォームでサポートされている場合)、指定した物理プロセッサの数のみを使用して実行するようにデータベースサーバを制限します。データベースサーバが  $n$  個のプロセッサのライセンスを受けている場合、デフォルトでは、サーバは  $n$  個の物理プロセッサにおいて、すべての論理プロセッサ (ハイパースレッドとコア) 上で実行します。-gtc オプションを使用すると、この動作にさらに制限を加えることができます。

## start\_iq データベースサーバ起動ユーティリティ

-gt オプションは、-gta オプションとは一緒に使用できません。

## -gtc iqsrv16 データベースサーバオプション

データベースサーバが許容するプロセッサ同時実行性の最大値を指定します。

### 構文

```
iqsrv16 -gtc logical-processors-to-use ...
```

### 適用対象

Linux、Solaris、Windows オペレーティングシステム (Intel 互換の x86 と x64 プラットフォームを実行するシステム)。

### 備考

データベースサーバを起動すると、検出された物理プロセッサと論理プロセッサの数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。

物理プロセッサは、コンピュータの CPU であり、**パッケージ**または**ダイ**と呼ばれることもあります。物理プロセッサがハイパースレッドをサポートする場合や、**マルチプロセッサ** (一般に「マルチコアプロセッサ」と呼ばれます) として設定されている場合は、追加の論理プロセッサが存在します。オペレーティングシステムは、論理プロセッサ上でスレッドをスケジュールします。

-gtc オプションでは、データベースサーバが使用する論理プロセッサの数を指定できます。このオプションを指定することで、サーバの起動時に作成されるデータベースサーバスレッドの数を制限します。これは、データベースサーバで同時に実行できるアクティブタスクの数を制限することになります。デフォルトでは、作成されるスレッドの数は (1 + ライセンスを受けているすべての物理プロセッサ上にある論理プロセッサの数) です。

デフォルトでは、ライセンスされた各物理プロセッサにおいて、すべての論理プロセッサ (コアまたはハイパースレッド) の同時使用が可能です。

たとえば、2 個の CPU を搭載し、各 CPU に 4 個のコアがあり、コアあたり 2 スレッドを使用できるシステムで、1 CPU ライセンスを使用する場合、ネットワークデータベースサーバでは、一方の CPU で 8 スレッドを同時に実行し、他方の CPU で 0 スレッドとすることができます。-gtc オプションが指定されている場合で、使用される論理プロセッサの数が、ライセンスされている物理プロセッサで利用できる総数よりも少なくなると、データベースサーバはラウンドロビン方式に基づいて論理プロセッサを割り付けます。-gtc オプションに対して暗黙的に 1 を指定すると、クエリ内並列処理 (クエリの並列処理) は無効になります。また、max\_query\_tasks オプションを使用して、クエリ内並列処理を明示的に制限したり無効にしたりすることもできます。

-gtc オプションは、-gta オプションとは一緒に使用できません。

## 例

Windows ベースの SMP コンピュータを例に挙げて考えてみます。ここでは、システムが 4 プロセッサ構成で、各物理プロセッサにコアが 2 つある (合計 8 つの論理プロセッサがある) ことを前提としています。物理プロセッサは文字で、論理プロセッサ (この例では、コア) は数字でそれぞれ区別します。このシステムには 4 プロセッサユニットとして A0、A1、B0、B1、C0、C1、D0、D1 が存在することになります。

シナリオ	ネットワークデータベースサーバの設定
シングル CPU ライセンス、または -gt に 1 が指定されている	-gt 1 -gtc 2 -gnh 20  スレッドは A0 または A1 で実行できます。
CPU のライセンスに制限がなく、-gtc に 5 が指定されている	-gt 4 -gtc 5 -gnh 20  スレッドは A0、A1、B0、C0、D0 で実行できます。
データベースサーバに 3 CPU ライセンスがあり、-gtc に 5 が指定されている	-gt 3 -gtc 5 -gnh 20  スレッドは A0、A1、B0、B1、C0 で実行できます。
CPU のライセンスに制限がなく、-gtc に 1 が指定されている	-gt 4 -gtc 1 -gnh 20  スレッドは A0 でのみ実行できます。

## -gu iqsrv16 データベースサーバオプション

データベースファイル管理文 (データベースの作成や削除などの文) の実行に必要な権限を設定します。

構文

```
iqsrv16 -gu { all | none | DBA | utility_db } ...
```

指定可能な値

-gu オプション	効果	適用対象
all	このオプションは推奨されなくなりました。すべてのユーザがファイル管理文を実行できます。	ユーティリティデータベースを含むすべてのデータベース
none	ファイル管理文の実行は許可されません。	ユーティリティデータベースを含むすべてのデータベース
DBA	SERVER OPERATOR システム権限を持つユーザのみがファイル管理文を実行できます。	ユーティリティデータベースを含むすべてのデータベース
utility_db	ユーティリティデータベースに接続できるユーザのみがファイル管理文を実行できます。	ユーティリティデータベースのみ

デフォルト

DBA

適用対象

すべてのオペレーティングシステムとデータベースサーバ。

備考

次のデータベースファイル管理文を実行できるユーザが制限を受けます。

- ALTER DATABASE dbfile ALTER TRANSACTION LOG
- CREATE DATABASE 文
- CREATE DECRYPTED DATABASE 文
- CREATE DECRYPTED FILE 文
- CREATE ENCRYPTED DATABASE 文
- CREATE ENCRYPTED FILE 文
- DROP DATABASE 文



- RESTORE DATABASE 文

utility\_db を指定した場合、これらの文はユーティリティデータベースからのみ実行できます。DBA を指定した場合、これらの文は SERVER OPERATOR システム権限を持つユーザのみ実行できます。none を指定した場合は、どのユーザもこれらの文を実行できません。

### 例

ファイル管理文の使用を防ぐため、-gu オプションの none 権限レベルを使用してデータベースサーバを起動します。次のコマンドは、データベースサーバを起動し、TestSrv という名前を付けます。このコマンドによって mytestdb.db データベースがロードされますが、どのユーザも、そのサーバを使用してデータベースを作成または削除したり、他のファイル管理文を実行したりすることはできません。これは、ユーザのリソース作成権の有無や、ユーティリティデータベースをロードして接続できるかどうかには関係ありません。

```
iqsrv16 -n TestSrv -gu none c:\mytestdb.db
```

ユーティリティデータベースのパスワードを知っているユーザだけにファイル管理文の実行を許可するには、次のコマンドを実行してサーバを起動します。

```
iqsrv16 -n TestSrv -su secret -gu utility_db
```

次のコマンドは、Interactive SQL をクライアントアプリケーションとして起動し、TestSrv という名前のサーバに接続し、ユーティリティデータベースをロードして、ユーザを接続させます。

```
dbisql -c
"UID=DBA;PWD=secret;DBN=utility_db;Host=host1;Server=TestSrv"
```

上記のコマンドが正常に実行されると、ユーザがユーティリティデータベースに接続し、ファイル管理文を実行できます。

## -iqfreq iqsrv16 サーバオプション

強制リカバリモードでデータベースを開きます。

構文

```
-iqfreq dbname
```

## -iqgovern iqsrv16 サーバオプション

サーバで同時に実行できるクエリの数を設定します。

構文

```
-iqgovern num
```

デフォルト

RECOVERY\_TIME データベースオプションの設定 (デフォルトでは 2 分)

### 備考

同時に実行できるクエリのは、接続の数とは異なります。1つの接続で複数のオープンカーソルを持つことができます。**-iqgovern** を指定することによって、SAP Sybase IQ でのディスクへのバッファデータのページングを最適化し、メモリの過剰使用を防止できます。このスイッチのデフォルト値は、マシンの CPU 数の 2 倍に 10 を加算した数です。場合によっては別の値、たとえば CPU 数の 2 倍に 4 を加算した数などを使うと、より高いスループットが得られることがあります。接続するユーザが多い場合、特にこの傾向があります。

## -iqlm iqsrv16 サーバオプション

SAP Sybase IQ が一時的に使用するためにオペレーティングシステムに対して動的に要求できるメモリの最大量を MB 単位で指定します。

### 構文

**-iqlm size**

### デフォルト

2,048 MB

### 備考

一部のロード操作では、デフォルトの 2 GB よりも多くのメモリが必要になることがあります。デフォルトのメモリ量よりも多くのメモリが必要な場合は、**-iqlm** 起動オプションを使用して、SAP Sybase IQ が OS に動的に要求できるメモリの量を増やします。**-iqlm** は、サーバを起動するコマンドまたは設定ファイルの一部として使用し、スイッチとして指定します。

原則として、ラージメモリ所要量は、SAP Sybase IQ に割り付けた利用可能な総物理メモリの 3 分の 1 を表します。IQ のメインストアとテンポラリストアに十分なメモリを確保するために、起動パラメータ **-iqlm**、**-iqtc**、および **-iqmc** の値をそれぞれ、SAP Sybase IQ に割り付けた利用可能な総物理メモリの 3 分の 1 の量に設定してください。

ほとんどの場合、SAP Sybase IQ プロセスがスワップアウトされないようにするため、総物理メモリの 80% を SAP Sybase IQ に割り付けます。同じシステム上で稼働する他のプロセスを考慮して、実際のメモリ割り付けを調整してください。たとえば、コア数が 32、利用可能な総物理メモリが 128 GB のマシンでは、100 GB (総メモリ 128 GB のおよそ 80%) を SAP Sybase IQ プロセスに割り付けます。上記の原則に従って、**-iqlm**、**-iqtc**、および **-iqmc** パラメータの値をそれぞれ 33 GB に設定します。

---

**注意：**サイズ値は、常に単位なしで指定します。たとえば、`-iqm 2048MB`ではなく、`-iqm 2048`と指定します。単位を指定した場合、`start_iq`はこのスイッチを無視します。

---

## **-iqmc iqsrv16 サーバオプション**

メイン IQ ストアのバッファキャッシュサイズを MB 単位で指定します。

構文

`-iqmc size`

デフォルト

64MB

備考

デフォルト値 64 MB を上書きするこのスイッチは、サーバが起動されてから停止されるまでの間に起動されたすべてのデータベースに適用されます。つまり、サーバの起動時にデータベースを 1 つ起動し、その後別のデータベースを 1 つ起動した場合、メインバッファキャッシュに `-iqmc` の 2 倍のサイズのキャッシュが必要になることとなります。

ラージメモリ所要量は、利用可能な総物理メモリの 3 分の 1 を表します。メインストアに十分なメモリを確保するために、`-iqmc` 起動パラメータの値を利用可能な総物理メモリの 3 分の 1 に設定してください。

サイズ値は、常に単位なしで指定します。たとえば、`-iqmc 32MB`ではなく、`-iqmc 32`と指定します。単位が必要な SQL Anywhere とは異なり、`start_iq`では、単位を指定するとこのスイッチは無視されます。

---

**注意：** SAP Sybase IQ サーバでは複数のデータベースを実行しないでください。

---

## **-iqmpx\_failover iqsrv16 サーバオプション**

マルチプレックスコーディネータフェールオーバーを初期化し、指定されたフェールオーバーセカンダリノードを新しいコーディネータとして確立します。

備考

このオプションを使用してコーディネータを起動しても影響はありません。セカンダリノードを実行するには、マルチプレックスグリッド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。iqmpx\_failoverの値については、『管理：マルチプレックス』を参照してください。

**start\_iq** データベースサーバ起動ユーティリティ

## **-iqmpx\_ov iqsrv16** サーバオプション

現在のノードのマルチプレックス設定の上書きを実行します。

### *備考*

ノードの場所またはその他のプロパティが変更された場合、起動時にノードのプロパティを変更するために使用されます。セカンダリノードを実行するには、マルチプレックスグリッドオプションのライセンスを取得する必要があります。iqmpx\_ov の値については、『管理：マルチプレックス』を参照してください。

## **-iqmpx\_reclaimwriterfreelist iqsrv16** サーバオプション

このオプションは、コーディネータノードを再起動している間のみ適用されません。

### *備考*

コーディネータは、サーバ名で識別されているライターノードのフリーリストを強制的に要求します。このスイッチは、ライターに障害が発生し、再起動できない場合にのみ必要となります。セカンダリノードを実行するには、マルチプレックスグリッドオプションのライセンスを取得する必要があります。

iqmpx\_reclaimwriterfreelist の値については、『管理：マルチプレックス』を参照してください。

## **-iqmpx\_sn iqsrv16** サーバオプション

現在のマルチプレックスノードをシングルノードモードで実行します。

### *備考*

このモードは、マルチプレックス設定での問題を解決するためのみに使用されません。使用の際には、十分な注意を払う必要があります。マルチプレックス内の他のすべてのノードを停止する必要があります。コーディネータノードのみで使用することをおすすめします。セカンダリノードを実行するには、マルチプレックスグリッドオプションのライセンスを取得する必要があります。iqmpx\_sn の値については、『管理：マルチプレックス』を参照してください。

## **-iqmsgnum iqsrv16** サーバオプション

サーバで維持する古いメッセージログのアーカイブ数を指定します。

### *構文*

**-iqmsgnum***num*

**備考**

使用できる値は整数 0 ~ 64 です。デフォルトは 0 で、メッセージはメインのメッセージログファイル内でラッピングされます。**-iqmsgsz** または **IQMsgMaxSize** サーバプロパティの値が 0 以外の場合にのみ有効になります。**IQMsgNumFiles** サーバプロパティは **-iqmsgnum** に相当し、**-iqmsgnum** の値より優先されます。

0 よりも大きい **-iqmsgnum** 値  $n$  は、サーバで  $n$  個のメッセージログアーカイブを維持することを意味します。たとえば、次のコマンドではサーバがメッセージログの 3 つのアーカイブを保持することを指定します。

```
start_iq -n iqdemo iqdemo.db ... <other options> ... -iqmsgsz 100 -iqmsgnum 3
```

**-iqmsgsz iqsrv16 サーバオプション**

メッセージログの最大サイズを制限します。

**構文**

**-iqmsgsz** *size*

**備考**

**-iqmsgsz** は 0 ~ 2047 の整数です (メガバイト)。デフォルト値は 0 で、メッセージログサイズの制限がないことを意味します。**IQMsgMaxSize** サーバプロパティは **-iqmsgsz** サーバスイッチに相当し、**-iqmsgsz** の値より優先されます。

0 よりも大きい **-iqmsgsz** 値  $n$  は、メッセージログが最大  $n$  メガバイトまで大きくなることを意味します。たとえば、次のコマンドはメッセージログのサイズを 100MB に制限します。

```
start_iq -n iqdemo iqdemo.db ... <other options> ... -iqmsgsz 100
```

**-iqmt iqsrv16 サーバオプション**

作成する SAP Sybase IQ スレッドの数を指定します。

**構文**

**-iqmt** *num*

**備考**

デフォルトは、最初の 4 CPU に対して CPU ごとに 60 スレッド、残りの CPU に対して CPU ごとに 50 スレッド、それにシステムで使用するための 3 スレッドとデータベース接続およびバックグラウンドタスクに必要なスレッドを加えた数です。たとえば、CPU が 12、接続スレッドが 10 のシステムの場合は、 $60 * 4 +$

$50 * (\text{numCPUs} - 4) + \text{numConnections} + 6 = 656$  と計算されます。

## start\_iq データベースサーバ起動ユーティリティ

num の最小値は num\_conn + 3 です。

サーバスレッドの総数は、4096 を超えてはいけません。

## -iqnotemp iqsrv16 サーバオプション

定義されたテンポラリ DB 領域を使用せずにテンポラリファイルを作成します。

### 構文

**-iqnotemp size**

### 備考

*size* はファイルサイズ (MB) です。このパラメータを指定すると、サーバはデータベースの起動時に SAP Sybase IQ のすべてのテンポラリ dbfile 定義を無視します。-**iqnotemp** を使用することで、破損したファイルを削除し、後で置き換えることができるようになり、テンポラリ dbfile の問題を解決できます。

リストア時に **-iqnotemp** を指定してユーティリティデータベースサーバを起動すると、リストア操作時に (中間起動を含む)、バックアップデータベースのすべてのテンポラリファイル定義が SAP Sybase IQ によって無視されます。このため、古いテンポラリファイル定義を再作成して使用することなく、異なるテンポラリファイルトポロジにデータベースをリストアできます。

**-iqnotemp** を指定して起動されたデータベースで許可されるテンポラリファイル操作は、**ALTER DBSPACE IQ\_SYSTEM\_TEMP DROP FILE** のみです。

## -iqnumbercpus iqsrv16 サーバオプション

リソース計画を目的として、SAP Sybase IQ で使用できる CPU 数を指定して、物理的な CPU 数を上書きします。

### 構文

**-iqnumbercpus num**

### 備考

**-iqnumbercpus** のデフォルト値は CPU の総数ですが、指定できる値は 1 ~ 128 の範囲です。

**-iqnumbercpus** は、次のマシンのみで使用してください。

- Intel の CPU を搭載し、ハイパースレッディングが有効になっているマシンで、**-iqnumbercpus** が利用可能な CPU コア数に設定されている。
- オペレーティングシステムのユーティリティを使って、SAP Sybase IQ で使用可能な CPU がマシンにある CPU の一部に制限されているマシン。

-**iqnumbercpus** を、使用可能な CPU 数より大きな値に設定すると、パフォーマンスに影響する可能性があります。

## -iqpartition iqsrv16 サーバオプション

IQ メインバッファキャッシュと一時バッファキャッシュのパーティション数を指定します。

構文

**-iqpartition** *num*

備考

IQ メインバッファキャッシュと一時バッファキャッシュのパーティション数を指定します。2 の累乗の値を指定する必要があります。デフォルトでは、指定可能な値は、0 (デフォルト)、1、2、4、8、16、32、64、128、256 です。デフォルトでは、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。キャッシュパーティションの数を調整することによって、パフォーマンスが向上することがあります。-

**iqpartition** スイッチは、この値を SAP Sybase IQ サーバに設定し、`Cache_Partitions` データベースオプションで設定された値を上書きします。

- jConnect™ for JDBC™ システムオブジェクトをデータから除外する。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するには、jConnect カタログサポートが必要 (デフォルトでインストールされる)。このオプションを指定した場合でも、システム情報にアクセスしないかぎり、JDBC を使用できる。jConnect サポートは、**ALTER DATABASE** 文を使用して後から追加できる。

## -iqrlvmem start\_iq サーバオプション

RLV ストアで使用可能なメモリ量をメガバイト単位で指定します。

構文

**-iqrlvmem** *size*

デフォルト

2048 (メガバイト)

備考

0 または無効な値を指定すると、デフォルト (2048 MB) が使用されます。値がシステム仮想メモリの上限の 2/3 を超えると、エラーメッセージが表示され、サーバが停止します。

### 使用法

**-iqrlvmem** は、ローレベルのバージョン管理用に確保するメモリ量をサーバに指示する目的で、起動時に使用されます。

## -iqro iqsrv16 サーバオプション

現在のノードを読み込み専用で実行します。

### 備考

シンプレックスおよびマルチプレックスの移行を検証するためと、従来のデータベースのアンロードで使用します。常に、このオプションの引数としてブール値 1 (唯一の有効なパラメータ) を指定します。

## -iqstart iqsrv16 サーバオプション

DB 領域に関する起動時診断を実行します。

### 構文

**-iqstart** *N*

### 備考

入力パラメータ *N* は、整数ビットマスクを表す数値です。値を組み合わせ、複数の機能を実行することもできます。IQ メッセージファイルが生成される前に生成された出力は、コンソールに送られます。-z 起動スイッチは、起動と接続に関する追加情報を提供します。

使用できる値を次に示します。

- N=1 – ファイル名に関する基本情報を SYSIQFILES から返します。この情報は、DB 領域を開くときに使用されます。その後、使用される完全修飾名を表示します。このオプションを使用すると、データベースで使用されているファイルのレコードを IQ メッセージファイル内に作成できます。
- N=2 – トランザクションログをリプレイした後、RecoveryComplete を実行する前に停止します。これにより、データベースを完全に開けずに調べることができます。N=2 はその他のオプションと組み合わせることができます。N=2 は、特定のモードでは commit\_identity を書き換えますが、それ以外の場合はデータベースを永久的に変更することはありません。これは、リカバリ処理をコミットするチェックポイントを完了できないためです。すべてのリカバリアクションは、次回データベースを開いたときに再実行されます。
- N=4 – SYSIQFILE のすべてのロー、データベースがマルチプレックスデータベースの場合に選択されているファイル名のサブセット、完全に解決されたファイル名、DB 領域ファイルの個々のヘッダブロック、database\_identity、



commit\_identity、チェックポイントの各ログエントリ、トランザクションの各ログエントリを含む、完全な診断情報を返します。

- N=8 – SYSIQFILE 内のファイルパスを上書きできるようにします。  
SYSIQFILE 値の代わりに、iqmsg.iqmsg, iqmain\_1, iqmain\_2, ..., iqtemp\_1, iqtemp\_2, などのファイル名が使用されます。これらにはリンクを使用できます。また、これらは .db ファイルと同じディレクトリ内に存在する必要があります。実際の .db ファイルへのリンクを使用できますが、サーバにリンクが指定されているときに、リンク先の .db ファイルがデータベースからの相対位置にあるトランザクションログを使用する場合、サーバはデータベースではなくリンクを基準にトランザクションログを検索する点に注意してください。この場合、トランザクションログのリンクも作成します。

### -iqtc iqsrv16 サーバオプション

IQ テンポラリストアのキャッシュサイズを MB で指定します。

構文

**-iqtc size**

デフォルト

64MB

備考

デフォルト値 64 MB を上書きするこのスイッチは、SAP Sybase IQ サーバが起動されてから停止されるまでの間に起動されたすべてのデータベースに適用されます。つまり、サーバの起動時にデータベースを 1 つ起動し、その後別のデータベースを 1 つ起動した場合、テンポラリキャッシュに **-iqtc** の 2 倍のサイズのキャッシュが必要になることとなります。通常、1 つの SAP Sybase IQ サーバで複数のデータベースを実行することは避けてください。

ラージメモリ所要量は、利用可能な総物理メモリの 3 分の 1 を表します。IQ テンポラリストアキャッシュに十分なメモリを確保するために、**-iqtc** 起動パラメータの値を利用可能な総物理メモリの 3 分の 1 に設定してください。

サイズ値は、常に単位なしで指定します。たとえば、**-iqtc 32MB** ではなく、**-iqtc 32** と指定します。単位が必要な SQL Anywhere とは異なり、**start\_iq** では、単位を指定するとこのスイッチは無視されます。

IQ\_SYSTEM\_TEMP DB 領域は、必ず **-iqtc** 以上に設定してください。

**start\_iq** データベースサーバ起動ユーティリティ

## **-iqtss iqsrv16** サーバオプション

バックグラウンドで実行されている、またはメインサーバ接続スレッドを支援するスレッドチームの一部として実行されている、サーバ実行スレッドのスタックサイズを KB 単位で指定します。

構文  
**-iqtss** *size*

備考  
64 ビットプラットフォームのデフォルトは 512KB、32 ビットプラットフォームの場合は 200KB です。

参照：

- `-gss iqsrv16` サーバオプション (168 ページ)

## **-iqwmem iqsrv16** サーバオプション

HP および Sun の UNIX システムで使用される「連結」メモリプールを作成しません。

構文  
**-iqwmem** *size*

備考  
このメモリはロックされているため、オペレーティングシステムによるページングはできません。このスイッチは、連結メモリを追加するためのメモリを確保できる場合にのみ使用してください。メモリが十分にないときにこのスイッチを使用すると、パフォーマンスが著しく低下することがあります。

## **-k iqsrv16** データベースサーバオプション

パフォーマンスモニタ統計値の収集を制御します。

構文  
`iqsrv16 -k ...`

デフォルト  
パフォーマンスモニタ統計値が収集されます

適用対象  
すべてのオペレーティングシステムとデータベースサーバ

### 備考

データベースサーバの起動時に `-k` を指定すると、パフォーマンスモニタ統計値は収集されません。`-k` オプションは、クエリオプティマイザが使用するカラム統計には影響しません。

このオプションは、データベースサーバをマルチプロセッサコンピュータで実行しており、テストによってパフォーマンスの向上が確認できた場合のみ使用してください。通常の負荷レベルでは、パフォーマンスの向上はわずかであるため、このオプションの使用は推奨できません。パフォーマンスカウンタを無効にすると、パフォーマンス上の問題が発生した場合、分析に必要な情報を取得できません。

パフォーマンスモニタ統計値の収集についての設定は `sa_server_option` システムプロシージャでも変更できます。

## **-kl iqsrv16 データベースサーバオプション**

Kerberos GSS-API ライブラリ (UNIX では共有オブジェクト) のファイル名を指定し、データベースサーバへの Kerberos 認証接続を有効にします。

### 構文

```
iqsrv16 -kl GSS-API-library-file ...
```

### 適用対象

すべてのオペレーティングシステム。

### 備考

このオプションでは、Kerberos GSS-API のロケーションと名前を指定します。このオプションが必要となるのは、Kerberos クライアントでデフォルトと異なる Kerberos GSS-API ライブラリファイル名が使用されているか、データベースサーバを実行しているコンピュータに複数の GSS-API ライブラリがインストールされている場合だけです。Kerberos クライアントのインストールと設定が完了し、データベースサーバが SSPI を使用できない状態であることが必要です。

このオプションを指定すると、データベースサーバに対する Kerberos 認証が有効になります。

### 例

次のコマンドは、Kerberos 認証に `libgssapi_krb5.so` 共有オブジェクトを使用するデータベースサーバを起動します。

```
iqsrv16 -kl libgssapi_krb5.so -n my_server_princ /opt/myapp/kerberos.db
```

## **-kr iqsrv16 データベースサーバオプション (廃止予定)**

Kerberos サーバプリンシパルの領域を指定し、データベースサーバへの Kerberos 認証接続を有効にします。

---

**注意：** -kr オプションを使用する方法は廃止されます。-kp オプションを使用して Kerberos サーバプリンシパルを指定することをおすすめします。-kp を指定するときに、サーバプリンシパルは、データベースサーバを実行しているコンピュータの Kerberos keytab ファイルに抽出されている必要があります。

---

### 構文

```
iqsrv16 -kr server-realm ...
```

### 適用対象

すべてのオペレーティングシステム。

### 備考

このオプションでは、Kerberos サーバプリンシパルの領域を指定します。通常は、データベースサーバが Kerberos 認証に使用するプリンシパルは *server-name@default-realm* です。 *default-realm* は Kerberos クライアントに設定されたデフォルト領域です。このオプションは、サーバプリンシパルの領域をデフォルト領域以外に変更する場合に使用します。このオプションを指定すると、サーバプリンシパルは *server-name@server-realm* となります。

このオプションを指定すると、データベースサーバに対する Kerberos 認証が有効になります。

-kp オプションが指定されている場合は、-kr オプションは指定できません。

### 例

次のコマンドは、Kerberos ログインを受け入れて認証にプリンシパル *my\_server\_princ@MYREALM* を使用するデータベースサーバを起動します。

```
iqsrv16 -kr MYREALM -n my_server_princ C:¥kerberos.db
```

## **-krb iqsrv16 データベースサーバオプション**

データベースサーバへの Kerberos 認証接続を有効にします。

### 構文

```
iqsrv16 -krb ...
```

### 適用対象

すべてのオペレーティングシステム。

### 備考

このオプションは、データベースサーバに対する Kerberos 認証を有効にします。データベースサーバが Kerberos を使用してクライアントを認証するためには、-krb、-kl、-kr のうち 1 つ以上を指定する必要があります。

Kerberos 認証を使用するためには、クライアントコンピュータとデータベースサーバコンピュータの両方に Kerberos クライアントがインストールされ、設定が完了している必要があります。さらに、Kerberos KDC にプリンシパル `server-name@REALM` があり、プリンシパル `server-name@REALM` のキータブがデータベースサーバコンピュータ上の keytab ファイルに抽出されていることも必要です。この準備が完了していない場合、-krb オプションを指定すると、データベースサーバは起動しません。

---

**注意：** データベースサーバ名には、/、¥、@ のいずれの文字も含めることはできません。また、マルチバイト文字が含まれているデータベースサーバ名を Kerberos で使用することはできません。

---

Kerberos ログインを許可するためには login\_mode データベースオプションを設定する必要があります。また、GRANT KERBEROS LOGIN 文を使用して、Kerberos クライアントプリンシパルをデータベースユーザ ID にマッピングすることも必要です。

### 例

次のコマンドは、データベースサーバの Kerberos プリンシパル `my_server_princ@MYREALM` に対し、データベースサーバ `my_server_princ` を起動します。

```
iqsrv16 -krb -n my_server_princ C:¥kerberos.db
```

## **-ks iqsrv16 データベースサーバオプション**

データベースサーバからカウンタ値を収集するためにパフォーマンスモニタで使用される共有メモリの作成を無効にします。

### 構文

```
iqsrv16 -ks 0 ...
```

### 適用対象

Windows

### 備考

このオプションを指定すると、パフォーマンスモニタには現在のデータベースサーバのデータベースサーバ名、データベース名、接続統計は表示されません。

**start\_iq** データベースサーバ起動ユーティリティ

## **-ksc iqsrv16** データベースサーバオプション

パフォーマンスモニタでモニタできる接続の最大数を指定します。

構文

```
iqsrv16 -ksc integer ...
```

デフォルト

10

適用対象

Windows

## **-ksd iqsrv16** データベースサーバオプション

パフォーマンスモニタでモニタできるデータベースの最大数を指定します。

構文

```
iqsrv16 -ksd integer ...
```

デフォルト

2

適用対象

Windows

## **-m iqsrv16** データベースサーバオプション

チェックポイントが終了すると、トランザクションログをトランケートします。

構文

```
iqsrv16 -m ...
```

適用対象

すべてのオペレーティングシステムとデータベースサーバ

備考

このオプションを指定すると、シャットダウン時、またはサーバでスケジュールされたチェックポイントの結果としてチェックポイントが実行されたときに、トランザクションログをトランケートします。

---

### **警告！ 警告**

このオプションを選択すると、データベースファイルを含むデバイスのメディア障害に対して無防備な状態になります。

---

このオプションを指定すると、トランザクションログの肥大化が自動的に制限されます。チェックポイントの頻度は、`checkpoint_time` と `recovery_time` オプションによって制御できます (また、コマンドラインでも設定できます)。

-m オプションは、高速な応答時間を必要とする大容量のトランザクションを処理する場合や、リカバリやレプリケーションがトランザクションログの内容に依存しない場合に、トランザクションログのサイズを制限するのに役立ちます。-m オプションは、各 COMMIT の後にチェックポイントが必要で、その結果パフォーマンスが低下するような場合に、トランザクションログなしで稼働する場合の代替策となります。-m オプションを指定すると、データベースファイルを含むデバイスのメディア障害に対して無防備な状態になります。-m オプションを使用する前に、トランザクションログを管理する他の代替策 (BACKUP 文やイベントの使用など) を検討してください。

データベースファイルの断片化を防ぐためには、このオプションを使用する場合には、トランザクションログをデータベースそのものとは別のデバイスまたはパーティションに保管することをおすすめします。

このオプションを指定すると、チェックポイントの実行中に他の操作は行われません。

### **-n iqsrv16 データベースサーバオプション**

データベースサーバの名前を設定します。

#### 構文

```
iqsrv16 -n server-name database-filename ...
```

#### デフォルト

データベースサーバで最初に起動したのデータベースファイルの名前 (パスと拡張子を除く)。

#### 適用対象

すべてのオペレーティングシステムとデータベースサーバ。

#### 備考

データベースサーバは起動するとき、そのコンピュータのデフォルトのデータベースサーバになろうとします。デフォルトのサーバがない場合、最初に起動したデータベースサーバが、デフォルトのデータベースサーバになります。そのコンピュータで、データベースサーバ名を明示的に指定しないで共有メモリに接続しようとする、デフォルトのサーバに接続されます。

---

**注意:** 配備されたアプリケーションで使用されているデータベースサーバには -xd オプションを使用すること、またすべてのクライアントが、ServerName (Server) 接続パラメータを使用して、接続先のデータベースサーバ名を明示的に指定する

## start\_iq データベースサーバ起動ユーティリティ

ことをおすすめします。このようにすると、コンピュータで複数のデータベースサーバが実行されているときに、データベースが正しいデータベースサーバに接続します。

---

サーバ名に対して文字セット変換は実行されません。クライアントとデータベースサーバとで文字セットが異なる場合、サーバ名に拡張文字が使用されているとサーバが見つからないことがあります。クライアントとサーバを異なるオペレーティングシステムやロケールで実行している場合は、サーバ名に7ビットのASCII文字を使用してください。

データベースサーバの名前は、有効な識別子であることが必要です。データベースサーバのロングネームは、プロトコルごとに異なる長さにtruncateされます。データベースサーバ名には、次に該当する値を指定できません。

空白スペース、一重引用符、または二重引用符で始まる値

空白スペースで終わる値

セミコロン、フォワードスラッシュ (/)、またはバックスラッシュ (\) を含む値

長さが 250 バイトを超える値

UNIX 上で実行するときにスペースを含む値

---

**注意：** Windows と UNIX では、データベースサーバがバージョン 10.0.0 以降で、名前が次の長さを超えている場合、バージョン 9.0.2 以前のクライアントから接続することはできません。

Windows 共有メモリの場合は、40 バイト

UNIX 共有メモリの場合は、31 バイト

TCP/IP の場合は、40 バイト

---

サーバ名は、クライアントアプリケーション接続文字列かプロファイルの ServerName (Server) 接続パラメータで使用する名前を指定します。共有メモリ環境では、-xd を指定しないかぎり、サーバ名が指定されておらず少なくとも1つのデータベースサーバがコンピュータで実行されている場合に使用されるデフォルトのデータベースサーバがあります。

同じ名前でも複数のデータベースサーバを実行することはおすすめしません。

---

**注意：** 2つの -n オプションがあります。-n オプションは指定する位置によって意味が異なります。データベースファイル名の前に指定すると、サーバオプションとしてサーバ名を指定します。データベースファイル名の後に指定すると、データベースオプションとしてデータベース名を指定します。

たとえば、次のコマンドでは、データベースサーバ名 **SERV** とデータベース名 **DATA** が指定されます。

```
iqsrv16 -n SERV sales.db -n DATA
```

---



## データベースサーバの命名制限

**start\_iq [server-options]** で **-n** スイッチを使用すると、特定の命名制限が適用されません。

サーバ名では文字セットの変換が実行されません。クライアントの文字セットとデータベースサーバの文字セットが異なる場合、サーバ名で拡張文字を使用すると、サーバが見つからなくなることがあります。クライアントとサーバが異なるオペレーティングシステムまたはロケールで実行されている場合、サーバ名には7ビット ASCII 文字を使用します。

データベースサーバ名は、有効な識別子にしてください。長いデータベースサーバ名は、プロトコルに応じてさまざまな長さにトランケートされます。データベースサーバ名では次の操作を行うことができません。

- 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。
- 128 バイトを超える。

サーバ名は、クライアントアプリケーション接続文字列またはプロファイルで使用する名前を指定します。同じ名前のデータベースサーバを複数実行することはおすすめしません。

### 参照：

- start\_iq のサーバオプション (146 ページ)

## -o iqsrv16 データベースサーバオプション

すべてのデータベースサーバメッセージをデータベースサーバメッセージログファイルに出力します。

### 構文

```
iqsrv16 -o filename ...
```

### 適用対象

すべてのオペレーティングシステムとデータベースサーバ。

### 備考

情報メッセージ、エラー、警告、MESSAGE 文の出力を含むすべてのデータベースサーバメッセージを、指定したファイルとデータベースサーバメッセージウィンドウに出力します。-o オプションとともに -qi オプションを指定すると、すべてのメッセージがデータベースサーバメッセージログファイルだけに出力されます。

トランザクションログを使用した操作を実行するユーティリティで問題が発生する可能性があるため、ファイル名の最後には .log を付けないようにしてください。

## **start\_iq** データベースサーバ起動ユーティリティ

データベースサーバメッセージログのファイル名を検索するには、次のコマンドを実行します。

```
SELECT PROPERTY ( 'ConsoleLogFile' );
```

### **start\_iq** エラーレポート

サーバの起動で問題がある場合、**start\_iq** は 0 以外の値を返します。

起動時に **-o** スイッチの後ろにログファイルを指定しなかった場合は、以下の中で最初に定義されているものにエラーが書き込まれます。

- (Windows) %ALLUSERSPROFILE%¥SybaseIQ¥16\_0¥logfiles¥
- (UNIX/Linux) \$IQDIR16/logfiles/

## **-oe iqsrv16** データベースサーバオプション

起動エラー、致命的なエラー、アサーションをロギングするファイルの名前を指定します。

### 構文

```
iqsrv16 -oe filename ...
```

### 適用対象

すべてのオペレーティングシステムとデータベースサーバ

### 備考

出力ログファイルの各行の先頭には、日付と時刻が記録されます。起動エラーには、次のようなエラーがあります。

- ファイルを開くことも読むこともできません : *database file*.
- その名前のデータベースサーバはすでに起動しています。

致命的なエラーとアサーションは、**-oe** が指定されているかどうかにかかわらず、Windows アプリケーションイベントログまたは UNIX システムログに記録されません。

トランザクションログを使用した操作を実行するユーティリティで問題が発生する場合があるため、ファイル名の最後には `.log` を付けないようにしてください。

## **-on iqsrv16** データベースサーバオプション

データベースサーバメッセージログファイルの最大サイズを指定します。ログファイルがこのサイズに達すると、現在のファイルが拡張子 `.old` の付いた名前に変更され、新しいファイルが作成されます。

### 構文

```
iqsrv16 -on { size[ k | m | g ] } ...
```

### 適用対象

すべてのオペレーティングシステムとデータベースサーバ。

### 備考

*size* には、データベースサーバメッセージログの最大サイズをバイト単位で指定します。単位をキロバイト、メガバイト、またはギガバイトで指定するには、それぞれ *k*、*m*、*g* のいずれかを使用してください。最小のサイズ制限は 10 KB です。デフォルトでは、最大サイズは無制限となります。

データベースサーバメッセージログが指定されたサイズに達すると、データベースサーバによって現在のファイルが拡張子 `.old` の付いた名前に変更され、元の名前を持つ新しいファイルが開始されます。

---

**注意：** `.old` データベースサーバメッセージログファイルがすでに存在する場合は、そのファイルが上書きされます。古いデータベースサーバメッセージログファイルを削除しないようにするには、代わりに `-os` オプションを使用します。

---

このオプションは、`-os` オプションと同時に使用できません。

トランザクションログを使用した操作を実行するユーティリティで問題が発生する可能性があるため、データベースサーバメッセージログファイル名の最後には `.log` を付けないようにしてください。

## **-os iqsrv16** データベースサーバオプション

データベースサーバメッセージログファイルの最大サイズを指定します (このサイズに達するとログのファイル名が変更されます)。

### 構文

```
iqsrv16 -os { size[ k | m | g ] } ...
```

### 適用対象

すべてのオペレーティングシステムとデータベースサーバ。

### 備考

*size* には、データベースサーバメッセージのログを取るファイルの最大サイズをバイト単位で指定します。単位をキロバイト、メガバイト、またはギガバイトで指定するには、それぞれ *k*、*m*、*g* のいずれかを使用してください。最小のサイズ制限は 10 KB です。デフォルトでは、最大サイズは無制限となります。

データベースサーバは、出力メッセージをデータベースサーバメッセージログファイルに書き込む前に、現在のファイルサイズを確認します。新しいログメッセージを書き込むと、指定されたファイルサイズを超える場合は、データベースサーバメッセージログのファイル名が `yyymmddxx.slg` に変更されます。`yyymmdd`

## **start\_iq** データベースサーバ起動ユーティリティ

は、そのファイルが作成された年、月、日を表します。xxは00で始まる番号で、1ずつ増えていきます。

このオプションによって、データベースサーバメッセージログファイルが古いことを確認して削除し、ディスク領域を解放できます。

このオプションは、-on オプションとは一緒に使用できません。

トランザクションログを使用した操作を実行するユーティリティで問題が発生する可能性があるため、データベースサーバメッセージログファイル名の最後には .log を付けないようにしてください。

## **-ot iqsrv16** データベースサーバオプション

データベースサーバメッセージログファイルをトランケートし、そのファイルに出力メッセージを追加します。

### 構文

```
iqsrv16 -ot logfile ...
```

### 適用対象

すべてのオペレーティングシステムとデータベースサーバ

### 備考

このオプションの機能は、データベースサーバメッセージログファイルのトランケート後にメッセージが書き込まれる点を除き、-o オプションと同じです。データベースサーバメッセージログのファイル名を検索するには、次のコマンドを実行します。

```
SELECT PROPERTY ( 'ConsoleLogFile' );
```

トランザクションログを使用した操作を実行するユーティリティで問題が発生する可能性があるため、データベースサーバメッセージログファイル名の最後には .log を付けないようにしてください。

## **-p iqsrv16** データベースサーバオプション

通信パケットの最大サイズを設定します。

### 構文

```
iqsrv16 -p integer ...
```

### デフォルト

7300 バイト

### 適用対象

すべてのオペレーティングシステムとデータベースサーバ。

### 備考

最小値は 500 バイトで、最大値は 65535 バイトです。

接続の通信バッファサイズを変更するには、CommBufferSize (CBSIZE) 接続パラメータを設定します。

## -pc iqsrv16 データベースサーバオプション

同一コンピュータ接続以外のすべての接続のパケットを圧縮します。

### 構文

```
iqsrv16 -pc ...
```

### 適用対象

すべてのオペレーティングシステムとネットワークサーバ (Web サーバを除く)

### 備考

クライアントとサーバの間で送信されるパケットは、-pc オプションを使用して圧縮できます。状況によっては、接続を圧縮することでパフォーマンスが向上する場合があります。大幅に圧縮可能なデータの大規模なデータ転送では、圧縮率が高くなります。クライアントの接続パラメータに COMPRESS=NO を指定すると、特定のクライアントについてこのオプションを上書きできます。

デフォルトでは、接続は圧縮されません。-pc オプションを指定すると、同一コンピュータ接続、Web サービス接続、および TDS 接続を除くすべての接続のパケットが圧縮されます。TDS 接続 (jConnect を含む) では、通信圧縮はサポートされません。

どの通信リンクを使用している場合でも、同一コンピュータ接続は -pc オプションまたは COMPRESS=YES 接続パラメータを使用しても圧縮されません。

## -pt iqsrv16 データベースサーバオプション

パケットの圧縮が適用される最小パケットサイズを増減します。

### 構文

```
iqsrv16 -pt size ...
```

### デフォルト

120 バイト

### 適用対象

すべてのオペレーティングシステムとネットワークサーバ

## start\_iq データベースサーバ起動ユーティリティ

### 備考

このパラメータには、圧縮が適用されるパケットサイズの最小バイト数を表す整数値を指定します。80 未満の値はおすすめしません。

状況によっては、圧縮のスレッシュホールドを変更すると、パケットの転送速度が上昇する場合のみパケットを圧縮できるようになり、圧縮された接続のパフォーマンスが向上することがあります。ほとんどの場合、デフォルト設定が適しています。

クライアントとサーバで圧縮スレッシュホールドの設定が異なる場合は、クライアントの設定が適用されます。

## -qi iqsrv16 データベースサーバオプション

データベースサーバシステムトレイアイコンとデータベースサーバメッセージウインドウを表示するかどうかを制御します。

### 構文

```
iqsrv16 -qi ...
```

### 適用対象

Windows

### 備考

このオプションは、起動エラーウインドウを除いて、サーバの実行中に視覚的な表示が出ないようにします。-o または -oe で指定したログファイルのいずれか (または両方) を使用してエラーを診断できます。

## -qp iqsrv16 データベースサーバオプション

データベースサーバメッセージウインドウにパフォーマンスに関するメッセージを表示しないことを指定します。

### 構文

```
iqsrv16 -qp ...
```

### 適用対象

すべてのオペレーティングシステムとデータベースサーバ

### 備考

データベースサーバメッセージウインドウにパフォーマンスに関するメッセージを表示ないようにします。表示されなくなるメッセージには次のものがあります。

- テーブル "*table-name*" にユニークなインデックスまたはプライマリーキーがありません。
- データベースファイル "*mydatabase.db*" は *nnn* のディスクフラグメントで構成されています。

## **-qs iqsrv16 データベースサーバオプション**

起動エラーウィンドウを表示しないようにします。

### 構文

```
iqsrv16 -qs ...
```

### 適用対象

Windows

### 備考

このオプションを指定すると、起動エラーウィンドウが表示されなくなります。起動エラーには、たとえばデータベースサーバでデータベースファイルを開いたり、読み取ったりできない場合や、指定された名前の別のデータベースサーバがすでに実行中なのでデータベースサーバが起動しない場合などがあります。

Windows プラットフォームでは、サーバが自動的に起動しない場合、これらのエラーがウィンドウに表示されるので、これらをクリアしてからサーバを停止する必要があります。これらのウィンドウは `-qs` オプションを使用すると表示されません。

言語 DLL のロード時のエラーの場合、`-qs` がコマンドラインで指定されていても `@data` 構文に指定されていないとウィンドウは表示されません。このエラーは `-o` または `-oe` で指定されたログには記録されず、Windows アプリケーションイベントログに記録されます。

`-qs` がコマンドラインで指定されていても、`@data` 拡張で指定されていない場合、使用法エラーは表示されません。

## **-qw iqsrv16 データベースサーバオプション**

データベースサーバメッセージウィンドウを表示しないことを指定します。

### 構文

```
iqsrv16 -qw ...
```

### 適用対象

すべてのオペレーティングシステムとデータベースサーバ

## start\_iq データベースサーバ起動ユーティリティ

### 備考

このオプションを指定すると、データベースサーバメッセージウィンドウが表示されなくなります。Windows プラットフォームでは、データベースサーバシステムトレイアイコンは表示されます。-o または -oe で指定したログファイルのいずれか (または両方) を使用してエラーを診断できます。

## **-s iqsrv16** データベースサーバオプション

Syslog メッセージのユーザ ID を設定します。

### 構文

```
iqsrv16 -s { none | user | daemon | localn } ...
```

### 適用対象

UNIX

### 備考

Syslog 機能へのメッセージに使用されるシステムユーザ ID を設定します。フォアグラウンドで起動しているデータベースサーバのデフォルトは user で、バックグラウンドで起動しているサーバ (dbspawn で起動した場合、クライアントが自動的に起動した場合、-ud データベースサーバオプションで起動した場合など) のデフォルトは daemon です。

none を指定すると、Syslog メッセージはログに記録されません。引数 localn を使用すると、機能識別子を使用してメッセージをファイルヘリダイレクトできます。n に 0～7 の数字を指定できます。詳細については、UNIX Syslog(3) の man ページを参照してください。

## **-sb iqsrv16** データベースサーバオプション

ブロードキャストに対するデータベースサーバの動作を指定します。

### 構文

```
iqsrv16 -sb { 0 | 1 } ...
```

### 適用対象

TCP/IP

### 備考

データベースサーバは、次の3種類のブロードキャストに応答できるように、1つまたは複数の UDP リスナを起動します。

1. このサーバを探しているクライアント接続ブロードキャスト。



2. データベースサーバ列挙ブロードキャスト (dblocate ユーティリティからのブロードキャストや管理ツールの[サーバ検索ウィザード] ([接続] ウィンドウ) からのブロードキャストなど)。
3. 同じ名前を持つすべてのデータベースサーバを探している別のデータベースサーバから送信されたブロードキャスト。

Embedded SQL 接続に対して、dblocate ユーティリティと db\_locate\_servers 関数によって、ブロードキャストパケットはすべてのデータベースサーバを検索する際にローカルネットワーク上に送出されます。データベースサーバ内の UDP リスナは、その後、データベースサーバへの接続方法について、情報付きで送信側に応答します。

-sb オプションは、データベースサーバ内の UDP リスナの動作を制御します。

-sb オプションを指定しなかった場合、データベースサーバは 3 種類すべての要求に応答します。

-sb に 0 を指定した場合、データベースサーバは (3) にのみ応答します。このため、クライアントがデータベースサーバに接続するときに、Host 接続パラメータまたは HOST プロトコルオプションの使用が強制されます。さらに、サーバがデフォルトポート (2638) 以外のポートを使用している場合、クライアントもサーバのポートを指定する必要があります。データベースサーバがデータベースサーバ列挙ブロードキャストに応答しないため、データベースサーバは dblocate の出力に含まれません。

-sb に 1 を指定した場合、データベースサーバは (1) と (3) に応答します。データベースサーバがデータベースサーバ列挙ブロードキャストに応答しないため、データベースサーバは dblocate の出力に含まれません。

## **-sf iqsrv16 データベースサーバオプション**

現在のデータベースサーバで実行中のデータベースで使用できる機能にユーザがアクセスできるかどうかを制御します。セキュアな機能(セキュリティで保護された機能)には適切な権限を持つユーザのみがアクセスできます。一方で、セキュアでない機能(セキュリティで保護されない機能)にはすべてのユーザがアクセスできます。

### 構文

```
iqsrv16 -sf feature-list ...
```

```
feature-list :
feature-name | feature-set [ ,feature-name | feature-set ] ...
```

## start\_iq データベースサーバ起動ユーティリティ

機能セット	含まれている機能 (機能セットは太字)
none	manage_features、manage_keys および disk_sandbox 以外の機能はすべてセキュリティで保護されません。
manage_server	processor_affinity
manage_security	manage_features manage_keys manage_disk_sandbox
server_security	disk_sandbox trace_system_event

機能セット	含まれている機能 (機能セットは太字)
all	<p><b>client</b> –</p> <ul style="list-style-type: none"> <li>read_client_file</li> <li>write_client_file</li> </ul> <p><b>remote</b> –</p> <ul style="list-style-type: none"> <li>remote_data_access</li> <li>send_udp</li> <li>send_email</li> <li>web_service_client</li> </ul> <p><b>local</b> –</p> <ul style="list-style-type: none"> <li>• <b>local_call</b> – <ul style="list-style-type: none"> <li>cmdshell</li> <li>external_procedure</li> <li>java</li> </ul> </li> <li>• <b>local_db</b> – <ul style="list-style-type: none"> <li>backup</li> <li>restore</li> <li>database</li> <li>dbspace</li> </ul> </li> <li>• <b>local_env</b> – <ul style="list-style-type: none"> <li>getenv</li> </ul> </li> <li>• <b>local_io</b> – <ul style="list-style-type: none"> <li>create_trace_file</li> <li>read_file</li> <li>write_file</li> <li>directory</li> <li>sp_list_directory</li> <li>sp_create_directory</li> <li>sp_copy_directory</li> <li>sp_move_directory</li> <li>sp_delete_directory</li> <li>sp_copy_file</li> <li>sp_move_file</li> </ul> </li> </ul>

機能セット	含まれている機能 (機能セットは太字)
	<p style="text-align: center;">sp_delete_file</p> <ul style="list-style-type: none"> <li>• <b>local_log</b> –</li> </ul> <p style="text-align: center;">request_log console_log webclient_log</p>

### パラメータ

- **none** – どの機能もセキュリティで保護されないことを指定します。
- **manage\_server** – ユーザは、すべてのデータベースサーバ関連機能にアクセスできなくなります。この機能セットには、次の機能が含まれています。
  - **processor\_affinity** – ユーザは、データベースサーバのプロセッサのアフィニティ (使用する論理プロセッサの数) を変更できなくなります。
- **manage\_security** – ユーザは、データベースサーバのセキュリティの管理を許可する機能にアクセスできなくなります。デフォルトでは、これらの機能はセキュリティで保護されています。
  - **manage\_features** – ユーザは、データベースサーバ上でセキュリティで保護できる機能のリストを変更できなくなります。
  - **manage\_keys** – セキュリティ機能キーの作成、変更、削除、またはリストができなくなります。  
  
 manage\_keys 機能にアクセスできるものの manage\_features 機能にアクセスできないユーザは、そのユーザに割り当てられているよりも多くのセキュリティ機能を持つキーを定義することができなくなります。
- **manage\_disk\_sandbox** – ユーザは、sa\_server\_option システムプロシージャまたは sa\_db\_option システムプロシージャを使用してディスクサンドボックス設定を一時的に変更することができなくなります。すべてのデータベースまたはユーザに対して manage\_disk\_sandbox セキュリティ機能をオフにすることはできません。sp\_use\_secure\_feature\_key システムプロシージャを使用して個々の接続に対してオフにすることのみ可能です。
- **server\_security** – ユーザは、セキュリティ設定を一時的にバイパスできる機能にアクセスできなくなります。デフォルトでは、これらの機能はセキュリティで保護されています。
- **disk\_sandbox** – ユーザは、メインデータベースファイルがあるディレクトリ以外の場所でデータベースファイルの読み込み／書き込み操作を実行できなくなります。

- **trace\_system\_event** – ユーザは、ユーザ定義のトレースイベントを作成できなくなります。
- **all** – ユーザは以下のグループにアクセスできなくなります。
  - **client** – ユーザは、クライアント関連入出力へのアクセスを許可するすべての機能にアクセスできなくなります。この機能は、クライアントコンピューティング環境へのアクセスを制御します。この機能セットには、次の機能が含まれています。
    - **read\_client\_file** – クライアントファイルの読み込みを可能にする文が使用できなくなります。たとえば、`READ_CLIENT_FILE` 関数や `LOAD TABLE` 文がこれに該当します。
    - **write\_client\_file** – クライアントファイルへの書き込みを可能にする文が使用できなくなります。たとえば、`UNLOAD` 文や `WRITE_CLIENT_FILE` 関数がこれに該当します。
  - **remote** – リモートアクセスまたはリモートプロセスとの通信を許可するすべての機能にアクセスできなくなります。この機能セットには、次の機能が含まれています。
    - **remote\_data\_access** – プロキシテーブルなどのリモートデータアクセスサービスがすべて使用できなくなります。
    - **send\_udp** – `sa_send_udp` システムプロシージャを使用して指定したアドレスに UDP パケットを送信する機能が使用できなくなります。
    - **send\_email** – `xp_sendmail` などの電子メールシステムプロシージャが使用できなくなります。
    - **web\_service\_client** – Web サービスクライアントのストアードプロシージャコール (HTTP 要求を発行するストアードプロシージャ) が使用できなくなります。
  - **local** – ユーザは、すべてのローカル関連機能にアクセスできなくなります。この機能は、サーバコンピューティング環境へのアクセスを制御します。この機能セットには、`local_call`、`local_db`、`local_io`、`local_log` の各機能サブセットが含まれています。
    - **local\_call** – ユーザは、データベースサーバの直接的な一部ではなく、データベースサーバによって制御されていないコードについて、その実行機能を提供するすべての機能にアクセスできなくなります。この機能セットには、次の機能が含まれています。
      - **cmdshell** – `xp_cmdshell` プロシージャが使用できなくなります。
      - **external\_procedure** – 外部ストアードプロシージャが使用できなくなります。この設定によってデータベースサーバに組み込まれている `xp_*` システムプロシージャ (`xp_cmdshell` や `xp_readfile` など) が使用で

きなくなることはありません。これらのシステムプロシージャには、個別の機能制御オプションがあります。

- **external\_procedure\_v3** – ユーザ定義関数を参照してください。
- **java** – Java プロシージャなどの Java 関連機能が使用できなくなります。
- **local\_db** – ユーザは、すべてのデータベースファイル関連機能にアクセスできなくなります。この機能セットには、次の機能が含まれています。
  - **backup** – BACKUP 文と、BACKUP 文を使用したサーバ側バックアップの実行機能が使用できなくなります。dbbackup ユーティリティによるクライアント側バックアップは引き続き実行できます。
  - **restore** – RESTORE DATABASE 文が使用できなくなります。
  - **database** – CREATE DATABASE 文、ALTER DATABASE 文、DROP DATABASE 文、CREATE ENCRYPTED FILE 文、CREATE DECRYPTED FILE 文、CREATE ENCRYPTED DATABASE 文、CREATE DECRYPTED DATABASE 文が使用できなくなります。
  - **dbspace** – CREATE DBSPACE 文、ALTER DBSPACE 文、DROP DBSPACE 文が使用できなくなります。
- **local\_env** – ユーザは、すべての環境変数関連機能にアクセスできなくなります。この機能セットには、次の機能が含まれています。
  - **getenv** – ユーザは、どの環境変数の値の読み込みもできなくなります。
- **local\_io** – ユーザは、ファイルとその内容への直接アクセスを許可するすべての機能にアクセスできなくなります。この機能セットには、次の機能が含まれています。
  - **create\_trace\_file** – イベントトレースターゲットを作成する文が使用できなくなります。
  - **read\_file** – ローカルファイルの読み込みを可能にする文が使用できなくなります。たとえば、xp\_read\_file システムプロシージャ、LOAD TABLE 文、OPENSTRING( FILE ...) の使用がこれに該当します。代替名の load\_table や xp\_read\_file は廃止されました。
  - **write\_file** – ローカルファイルへの書き込みを可能にする文が使用できなくなります。たとえば、UNLOAD 文や xp\_write\_file システムプロシージャがこれに該当します。代替名の unload\_table や xp\_write\_file は廃止されました。
  - **delete\_file** – ローカルファイルの削除を可能にするすべての文が使用できなくなります。たとえば、この機能をセキュリティで保護する

ことで、-x オプションや -xo オプションを指定した場合に dbbackup ユーティリティの実行が失敗します。

- **directory** – ディレクトリクラスプロキシテーブルが使用できなくなります。remote\_data\_access が無効の場合、この機能は無効になります。
- **sp\_list\_directory** – sp\_list\_directory システムプロシージャが使用できなくなります。
- **sp\_create\_directory** – sp\_create\_directory システムプロシージャが使用できなくなります。
- **sp\_copy\_directory** – sp\_copy\_directory システムプロシージャが使用できなくなります。
- **sp\_move\_directory** – sp\_move\_directory システムプロシージャが使用できなくなります。
- **sp\_delete\_directory** – sp\_delete\_directory システムプロシージャが使用できなくなります。
- **sp\_copy\_file** – sp\_copy\_file システムプロシージャが使用できなくなります。
- **sp\_move\_file** – sp\_move\_file システムプロシージャが使用できなくなります。
- **sp\_delete\_file** – sp\_delete\_file システムプロシージャが使用できなくなります。
- **local\_log** – ユーザは、結果としてディスク上のファイルにデータを直接作成したり書き込んだりするすべてのロギング機能にアクセスできなくなります。この機能セットには、次の機能が含まれています。
  - **request\_log** – 要求ログのファイル名を変更する機能と、その最大サイズまたは最大ファイル数を増やす機能が使用できなくなります。データベースサーバの起動コマンドには、要求ログファイルとそのファイルの最大サイズを指定できます。ただし、それらをデータベースサーバの起動後に変更することはできません。要求ログの機能が無効になっていても、要求ロギングのオンとオフを切り替えたり、要求ログファイルの最大ファイルサイズや最大ファイル数を減らしたりすることは引き続き可能です。
  - **console\_log** – sa\_server\_option システムプロシージャの ConsoleLogFile オプションを使用してデータベースサーバメッセージログのファイル名を変更する機能が使用できなくなります。また、この機能をセキュリティで保護すると、sa\_server\_option システムプロシージャの ConsoleLogMaxSize オプションを使用してログファイルの最大サイズを増やす機能が使用できなくなります。データベースサーバの起動時には、サーバログファイルとそのサイズを指定できます。
  - **webclient\_log** – sa\_server\_option システムプロシージャの WebClientLogFile オプションを使用して Web サービスクライアントロ

## start\_iq データベースサーバ起動ユーティリティ

グのファイル名を変更する機能が使用できなくなります。データベースサーバの起動時に Web サービスクライアントログファイルを指定できます。

### 適用対象

すべてのオペレーティングシステムとデータベースサーバ。

### 備考

データベースサーバの所有者はこのオプションを使用して、データベースサーバで実行中のデータベースで使用できる機能にユーザがアクセスできるかどうかを制御できます。また、データベースサーバの所有者は、-sk オプションを使用してシステムセキュリティ機能キーを作成し、-sf オプションで指定された機能にユーザがアクセスできないようにすることができます。

システムのセキュリティ機能キーを指定しないでデータベースを起動した場合は、デフォルトのセキュリティ機能がセキュリティで保護され、データベースサーバやそのサーバで実行中のすべてのデータベースについてセキュリティ機能の設定を変更できなくなります。後でシステムのセキュリティ機能キーを作成することはできません。データベースサーバをシャットダウンし、再起動時にシステムのセキュリティ機能キーを指定する必要があります。

*feature-list* は、データベースサーバで保護する機能名または機能セットをカンマで区切って示したリストです。機能をセキュリティで保護すると、管理者以外のすべてのデータベースユーザがアクセスできなくなります。機能セットを指定すると、そのセットに含まれているすべての機能が保護されます。機能セット内の機能の全部ではなく一部を保護する場合は、個別の機能名を指定します。

---

**注意：** 機能セットの部分機能のうち、デフォルトで保護されている機能について、コマンドラインから保護を解除することはできません。たとえば、次のコマンドは動作しません。

```
-sf manage_security, -manage_keys
```

---

機能を保護する (アクセスできなくする) ことを指定するには *feature-name* を使用し、機能の保護を解除する (すべてのデータベースユーザがアクセスできるようにする) ことを指定するには *-feature-name* または *feature-name-* を使用します。たとえば、次のコマンドは、DB 領域機能にのみすべてのユーザがアクセスできることを指定します。

```
iqsrv16 -n secure_server -sf all,-dbspace
```

### 例

次のコマンドは、要求ログへのアクセスを許可し、すべてのリモートデータアクセス機能を保護した状態で、secure\_server という名前のデータベースサーバを起動します。-sk オプションで指定したキーを、後で sp\_use\_secure\_feature\_key シス



テムプロシージャで使用することで、現在の接続のすべてのユーザがそれらの機能にアクセスできるようになります。

```
iqsrv16 -n secure_server -sf remote,-request_log -sk j978kls12
```

secure\_server データベースサーバで実行中のデータベースに接続しているユーザが、sp\_use\_secure\_feature\_key システムプロシージャを使用して、-sk で指定された値と同じ値を authorization\_key パラメータに設定した場合は、その接続からリモートデータアクセス機能にアクセスできます。

```
CALL sp_use_secure_feature_key ( 'MyKey' , 'j978kls12' );
```

次のコマンドは、ローカルデータベース機能を除くすべての機能を保護します。

```
iqsrv16 -n secure_server -sf all,-local_db
```

### **-sk iqsrv16 データベースサーバオプション**

データベースサーバに対して保護されている機能へのアクセスを許可にするの  
使用できる、システムセキュリティ機能キーを指定します。

#### 構文

```
iqsrv16 -sk key ...
```

#### 適用対象

すべてのオペレーティングシステムとデータベースサーバ。

#### 備考

-sf オプションを使用してデータベースサーバの機能を保護するときに、sk オプションを含めることもできます。このオプションで指定したキーを sp\_use\_secure\_feature\_key システムプロシージャで使用すると、接続に対して保護されている機能へのアクセスを許可できます。この場合、sa\_server\_option システムプロシージャを使用して、データベースサーバ上で実行されているすべてのデータベースを保護する機能または機能セットに変更を加えることもできます。

キーは、6文字以上の空でない文字列にする必要があり、二重引用符、制御文字 (0x20 未満のすべての文字)、またはバックスラッシュを含めることはできません。データベースごとのセキュリティ機能キーは 1000 個に制限されています。

sp\_use\_secure\_feature\_key システムプロシージャの authorization\_key パラメータの値として、-sk で指定した値以外の値を設定した場合、エラーにはならず、-sf で指定した機能が接続に対して引き続き保護されます。

-sf を指定しないで -sk を指定した場合は、デフォルトのセキュリティ機能のみが有効になりますが、データベースサーバの実行中にシステムセキュリティ機能キーを使用してセキュリティ機能の設定を変更することができます。

## start\_iq データベースサーバ起動ユーティリティ

### 例

次のコマンドは、バックアップ機能を保護した状態で、secure\_server という名前のデータベースサーバを起動します。後で、-sk オプションで指定したキーを使用し、特定の接続に対してそれらの機能へのアクセスを許可できます。

```
iqsrv16 -n secure_server -sf backup -sk j978kls12
```

secure\_server データベースサーバで実行中のデータベースへの接続に対して、-sk で指定した値を authorization\_key パラメータに設定すると、secure\_server データベースサーバで保護されている機能のバックアップや変更をその接続で実行できるようになります。

```
CALL sp_use_secure_feature_key ( 'MyKey' , 'j978kls12' );
```

その後、次のコマンドを実行すると、secure\_server で実行中のデータベースに対してすべての機能を保護できます。

```
CALL sa_server_option( 'SecureFeatures', 'all' );
```

## -su iqsrv16 データベースサーバオプション

ユーティリティデータベース (utility\_db) の DBA ユーザのパスワードを設定します。または、ユーティリティデータベースへの接続を無効にします。

### 構文

```
iqsrv16 -su password ...
```

### 適用対象

すべてのオペレーティングシステムとデータベースサーバ。

### 備考

このオプションは、ユーティリティデータベースの DBA ユーザの最初のパスワードを指定します。パスワードでは大文字と小文字が区別されます。ユーティリティデータベースへのすべての接続を無効にするには、パスワードに none を指定します。

- ネットワークデータベースサーバを使用している場合で、-su オプションを指定しないときに、ユーティリティデータベースに接続するには、util\_db.ini ファイルが存在することと、ユーザ ID = DBA および util\_db.ini ファイルの内容と一致するパスワードを使用することが必要です。ネットワークサーバで -su と util\_db.ini ファイルの両方を使用すると、util\_db.ini ファイルが無視されます。util\_db.ini ファイルの使用は推奨されません。  
ネットワークデータベースサーバ用のユーティリティデータベースにパスワードを指定することは、データベースサーバが実行中の場合には便利ですが、データベースに接続できません。

utility\_db に接続しているときに、CREATE USER DBA IDENTIFIED BY *new-password* 文を実行して、ユーティリティデータベースの DBA ユーザのパスワードを変更できます。utility\_db データベースへの接続を無効にするには、REVOKE CONNECT FROM DBA 文を使用します。すべての SQL 文がユーティリティデータベースに対してサポートされているわけではありません。

ユーティリティデータベースのパスワードがコマンドライン上でクリアテキストで入力されるのを避けるには、dbfhide ユーティリティを使用してパスワードを含むファイルの内容を読みにくくし、難読化されたファイルをコマンドライン上で参照します。

### 例

次のコマンドは、ユーティリティデータベースへのすべての接続を無効にします。

```
iqsrv16 -su none c:¥inventory.db
```

次の例では、ユーティリティデータベースのパスワードを含む util\_db\_pwd.cfg ファイルが dbfhide によって難読化され、ファイル名が util\_db\_pwd\_hide.cfg に変更されます。

```
dbfhide util_db_pwd.cfg util_db_pwd_hide.cfg
```

その後、util\_db\_pwd\_hide.cfg ファイルを使用して、ユーティリティデータベースのパスワードを指定できます。

```
iqsrv16 -su @util_db_pwd_hide.cfg -n my_server c:¥inventory.db
```

## -tdsl iqsrv16 データベースサーバオプション

TDS ログインモードを設定します。

### 構文

```
iqsrv16 -tdsl { all | RSA | RSANonce }...
```

### デフォルト

all

### 適用対象

すべてのオペレーティングシステムとデータベースサーバ。

### 備考

このオプションは、データベースサーバでサポートされる TDS ログイン要求のタイプを制限します。

モードのタイプ	説明
all	暗号化されたパスワードと暗号化されていないパスワード (臨時のパスワードを含む) が TDS ログイン要求で許可されます。
RSA	RSA 暗号化されたパスワード (臨時のパスワードを含む) のみが TDS ログイン要求で許可されます。
RSANonce	RSA 暗号化された臨時のパスワードのみが TDS ログイン要求で許可されます。

臨時ではない RSA をサポートする TDS アプリケーションからログイン要求を行うと、ログイン要求用の新しい暗号化キーのセットがデータベースサーバで生成されます。新しい暗号化キーの生成には時間がかかることがあります。

臨時の RSA をサポートする TDS アプリケーションからログイン要求を行うと、RSA 暗号化キーのセットがデータベースサーバで再利用されます。これらの暗号化キーは 24 時間ごとに再生成されます。RSA 暗号化キーを再利用することによって、パフォーマンスを向上させると同時に、データベースサーバをリプレイ攻撃から守ることができます。

jConnect と Open Client のどちらも、臨時の RSA ログイン要求と臨時ではない RSA ログイン要求をサポートしています。

## **-ti iqsrv16 データベースサーバオプション**

非アクティブ接続を切断します。

### 構文

```
iqsrv16 -ti minutes ...
```

### デフォルト

240 (4 時間)

### 適用対象

すべてのオペレーティングシステムとデータベースサーバ。

### 備考

*minutes* で指定された分数の間、要求を送信しなかった接続を切断します。最大値は 32767 です。データベーストランザクション処理中のクライアントコンピュータは、トランザクションが終了するか、接続が切断されるまでロックされます。-ti オプションを指定すると、非アクティブ接続が切断され、ロックが解除されません。

ほとんどの接続はネットワークリンク (TCP) 経由で行われているため、`-ti` オプションは `iqsrv16` と一緒に使用すると非常に便利です。

値を 0 に設定すると、非アクティブ接続は検査されず、接続が切断されません。Idle 接続パラメータが使用されていない場合、TCP/IP 接続のアイドルタイムアウト値は `-ti` データベースサーバオプションによって制御されます。両方の `-ti` データベースサーバオプションと Idle 接続パラメータが指定された場合、アイドルタイムアウト値は接続パラメータによって制御されます。

### **-tl iqsrv16 データベースサーバオプション**

活性パケットを送信する期間を設定します。

#### 構文

```
iqsrv16 -tl seconds ...
```

#### 適用対象

TCP/IP を使用するすべてのデータベースサーバ

#### 備考

接続が維持されていることを確認するため、クライアント/サーバの TCP/IP 通信プロトコルを介して、定期的に活性パケットが送信されます。接続で活性パケットを検出することなく、指定した LivenessTimeout 時間 (デフォルトは 2 分) にわたってサーバが実行されていると、通信は切断され、サーバはそのクライアントに関連付けられている接続を削除します。非スレッドの UNIX クライアントと TDS 接続では、活性パケットによる確認は行われません。

サーバで `-tl` オプションを指定すると、活性期間が指定されていないすべてのクライアントに対して LivenessTimeout 値を設定できます。

LivenessTimeout 値の 3 分の 1 から 3 分の 2 の期間で接続がパケットを送信しない場合に、活性パケットが送信されます。

接続数が 200 を超えると、サーバは指定されている LivenessTimeout 値に基づいて、それより大きい LivenessTimeout 値を自動的に計算するため、多数の接続を効率的に処理できます。活性パケットは、各アイドル接続において、LivenessTimeout 値の 3 分の 1 から 3 分の 2 の期間で送信されます。大量の活性パケットが同時に送信されることはありません。送信に時間がかかる活性パケットは、LivenessTimeout 値の 3 分の 2 の期間が経過した後で送信できます。活性パケットの送信に時間がかかる場合、データベースサーバメッセージログに警告が表示されます。この警告が発生したら、LivenessTimeout 値の増加を検討してください。

これは一般的にはおすすめしませんが、次のように指定して活性タイムアウトを無効にできます。

## start\_iq データベースサーバ起動ユーティリティ

```
iqsrv16 -tl 0 -n my_server
```

LivenessTimeout オプションを無効にせずに、次のように値を 1 時間に増やすことを検討してください。

```
iqsrv16 -tl 3600 -n my_server
```

## -tmf iqsrv16 データベースサーバオプション

トランザクションマネージャに、強制的に分散トランザクションをリカバリさせます。

### 構文

```
iqsrv16 -tmf ...
```

### 適用対象

Windows

### 備考

このオプションは、分散トランザクションコーディネーターが使用できない場合に、分散トランザクションのリカバリ中に使用します。DTC が特定できないときは、未処理の操作をロールバックしてリカバリを続行します。また、DTC が使用できないプラットフォームで、トランザクションログに分散トランザクションがあるデータベースを起動する場合にも使用できます。

---

### 警告！ 警告

このオプションを使用すると、分散トランザクションは正常にはリカバリされません。日常的な使用は想定していません。

---

## -tq iqsrv16 データベースサーバオプション

指定の時刻にサーバを停止します。

### 構文

```
iqsrv16 -tq { datetime | time } ...
```

### 適用対象

すべてのオペレーティングシステムとデータベースサーバ。

### 備考

このオプションは、自動オフラインバックアッププロシージャの設定に役立ちません。

時間のフォーマットは *hh:mm* (24 時間表記) で、オプションで前に日付を付けることができます。日付を指定する場合は、日付と時間を二重引用符で囲んで、"*YYYY/MM/DD HH:MM*" の形式にする必要があります。

## **-u iqsrv16 データベースサーバオプション**

オペレーティングシステムのディスクキャッシュを使用してファイルを開きます。

### 構文

```
iqsrv16 -u ...
```

### 適用対象

Windows、UNIX

### 備考

データベースキャッシュに加え、オペレーティングシステムのディスクキャッシュも使ってファイルが開かれます。

オペレーティングシステムのディスクキャッシュによってパフォーマンスが向上することもあります。データベースキャッシュだけを使用すると、パフォーマンスがさらに向上します。

サーバを専用コンピュータで実行している場合は、`-u` オプションを使用しないでください。通常、データベースキャッシュを使用する方が効率的です。`-u` オプションを使用するのは、サーバを実行するコンピュータに他にもいくつかのアプリケーションがインストールされていて(サイズの大きなデータベースキャッシュが他のアプリケーションを実行するための障害となり)、しかも I/O の多いタスクをサーバ上で断続的に実行する(キャッシュサイズを大きくすることでパフォーマンスが向上する)場合です。

## **-ud iqsrv16 サーバオプション**

(UNIX サーバ) プロセスをルートディレクトリ内でデーモンとして実行します。IQ サーバではこのスイッチを使用しないことをおすすめします。

## **-uf iqsrv16 データベースサーバオプション**

致命的なエラーまたはアサーションの失敗がデータベースサーバで発生したときにデータベースサーバが実行するアクションを指定します。

### 構文

```
iqsrv16 -uf action ...
```

### デフォルト

default

## start\_iq データベースサーバ起動ユーティリティ

### 指定可能な値

- **abort** – データベースサーバが停止し、コアファイルが生成されます。
- **default** – データベースサーバは、abortと同じ動作をします。ただし、デバイスフルの致命的なエラーが発生した場合は、defunctと同じ動作となります。このアクションによって、システムはフルデバイスとなってもコアファイルを作成しようとしません。
- **defunct** – データベースサーバは実行を続け、abortを呼び出しません。データベースサーバに対して新しい接続を試みると、元のエラーが発生します。

### 適用対象

すべてのオペレーティングシステム。

## -ut iqsrv16 サーバオプション

(UNIX サーバ) *min* によって指定された時間の間隔で、サーバがカタログストアのテンポラリファイルにタッチします。

### 構文

**-ut min**

## -v iqsrv16 サーバオプション

メッセージボックス (Windows)、またはバージョン文字列 (UNIX/Linux) にデータベースサーババージョンを表示します。

### 構文

**-v** または **-v2**

## -x iqsrv16 データベースサーバオプション

サーバ側のネットワーク通信プロトコルを指定します。

### 構文

```
iqsrv16 -x { all | none | tcpip [ ( parm=value;... ) ] ...
```

### 指定可能な値

-x オプションについて選択した設定に関係なく、データベースサーバは常に共有メモリ接続を受け入れます。

デフォルトでは、ネットワークデータベースサーバは共有メモリプロトコルとTCP/IP プロトコルを起動します。

-x オプションには次の値を指定できます。



- **TCPIP (TCP)** – 共有メモリプロトコルと TCP/IP プロトコルを使用するクライアントによる接続試行を受信します。
- **NONE** – 共有メモリプロトコルのみを使用するクライアントによる接続要求を受信します。
- **ALL** – 共有メモリプロトコルと TCP/IP プロトコルを使用するクライアントによる接続試行を受信します。

#### 適用対象

すべてのオペレーティングシステムとデータベースサーバ。

#### 備考

-x オプションを使用して、クライアント接続の受信に使用する通信プロトコルを指定します。

TCP/IP プロトコルの場合、追加のパラメータは次のフォーマットで提供されません。

```
-x tcpip(PARM1=value1; PARM2=value2; ...)
```

UNIX では、複数のパラメータを指定する場合、または丸カッコなどの特定の記号文字を使用する場合に二重引用符が必要です。次に例を示します。

```
-x "tcpip(PARM1=value1; PARM2=value2; ...)"
```

データベースサーバが TCP/IP 接続を受信するとき、デフォルトでポート 2638 からすべてのネットワークカードを受信します。

#### 例

ネットワークサーバで次のように共有メモリ接続と TCP/IP 接続を許可します。

```
iqsrv16 -n server_name "%IQDIRSAMP16%¥demo.db"
```

-x オプションを使用して、TCP/IP の動作を調整するプロトコルオプションを指定します。ネットワークサーバに 2 つの具体的なネットワークカードを使用することを許可します。

```
iqsrv16 -x "tcpip(MyIP=192.75.209.12,192.75.209.32)" "%IQDIRSAMP16%¥demo.db"
```

## -xd iqsrv16 サーバオプション

データベースサーバがデフォルトのデータベースサーバにならないようにします。

データベースサーバを起動すると、そのサーバは、コンピュータのデフォルトデータベースサーバになろうとします。デフォルトサーバがない状態で最初に起動するデータベースサーバが、デフォルトのデータベースサーバになります。そのコンピュータで共有メモリ接続を試みたとき、データベースサーバ名が明示的に指定されていないと、デフォルトサーバに接続されます。

## start\_iq データベースサーバ起動ユーティリティ

このオプションを指定すると、データベースサーバが、デフォルトのデータベースサーバにならなくなります。このオプションが指定されている場合、データベースサーバ名を指定していないクライアントは、共有メモリを通じてデータベースサーバを検索できません。また、**-xd** オプションを指定すると、データベースサーバはデフォルトの TCP ポートを使用できなくなります。TCP ポートが指定されていない場合、データベースサーバはポート 2638 以外のポートを使用します。

## -xs iqsrv16 データベースサーバオプション

サーバ側の Web サービス通信プロトコルを指定します。

### 構文

```
iqsrv16 -xs [,...] { protocol[,...] }
```

```
protocol : {  
NONE  
| HTTP [ ( option=value;... ) ]  
| HTTPS [ ( option=value;... ) ]
```

### 指定可能な値

次のいずれかを指定できます。

- **option** – ネットワークプロトコルオプション
- **HTTP** – HTTP プロトコルを使用するクライアントによる Web 要求を受信します。受信するデフォルトのポートは 80 です。
- **HTTPS** – HTTPS プロトコルを使用するクライアントによる Web 要求を受信します。受信するデフォルトのポートは 443 です。HTTPS を使用するためには、サーバの証明書とパスワードを指定する必要があります。HTTPS は RSA 暗号化を使用するため、パスワードは RSA 証明書であることが必要です。

HTTP サーバは、SSL バージョン 3.0 と TLS バージョン 1.0 および 1.1 を使用した HTTPS 接続をサポートしています。

HTTPS を指定するか、FIPS 認定の RSA 暗号化の場合は FIPS=Y を付けて HTTPS を指定します。FIPS 認定の HTTPS は別の認定ライブラリを使用しますが、HTTPS と互換性があります。

- **NONE** – Web 要求を受信しません。これはデフォルトです。

### 適用対象

すべてのオペレーティングシステムとデータベースサーバ

### 備考

-xs オプションを使用して、要求の受信に使用する Web プロトコルを指定します。

-xs オプションを指定しない場合、データベースサーバは Web 要求を受信しようとしません。

複数のプロトコルを指定する場合は、プロトコルごとに -xs オプションを指定することも、1つの -xs オプションで複数のプロトコルを指定することもできます。サーバは、指定されたすべてのプロトコルを使用して Web 要求を受信します。

---

**注意：** 複数の Web サーバを同時に起動する場合は、どのサーバにも同じデフォルトポートがあるので、いずれか1つのサーバのポートを変更します。

---

トランスポートレイヤセキュリティには、HTTPS または FIPS 認定の HTTPS プロトコルを使用できます。

UNIX では、複数のパラメータを指定する場合に二重引用符が必要です。

```
-xs "HTTP (OPTION1=value1;OPTION2=value2;...)"
```

---

**注意：** 強力な暗号化テクノロジーはすべて、輸出規制対象品目です。

---

### 例

HTTP Web 要求をポート 80 で受信します。

```
iqsrv16 web.db -xs HTTP (PORT=80)
```

HTTPS を使用して Web 要求を受信します。

```
iqsrv16 web.db -xs  
HTTPS (FIPS=N;PORT=82;IDENTITY=ecserver.id;IDENTITY_PASSWORD=test)
```

ポート 80 とポート 8080 で受信：

```
iqsrv16 -xs HTTP (port=80),HTTP (PORT=8080)
```

または

```
iqsrv16 -xs HTTP (port=80) -xs HTTP (PORT=8080)
```

## **-z iqsrv16 データベースサーバオプション**

診断通信メッセージなどのメッセージをトラブルシューティングのために表示します。

### 構文

```
iqsrv16 -z ...
```

### 適用対象

すべてのオペレーティングシステムとデータベースサーバ。

### 備考

このオプションは、問題の原因を突き止める場合にだけ使用します。情報は、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。

## **-ze iqsrv16 データベースサーバオプション**

データベースサーバ環境変数をデータベースサーバメッセージウィンドウに表示します。

### 構文

```
iqsrv16 -ze ...
```

### 適用対象

すべてのオペレーティングシステムとデータベースサーバ。

### 備考

-ze オプションを指定すると、起動時に環境変数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。データベースサーバメッセージウィンドウの内容をファイルに保存するには、データベースサーバの起動時に -o オプションを指定します。

### 例

次のコマンドは、データベースサーバ myserver を起動し、このサーバに設定されている環境変数をデータベースサーバメッセージウィンドウと server-log.txt ファイルに出力します。

```
iqsrv16 -n myserver -ze -o server-log.txt
```

## **-zl iqsrv16 データベースサーバオプション**

サーバ上の各データベース接続の、最後に作成された SQL 文の取得をオンにします。

### 構文

```
iqsrv16 -zl ...
```

### 適用対象

すべてのオペレーティングシステムとデータベースサーバ。

### 備考

この機能は、RememberLastStatement サーバ設定を使用してオンにすることもできます。ある接続で最後に準備された SQL 文は、CONNECTION\_PROPERTY 関数の LastStatement 値を使用して取得できます。sa\_conn\_activity ストアドプロシージャを使用すると、サーバ上の現在のすべてのデータベース接続について、最後に作成された SQL 文を取得できます。

LastStatement の値は、文が準備されると同時に設定され、文が削除されると同時にクリアされます。各接続につき 1 つの文の文字列のみが記憶されます。

ある接続について sa\_conn\_activity が空でない値を返した場合、その接続で現在実行されている文である可能性が高くなります。その文が完了している場合は、文がすでに削除され、このプロパティの値がクリアされている可能性があります。アプリケーションが複数の文を準備し、それらの文のステートメントハンドルを保持している場合、LastStatement が返す値は接続で現在実行されている処理を表しません。

ストアドプロシージャコールの場合、プロシージャ内の文ではなく、最も外側のプロシージャコールのみが表示されます。

---

### 警告！ 警告

-zl が指定されている場合、または RememberLastStatement サーバの設定がオンになっている場合は、任意のユーザが sa\_conn\_activity システムプロシージャを呼び出すか LastStatement 接続プロパティの値を取得して、他のユーザに対して最後に作成された SQL 文を調べることができます。このオプションは注意して使用し、不要な場合はオフにしてください。

---

## -zn iqsrv16 データベースサーバオプション

保持する要求ログファイルのコピー数を指定します。

### 構文

```
iqsrv16 -zn integer
```

### 適用対象

すべてのオペレーティングシステムとデータベースサーバ。

### 備考

要求ロギングが長時間にわたって有効になっていると、要求ログファイルが大きくなることがあります。-zn オプションを使用すると、保持する要求ログファイルのコピー数を指定できます。ただし、-zs も指定されていなければ有効になりません。-zs オプションにより、元のログファイルが指定のサイズに到達すると、新しいログファイルを作成し、元のログファイルの名前を変更できます。

たとえば、要求ロギング情報を req.out ファイルにリダイレクトし、-zn オプションを使って5つのログファイルコピーを要求すると、サーバは req.out.1、req.out.2、req.out.3、req.out.4、req.out.5 の順にファイルを作成します。これらのファイルが存在する場合、アクティブな要求ログが再び一杯になると以下の動作が発生します。

- req.out.1 が削除される。
- req.out.2 ~ req.out.5 のファイル名が req.out.1 ~ req.out.4 に変更される。

## start\_iq データベースサーバ起動ユーティリティ

- アクティブなログファイルのコピーの名前が req.out.5 に変更される。

要求ロギングを有効にするには、`-zr` オプションを使用します。このログは、`-zo` オプションにより別のファイルにリダイレクトできます。また、`sa_server_option` システムプロシージャを使用して、要求ログの数を設定することもできます。その場合、`nn` には、要求ログファイルのコピーの数を指定します。

```
CALL sa_server_option('RequestLogNumFiles',nn);
```

### 例

次の例 (全体を 1 行に入力) では、要求ロギング情報は `mydatabase.log` という要求ログファイルに出力されます。このファイルは最大サイズが 10 KB で、要求ログのコピーが 3 つ保持されます。

```
iqsrv16 "c:¥my data¥mydatabase.db" -zr all -zn 3  
-zs 10 -zo mydatabase.log
```

## -zo iqsrv16 データベースサーバオプション

通常のログファイルとは別のファイルに、要求ロギング情報をリダイレクトします。

### 構文

```
iqsrv16 -zo filename ...
```

### 適用対象

すべてのオペレーティングシステムとデータベースサーバ。

### 備考

要求ロギングは、`-zr` オプションを使用すると有効になります。`-zo` オプションを指定することによって、出力をこのファイルから通常のログファイルではない別のファイルにリダイレクトできます。

また、このオプションにより、要求ロギング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なくなります。

## -zoc iqsrv16 データベースサーバオプション

Web サービスクライアント情報をファイルにリダイレクトします。

### 構文

```
iqsrv16 -zoc filename ...
```

### 適用対象

すべてのオペレーティングシステムとデータベースサーバ。

### 備考

Web サービスクライアントログファイルには、HTTP 要求とアウトバウンド Web サービスクライアントログコール用に記録されたトランスポートデータが含まれています。Web サービスクライアントログファイルは、sa\_server\_option システムプロシージャを使用して指定することもできます。

```
CALL sa_server_option( 'WebClientLogFile', 'clientinfo.txt' );
```

-zoc サーバオプションを指定すると、ロギングが自動的に有効になります。このファイルへのロギングの有効/無効を切り替えるには、sa\_server\_option システムプロシージャを使用します。

```
CALL sa_server_option( 'WebClientLogging', 'ON' );
```

### 例

次のコマンドは、データベースサーバを起動し、HTTP Web 要求をポート 80 で受信し、アウトバウンド Web サービスクライアント情報を clientinfo.txt ファイルに記録します。

```
iqsrv16 web.db -xs HTTP(PORT=80) -zoc clientinfo.txt
```

## -zr iqsrv16 データベースサーバオプション

操作の要求ロギングを有効にします。

### 構文

```
iqsrv16 -zr { SQL | HOSTVARS | PLAN | PROCEDURES | TRIGGERS | OTHER | BLOCKS | REPLACE | ALL | NONE } ...
```

### 指定可能な値

- **SQL** – 以下の項目のロギングを有効にします。

START DATABASE 文

STOP DATABASE 文

STOP SERVER 文

文の準備と実行

EXECUTE IMMEDIATE 文

オプション設定

COMMIT 文

ROLLBACK 文

PREPARE TO COMMIT 操作

接続と接続解除

トランザクションの開始

## DROP STATEMENT 文

カーソルの説明

カーソルを開く、閉じる、再開する

エラー

- **PLAN** – 実行プランのロギングを有効にします (短いプラン)。プロシージャの実行プランは、プロシージャ (PROCEDURES) のロギングが有効な場合にも記録されます。
- **HOSTVARS** – ホスト変数の値のロギングを有効にします。HOSTVARS を指定した場合、SQL にリストされている情報もロギングされます。
- **PROCEDURES** – プロシージャとユーザ定義の関数内から実行される文のロギングを有効にします。
- **TRIGGERS** – トリガ内から実行されている文のロギングを有効にします。
- **OTHER** – SQL に含まれないその他の要求タイプのロギングを有効にします (FETCH や PREFETCH など)。ただし、OTHER を指定して SQL を指定しない場合、SQL+OTHER を指定した場合と同じです。OTHER を含めると、ログファイルが急速に拡大し、サーバのパフォーマンス低下につながることがあります。
- **BLOCKS** – 別の接続で接続がブロックされたときと、接続のブロックが解除されたときに表示する詳細のロギングを有効にします。
- **REPLACE** – ロギングの開始時に、既存の要求ログは同じ名前を持つ新規の (空の) ログで置換されます。それ以外の場合、既存の要求ログが開き、新規エントリがファイルの末尾に追加されます。
- **ALL** – すべてのサポート情報をロギングします。この設定は、SQL+PLAN+HOSTVARS+PROCEDURES+TRIGGERS+OTHER+BLOCKS を指定した場合と同じです。この設定では、ログファイルが急速に拡大し、サーバのパフォーマンス低下につながることがあります。
- **NO または NONE** – 要求ログに対するロギングを無効にします。

## 適用対象

すべてのオペレーティングシステムとデータベースサーバ。

## 備考

このオプションは、問題の原因を突き止める場合にだけ使用します。情報は、データベースサーバメッセージウィンドウに表示されるか、要求ログに送信されます。複数の値を指定する場合、これらの値は、または + で区切られます。

データベースサーバを起動した後で、sa\_server\_option システムプロシージャを使用し、要求ログ設定を変更してロギングの対象とする情報を増減できます。

RequestLogging 設定の現在の値は、次のクエリを使用して検索できます。

```
SELECT PROPERTY( 'RequestLogging' );
```



## **-zs iqsrv16 データベースサーバオプション**

要求ログのサイズを制限します。

### 構文

```
iqsrv16 -zs { size[ k | m | g ] } ...
```

### 適用対象

すべてのオペレーティングシステムとデータベースサーバ。

### 備考

要求ロギングを有効にするには、`-zr` オプションを使用します。このログは、`-zo` オプションにより別のファイルにリダイレクトできます。また、`-zs` オプションを使用してファイルのサイズを制限できます。

`size` には、要求ログの最大ファイルサイズを、バイト単位で指定します。単位をキロバイト、メガバイト、またはギガバイトで指定するには、それぞれ `k`、`m`、`g` のいずれかを使用してください。

`-zs 0` を指定した場合は、要求ロギングファイルの最大サイズは適用されず、名前は変更されません。これはデフォルト値です。

要求ログファイルが `-zs` オプションまたは `sa_server_option` システムプロシージャで指定したサイズに到達すると、ファイルが拡張子 `.old` の付いた名前に変更されます (既存のファイルが存在する場合は、同じ名前で置換されます)。要求ログファイルが再起動します。

### 例

次の例では、ログファイルのサイズを制御するための `-zs` オプションの使用方法を示します。次のコマンドラインを使用してデータベースサーバを起動するとします。

```
iqsrv16 -zr all -zs 10k -zo mydatabase.log
```

新規ログファイル `mydatabase.log` が作成されます。このファイルのサイズが `10 KB` に達すると、既存の `mydatabase.old` ファイルが削除され、`mydatabase.log` の名前が `mydatabase.old` に変更されて、新しい `mydatabase.log` ファイルが開始されます。このプロセスは、`mydatabase.log` ファイルが指定したサイズ (この場合は `10 KB`) に達するたびに繰り返されます。

## start\_iq データベースファイルパラメータ

ここでは、データベースサーバ/データベースファイルのパラメータを示します。コマンド構文のサーバオプションの後にデータベースファイルを指定します。

表 34 : start\_iq データベースファイルパラメータ

パラメータ	説明
<code>-n server-name</code>	データベースサーバの名前を指定する。
<code>database-file</code>	<p>データベースファイル名を指定する。ファイル拡張子を付けないで <code>database-file</code> を指定した場合、SAP Sybase IQ は拡張子 <code>.db</code> を付けて <code>database-file</code> を検索する。</p> <p>相対パスを使用すると、パスはサーバの現在の作業ディレクトリからの相対で読み込まれる。フルパスを指定できる。</p> <p>Windows では、次のような UNC (Universal Naming Convention) に従ってパスを指定できる。</p> <pre>¥¥server¥volume¥path¥file.ext</pre>

**警告!** データベースファイルは、データベースサーバと同じマシン上にある必要があります。ネットワークドライブにあるデータベースファイルを操作すると、ファイルが破損することがあります。

## start\_iq のデータベースオプション

`start_iq database-options` パラメータで利用可能なスイッチについて説明します。

データベースファイルの後にこれらのオプションを指定します。オプションはそのデータベースにのみ適用されます。これらのオプションは、コマンド構文内の先行するデータベースのみに適用されます。

`start_iq` データベースオプションには、2つの構文形式があります。

- 設定ファイルでオプションを指定する場合は、オプション値を一重引用符で囲みません。次に例を示します。

```
iqdemo.db -ek xxx
```

- コマンドラインでオプションを指定する場合は、オプション値を一重引用符で囲みます。次に例を示します。

```
start_iq @iqdemo.cfg iqdemo.db -ek 'xxx'
```

**参照:**

- dblocate データベース管理ユーティリティ (75 ページ)

**-a iqsrv16 データベースオプション**

指定したトランザクションログを適用します。

**構文**

```
iqsrv16 [ server-options ] database-file -a log-filename ...
```

**適用対象**

すべてのオペレーティングシステムとデータベースサーバ

**備考**

このオプションは、データベースファイルのメディア障害からのリカバリに使用されます。このオプションを指定すると、データベースサーバはログを適用して終了します (実行は継続しません)。複数のトランザクションログを適用する場合は、-a オプションを使用するときにトランザクションログの正しい適用順序を把握する必要があります。-ad オプションまたは -ar オプションを使用すると、複数のトランザクションログが自動的に正しい順序で適用されます。

-a データベースオプションは、*database-file* の後に指定する必要があり、そのデータベースだけに適用されます。

サーバの起動時にキャッシュサイズを指定すると、リカバリ時間を短縮できます。

**例**

次の例 (全体を 1 行に入力) では、ログファイルが mydemo データベースのバックアップコピーに適用されます。

```
iqsrv16 "c:¥backup¥mydemo.db" -a "c:¥backup¥mydemo.log"
```

**-ad iqsrv16 データベースオプション**

データベースに適用されるトランザクションログファイルがあるディレクトリを指定します。

**構文**

```
iqsrv16 [ server-options ] database-file -ad log-directory ...
```

**適用対象**

すべてのオペレーティングシステムとデータベースサーバ

**備考**

-ad オプションを使用すると、指定したディレクトリでデータベースのトランザクションログファイルが検索されます。トランザクションログファイルの開始ログ

## start\_iq データベースサーバ起動ユーティリティ

オフセットがデータベースファイルに保存されている開始ログオフセットと等しいか、それより大きい場合、トランザクションログファイルはログオフセット順に適用されます。すべてのトランザクションログファイルが適用されると、データベースは停止します。トランザクションログファイルの適用が完了した後もデータベースの実行を継続する場合は、`-as` オプションも一緒に指定する必要があります。

指定したログディレクトリが相対パスを使用する場合、このディレクトリは *database-file* ディレクトリに対して相対的に読まれます。

`-ad` データベースオプションは、*database-file* の後に指定する必要があり、そのデータベースだけに適用されます。

### 例

次の例では、`backup` ディレクトリ内のログファイルが `mysample.db` データベースに適用され、適用が完了すると、データベースが停止します。

```
iqsrv16 "c:¥mysample.db" -ad "c:¥backup"
```

次の例では、`backup` ディレクトリ内のログファイルが `mysample.db` データベースに適用され、適用の完了後もデータベースの実行は継続します。

```
iqsrv16 "c:¥mysample.db" -ad "c:¥backup" -as
```

## -al iqsrv16 データベースオプション

標準認証を使用して、特定数のユーザのみに LDAPUA の LOGIN\_MODE を拡張します。

### 構文

```
-al "user1;user2;user3" server_name.cfg database_name.db
```

### 備考

- 最大5つのユーザ ID をセミコロンで区切って指定できます。各ユーザ ID は二重引用符で囲みます。
- データベースレベルでの実行時には、データベースの次の停止/起動時まで有効です。

## -ar iqsrv16 データベースオプション

現在のトランザクションログと同じディレクトリ内にあるトランザクションログファイルをデータベースに適用します。

### 構文

```
iqsrv16 [ server-options ] database-file -ar ...
```

### 適用対象

すべてのオペレーティングシステムとデータベースサーバ

### 備考

-ar オプションを使用すると、データベースサーバのトランザクションログファイルは現在のトランザクションログと同じディレクトリで検索されます。トランザクションログのディレクトリはデータベースから取得されます。トランザクションログファイルの開始ログオフセットがデータベースに保存されている開始ログオフセットと等しいか、それより大きい場合、トランザクションログファイルはログオフセット順に適用されます。すべてのトランザクションログファイルが適用されると、データベースは停止します。トランザクションログファイルの適用が完了した後もデータベースの実行を継続する場合は、-as オプションも一緒に指定する必要があります。

-ar データベースオプションは、*database-file*の後に指定する必要があり、そのデータベースだけに適用されます。

### 例

次の例では、トランザクションログファイル (保存されているディレクトリはデータベースから取得されます) が `mysample.db` データベースに適用されます。トランザクションログファイルの適用が完了した後もデータベースの実行は継続します。

```
iqsrv16 "c:\mysample.db" -ar -as
```

## -as iqsrv16 データベースオプション

トランザクションログの適用後もデータベースの実行を継続します (-ad または -ar とともに使用)。

### 構文

```
iqsrv16 [ server-options ] database-file { -ad log-dir | -ar } -as ...
```

### 適用対象

すべてのオペレーティングシステムとデータベースサーバ

### 備考

-as オプションを指定する場合は、-ad オプションまたは -ar オプションも指定する必要があります。-as オプションを指定すると、トランザクションログの適用後もデータベースの実行が継続します。

-as データベースオプションは、*database-file*の後に指定する必要があり、そのデータベースだけに適用されます。

## start\_iq データベースサーバ起動ユーティリティ

### 例

次の例では、トランザクションログファイルが `mysample.db` データベースに適用されます。-ar が指定されているため、データベースサーバはトランザクションログのロケーションをデータベースから取得します。ログファイルの適用が完了した後もデータベースの実行は継続します。

```
iqsrv16 "c:¥mysample.db" -ar -as
```

次の例では、`backup` ディレクトリ内のログファイルが `mysample.db` データベースに適用されます。ログファイルの適用が完了した後もデータベースの実行は継続します。

```
iqsrv16 "c:¥mysample.db" -ad "c:¥backup" -as
```

### -dh iqsrv16 データベースオプション

サーバに対してサーバ検出ユーティリティ `dblocate -d` が実行されたときに、データベースを検知不能にします。

### -ds iqsrv16 データベースオプション

データベースの DB 領域とトランザクションログが配置されているディレクトリを指定します。

#### 構文

```
iqsrv16 database-file -ds dbspace-directory ...
```

#### 適用対象

すべてのオペレーティングシステムとデータベースサーバ。

このオプションは、IQ DB 領域ではなくカタログ DB 領域にのみ適用されます。

#### 備考

DB 領域のディレクトリを指定すると、データベースサーバは、DB 領域を探すときにこのディレクトリだけを検索対象とします。DB 領域のロケーションは、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。

フルパス名を指定された DB 領域がバックアップに含まれている場合は、このオプションを使用して、元のデータベースが実行中である間に、元のデータベースと同じコンピュータ上にあるデータベースのバックアップコピーを起動できます。

-ds データベースオプションは、`database-file` の後に指定する必要があり、そのデータベースだけに適用されます。

このオプションで指定されたディレクトリでトランザクションログが見つからない場合は、このロケーションにトランザクションログが作成されます。

---

**警告！ 警告**

-ds オプションはリカバリにのみ使用してください。このオプションを指定し、ここで指定したディレクトリに現在のライブトランザクションログが存在しない場合は、新しいトランザクションログが指定のロケーションで作成されます。

---

**例**

次の例では、ディレクトリ `c:\¥backup¥Nov15` に対して DB 領域を検索するデータベースサーバを起動します。

```
iqsrv16 c:\¥backup¥Nov15¥my.db -ds c:\¥backup¥Nov15¥
```

次の例では、現在のディレクトリに対して DB 領域を検索するデータベースサーバを起動します。

```
iqsrv16 my.db -ds .
```

**-ek iqsrv16 データベースオプション**

データベースの暗号化キーを指定します。

**構文**

**-ek** key

**備考**

強い暗号で保護されたデータベースのファイル名の後に指定します。暗号化データベースを起動するためのキー値が引数として必要です。キー値は文字列で、大文字と小文字、数値、英字、特殊文字を併用できます。強い暗号で保護されたデータベースがある場合は、データベースまたはトランザクションログを使用するために、暗号化キーを指定する必要があります。暗号化キーを指定しないと、コマンドは失敗します。強い暗号で保護されたデータベースの場合、**-ek** または **-ep** を指定する必要があります (両方を同時に指定することはできません)。

**-iqfreq iqsrv16 データベースオプション**

指定したデータベースを使用中であるとマーク付けし、データベースの IQ メインストア部分を一貫性のあった最後の状態に戻します。

**構文**

**-iqfreq** dbname

**備考**

**-iqfreq** は、通常の操作では使用しないでください。このオプションは、SAP Sybase IQ サーバ障害後のリカバリ中に `s_buf` またはフリーリストのエラーが発生し、データベースの強制リカバリを行うときにのみ使用してください。 *dbname*

## start\_iq データベースサーバ起動ユーティリティ

は、論理データベース名やニックネームではなく物理データベース名である必要があります。

---

**注意：** `-iqfreq` オプションは、IQ カタログストアではなくデータベースの IQ メインストア部分にのみ適用されます。 `-iqfreq` を使用して、カタログストアの強制リカバリを実行することはできません。

---

`-iqfreq` を使用する際には、システムリカバリおよびデータベース修復の正しい手順に従ってください。

## -m iqsrv16 データベースオプション

停止時、またはサーバによってスケジュールされたチェックポイントの結果として、チェックポイント実行後に、トランザクションログをトランケート (削除) します。

トランケーションによって、トランザクションのログの肥大化が自動的に制限されます。チェックポイントの頻度は、`CHECKPOINT_TIME` および `RECOVERY_TIME` オプションによって制御できます (コマンドラインでも定義できます)。

`-m` オプションは、大量のトランザクションを高速に処理する必要がある場合や、トランザクションログの信頼性が低くてリカバリやレプリケーションができない場合に役立ちます。このオプションを選択すると、データベースファイルを含むデバイスのメディア障害に対して無防備な状態になります。

データベースファイルの断片化を防ぐためには、トランザクションログをデータベースそのものとは別のデバイスまたはパーティションに保管してください。

このオプションは、`-m` サーバオプションと同じですが、現在のデータベースまたは `database-file` コマンドライン変数で指定されたデータベースにのみ適用されません。

---

**注意：** 本質的に、レプリケーションはトランザクションログ情報に依存するため、レプリケートされるデータベースでは `-m` オプションを使用しないでください。この理由から、`-m` オプションをマルチプレックスデータベースで使用しないでください。

---

### 参照：

- `dbbackup` データベース管理ユーティリティ (5 ページ)
- `dbfhide` データベース管理ユーティリティ (9 ページ)



## **-n iqsrv16 データベースオプション**

データベースの別名またはニックネームを指定します。

構文

**-n name**

備考

ニックネームを使うと、接続が単純化されます。Open Client の場合、**-n** のニックネームは `interfaces` ファイル内のエントリと同じである必要があります。

---

**注意：** データベースサーバはいくつかのデータベースをロードできるので、データベース名を使用して、各データベースを区別します。しかし、SAP Sybase IQ サーバで実行するデータベースは1つだけにしてください。2つのデータベースを実行する必要がある場合は、2つの SAP Sybase IQ データベースを別々のポートで起動します。

---

デフォルトでは、データベースはパスと拡張子を除いたファイル名を名前として受け取ります。たとえば、`c:\¥sybase¥IQ-16_0¥demo¥iqdemo.db` でサーバを起動するときに、**-n** スイッチを指定しないと、サーバの名前は `iqdemo` になります。デフォルト名の使用を避けるには、常にサーバ名を指定してください。

命名規則については、**-n** サーバオプションの説明を参照してください。

---

**注意：** **-n** スイッチは2種類あり、位置によって意味が変わります。**-n** がデータベースファイル名の後以外にある場合は、サーバ名を指定します。データベースファイル名の後に **-n** を使用した場合、このスイッチはデータベーススイッチになります。

---

参照：

- **-n iqsrv16** データベースサーバオプション (187 ページ)

## **-r iqsrv16 データベースオプション**

データベースサーバで起動されるすべてのデータベースを強制的に読み取り専用にします。

データベースに変更を加えることはできません。データベースサーバはデータベースファイルを変更しません。

**start\_iq** データベースサーバ起動ユーティリティ

## **-sm iqsrv16** データベースオプション

読み取り専用のミラーデータベースにアクセスするために使用できる代替データベースサーバ名を指定します。

構文

**-smname**

備考

代替サーバ名は、データベースサーバがデータベースのミラーとして動作している場合にのみアクティブになります。 コマンドラインオプションの **-sm** と **-sn** を使用することにより、アプリケーションは、どの物理サーバがプライマリまたはミラーとして実行されているかがわからない場合も、プライマリサーバまたはミラーサーバ上のデータベースに常に接続できます。

## **-sn iqsrv16** データベースオプション

データベースサーバ上で実行されている単一のデータベースの代替サーバ名を指定します。

構文

**-snname**

備考

データベースサーバは、特定のデータベースサーバについて、複数のサーバ名を受信するように設定できます。 実際のサーバ名以外のサーバ名は、代替サーバ名と呼ばれ、データベースサーバ上で実行されている特定のデータベースに固有のもので、代替サーバ名を使用して接続するクライアントは、代替サーバ名を指定したデータベースのみに接続できます。 代替サーバ名は、ネットワーク上で一意でなければなりません。 重複すると、データベースを起動できません。 データベースがサーバコマンドで起動され、代替サーバ名が一意でない場合、サーバの起動は失敗します。

## **-xp iqsrv16** データベースオプション

データベースミラーリングの使用時に、運用サーバがパートナーや決定者に接続できるようにする情報を提供します。

構文

```
iqsrv16 [ server-options ] database-file -xp on
```

備考

**-xp** データベースオプションは、*database-file* の後に指定する必要があり、そのデータベースだけに適用されます。

**-xp** オプションで指定した接続パラメータが無効であり、サーバで複数のデータベースが実行されている場合、ミラーデータベースの起動に失敗し、再接続は試行されません。ミラーデータベースが、データベースサーバで実行されている唯一のデータベースである場合は、データベースサーバが起動しません。

- **on** – ミラーリング情報がデータベースに格納されていても、データベースのミラーリングを使用できるのは、データベースサーバの起動時に **-xp** オプションを指定した場合だけです。**-xp on** オプションを指定した場合、**-xp** オプションで他のミラーリングオプションを指定することはできません。値 **off** はサポートされていません。データベースでデータベースミラーリング設定を定義するには、次の文を使用します。

- **CREATE MIRROR SERVER**
- **SET MIRROR OPTION**

**-xp on** を指定するときには、**-n** オプションを使用して、ミラーリングシステム内のデータベース名も指定します。また、**-su** オプションも使用して、ユーティリティデータベースのパスワードを指定することをおすすめします。これにより、ユーティリティデータベースを使用してデータベースサーバをシャットダウンしたり、必要に応じてミラーサーバを強制的にプライマリサーバにしたりできるようになります。

**start\_iq** データベースサーバ起動ユーティリティ

## stop\_iq データベース停止ユーティリティ

UNIX と Linux のプラットフォームでは、**stop\_iq** ユーティリティを使用してデータベースサーバを停止できます。

**stop\_iq** を使用してサーバを停止し、サーバへのすべてのユーザ接続を閉じます。

**stop\_iq** は、ユーザ接続またはロード処理のステータスを考慮せずに、サーバを停止します。通常、サーバが 1 つ以上のクライアントに接続している間はサーバを停止しないでください。

---

**注意：** **dbstop** ユーティリティではより細かなレベルの制御が可能で、ユーザ接続の状況に基づいてサーバを停止するかどうかを制御するオプションが用意されています。

---

### サーバを停止または再起動する状況

---

いくつかの状況では、サーバを停止して、再起動する必要があります。

次に例を示します。

- SAP Sybase IQ の新しいバージョンをインストールするとき。
- サーバコマンドラインオプションをリセットするとき。
- サーバ全体を対象とする少数のデータベースオプションを有効にするとき。
- オペレーティングシステムセッションを終了する前。

### データベースサーバを停止する方法

---

SAP Sybase IQ のデータベースサーバを停止するときの推奨方法は次のとおりです。

- UNIX では、オペレーティングシステムのコマンドラインで **stop\_iq** ユーティリティを使用します。

**stop\_iq** を実行すると、次のメッセージが表示されます。

```
"Please note that 'stop_iq' will shut down a server completely without regard for users, connections, or load process status. For more control, use the 'dbstop' utility, which has options that control stopping servers based on active connections."
```

- Windows では、データベースサーバ画面で[シャットダウン]をクリックするか、システムトレイの IQ アイコンを右クリックして[終了]を選択します。

## stop\_iq データベース停止ユーティリティ

- Windows でサーバがサービスとして実行されている場合は、コントロールパネルでサービスマネージャを開きます。サービスを選択して[停止]をクリックします。

1つ以上のクライアントがサーバに接続している間は、サーバの停止は避けてください。停止しようとする、コミットされていないトランザクションが失われることを通知する警告が表示されます。すべてのクライアントを切断するか閉じ、再度実行してください。

データベースサーバは次の方法で停止できます。

- オペレーティングシステムのコマンドラインで、適切なパラメータを指定して **DBSTOP** コマンドを発行します。そのサーバを起動したときと同じパラメータを使用します。正しい接続パラメータを指定しないと、**DBSTOP** はサーバに接続できないため、サーバをシャットダウンできません。
- Interactive SQL ウィンドウまたはコマンドファイルで、**STOP ENGINE** コマンドを発行し、指定したデータベースサーバを停止します。
- UNIX では、そのデータベースサーバを起動したウィンドウで、次のコマンドを入力します。

```
q
```

このコマンドは、入力を別のデバイスにリダイレクトしている場合や **start\_iq** を使用してサーバを起動した場合には、機能しません。 **iqsrv16** でのみ機能します。

- UNIX の **cron** ジョブまたは **at** ジョブでは、適切な **-stop** オプションを指定して **stop\_iq** を使用します。指定されたパラメータに基づいて、**cron** ジョブまたは **at** ジョブを起動するユーザに関連する、1台または複数のサーバが停止されます。ユーザは、サーバを起動したユーザと同じでなければなりません。オペレータに対してメッセージは表示されず、オペレータの操作は不要です。このようなジョブで **stop\_iq** を使用するには、適切な **-stop** オプションを使用してユーティリティを指定します。

```
stop_iq -stop one
```

**cron** ジョブまたは **at** ジョブを開始するユーザ ID によって起動されているサーバが1台しかない場合、**-stop one** を設定すると、1台のサーバが停止します。

```
stop_iq -stop all
```

**-stop all** を設定すると、**cron** ジョブまたは **at** ジョブを起動するユーザ ID によって起動されているすべてのサーバが停止します。

---

**注意：** **cron** 文では **stop\_iq** 実行プログラムのフルパス名を指定する必要があります。

---

**例 — stop\_iq を使用したサーバの停止**

次の例では、UNIX オペレーティングシステムのコマンドラインで **stop\_iq** ユーティリティを使用して SAP Sybase IQ サーバを停止し、そのサーバへのすべてのユーザ接続を閉じます。

**stop\_iq** コマンドを発行すると、SAP Sybase IQ によって、自分以外のユーザが所有しているすべてのサーバがリストされ、次に、自分が所有しているサーバがリストされます。サーバを停止するかどうかを尋ねるメッセージが表示されます。次に例を示します。

```
% stop_iq

Checking system for IQ 16 Servers ...
The following 2 server(s) are owned by other users.
##      Owner      PID      Started      CPU_Time      Additional Information
-----
          handari 19895    15:43:44    183:38
start_iq @iqdemo.cfg iqdemo.db -gn 105 -o /server1/users/surya/
IQ-16_0/logfiles/surya_ibm2.001.srvlog -hn 8          pamela    409802
18:05:02      0:05    SVR:ibm1_iqdemo2 DB:iqdemo
PORT:2678/ibm1/users/sybase/iq160/IQ-16_0/bin64/iqsrv16 @iqdemo.cfg
iqdemo.db -ti 4400 -gn 25 -o /ibm1/users/sybase/iq160/IQ
16_0/logfiles/ibm64qa iq
The following 1 server(s) are owned by 'kermit'
##      Owner      PID      Started      CPU_Time      Additional Information
-----
1:      kermit    422034   15:11:37      0:07    SVR:myserver_iqdemo
DB:iqdemo PORT:2638 /myserver/users/sybase/iq160/IQ-16_0/bin64/
iqsrv16
@iqdemo.cfg iqdemo.db -ti 4400 -gn 25 -o /myserver/users/sybase/
iq160/IQ-
16_0/logfiles/myserver_iq
start_iq -c 32m -gd all -gm 10 -gn 25 -gp 4096 -ti 4400 -tl 300
@iqdemo.cfg

--
Please note that 'stop_iq' will shut down a server completely
without regard for users connections or load processes status.
For more control, use the 'dbstop' utility, which has options
that control stopping servers based on active connections.

Do you want to stop the server displayed above <Y/N>?
```

サーバを停止するには、 Y (yes) と入力します。次のようなメッセージが表示されます。

```
Shutting down server (422034) ...
Checkpointing server (422034) ...
Server shutdown.
```

サーバの実行を続けるには、 N (no) と入力します。N を入力すると、システムプロンプトに戻り、IQ はサーバを停止しません。

## stop\_iq データベース停止ユーティリティ

実行中のサーバの中に、自分自身のユーザ ID を使用して起動したものがない場合は、SAP Sybase IQ は、他のユーザによって実行されているサーバの情報を表示した後に、次のようなメッセージを表示します。

```
There are no servers owned by 'kermit'
```

### 例 — Interactive SQL からのサーバの停止

次の例では、Interactive SQL からサーバを停止します。

```
STOP ENGINE Ottawa UNCONDITIONALLY
```

オプションの **UNCONDITIONALLY** キーワードを指定すると、データベースとの接続がある場合でも、そのデータベースサーバを停止します。

**注意：**サーバで実行されているデータベース (utility\_db データベースを含む) の 1 つに DBA として接続している場合、または **-gk ALL** オプションを指定してサーバを起動した場合は、Interactive SQL からそのサーバを停止できます。

## サーバを停止するために必要なパーミッション

サーバの起動時に **-gk** オプションを使用すると、ユーザが **DBSTOP** または **STOP ENGINE** を使用してサーバを停止するために必要なパーミッションレベルを設定できます。

必要なパーミッションレベルのデフォルトは、SERVER OPERATOR システム権限を必要とする **DBA** ですが、この値を **ALL** または **NONE** に設定することもできます。**NONE** に設定すると、SERVER OPERATOR システム権限を持つユーザも **STOP ENGINE** を実行できません。運用環境では、DBA のみがデータベースサーバを停止できるようにしてください。

UNIX コマンドラインで **stop\_iq** を実行するか、Windows プラットフォームで [シャットダウン] を選択すると、サーバが起動されたマシン上のサーバとデータベースも停止できます。

## データベースサーバの停止 (stop\_iq)

コマンドラインから **stop\_iq** を実行します。

1. 次の形式でコマンドを発行します。

```
stop_iq [ -cleanup ] [ -stop [ one | all ] ] [ -user <user_name> ] [ -version [ 12 | 15 | all ] > ] [ -wait <seconds> ]
```

2. サーバを停止するかどうかを確認するメッセージが表示されたら、Y (yes) と応答します。



3. サーバに1つまたは複数のクライアントが接続している場合、コミットされていないトランザクションが失われる可能性があることを警告するメッセージに応答します。

次の例では、**stop\_iq** を対話的に使用して、実行中のすべてのサーバをリストし、その内の1台を停止する方法を示しています。

```
% stop_iq

Checking system ...
The following 1 server(s) are owned by 'TEST'

##      Owner      PID   Started   CPU_Time   Additional Information
--      -
1:    TEST      22399  08:56:39    1:43   SVR:QA_sun7qa DB:iqdemo PORT:
8888
/sun7qa1/users/QA/090513/IQ-16_0/bin64/iqsrv16 @iqdemo.cfg iqdemo.db
-ti 4400
--

Please note that 'stop_iq' will shutdown a server completely
without regard for users connections or load processes status.
For a finer level of detail the utility 'dbstop' has the options
to control whether a server is stopped based on active
connections.

Do you want to stop the server displayed above <Y/N>? Y

Shutting down server (22399) ...
Checkpointing server (22399) ...
Server shutdown.
```

## stop\_iq オプション

次の表は、**stop\_iq** ユーティリティの利用可能なオプションを示します。

表 35 : stop\_iq オプション

スイッチ	説明
<b>-cleanup</b>	Linux 上の孤立した SAP Sybase IQ プロセスを削除する。
<b>-stop [ one   all ]</b>	ユーザと <b>stop_iq</b> の対話を削除する。すべての質問に対する応答が yes であると見なす。

## stop\_iq データベース停止ユーティリティ

スイッチ	説明
-user	次の2つの機能を実行する。 <ul style="list-style-type: none"><li>省略や置換が原因で、現在のユーザが所有するサーバやエージェントが <b>stop_iq</b> で見つからない場合、システムマネージャはプロセステーブルにある名前と ID を <b>-user</b> 引数で指定して、<b>stop_iq</b> ユーティリティでサーバ/エージェントを停止できる。</li><li>ルート権限があるユーザは、別のユーザのサーバやエージェントを、そのユーザとしてログインしなくても停止できる。<b>stop_iq</b> ユーティリティにはスーパーユーザ (su) またはルート権限がないので、権限のないユーザは、別のユーザが所有するサーバを停止できない。</li></ul>
-version	使用している SAP Sybase IQ のメジャーバージョンを指定する。デフォルトは現在のメジャーバージョン (16.0)。
-wait	タイムアウトが経過するまでサーバの停止を待つ時間を指定する。

## cron または at ジョブでのサーバの停止

**cron** または **at** ジョブで **stop\_iq** を使用する場合は、適切な **-stop** オプションを使用します。

表 36 : stop\_iq -stop オプション

オプション	説明
-stop one	<b>cron</b> または <b>at</b> ジョブを開始するユーザ ID によって起動されているサーバが 1 台しかない場合、1 台のサーバを停止する。これによって、複数のサーバが実行されている場合、対象と異なるサーバを停止することがなくなる。
-stop all	サーバを起動するユーザ ID によって起動されたすべてのサーバが停止される。

次に例を示します。

```
stop_iq -stop one
```

```
stop_iq -stop all
```

**注意：** **cron** 文では、**stop\_iq** 実行可能ファイルの完全パスを指定する必要があります。

## 長いパスのサーバ

オペレーティングシステムによっては、サーバのパスが 74 文字を超える場合、**stop\_iq** ユーティリティが、実行中のサーバをレポートしないことがあります。

対処方法として、**ps -ef** コマンドを使用して長いパスのサーバを表示します。例：

## stop\_iq データベース停止ユーティリティ

```
ps -ef|grep myserver
rsmithson 1133      1    0 07:04:32 ?                223:35
/sunsys1234/users/rsmithson/mybigtest1234_withdeletion
_allcol/IQ-16_0/bin64/myserver
rsmithson 2046    862    0 10:02:30 pts/3            0:00 grep
myserver
```

**stop\_iq** データベース停止ユーティリティ

## 付録： **dbisqlc** Interactive SQL Classic ユーティリティ (廃止)

Interactive SQL Classic (**dbisqlc**) ユーティリティは、SQL 文をデータベースに対して実行します。このユーティリティは廃止されました。

このユーティリティは、Java で実装されていないことを除けば、Interactive SQL ユーティリティに類似しています。Java を使用しないことは、リソースに制限があるコンピュータにユーティリティを展開する場合に利点があります。

---

**注意：** Interactive SQL Classic は廃止されましたが、今のところ削除される予定はありません。Interactive SQL Classic は、展開用のライトウェイトツールとして、SQL スクリプトの実行に関して下位互換性を提供します。Interactive SQL Classic は、Interactive SQL がサポートする機能のすべてをサポートしているわけではありません。SAP Sybase IQ の現在のバージョンで使用可能な機能の中には、Interactive SQL Classic でサポートしていないものもあります。代わりに Interactive SQL ユーティリティを使用してください。

---

Interactive SQL Classic ユーティリティは、Microsoft Windows と UNIX でサポートされています。

### Interactive SQL Classic の構文

---

コマンドプロンプトから Interactive SQL Classic を起動します。

```
dbisqlc [ options ] [ dbisqlc-command | command-file ]
```

*dbisqlc-command* を指定している場合、Interactive SQL Classic によってコマンドが実行されます。コマンドファイル名を指定することもできます。*dbisqlc-command* または *command-file* の引数が指定されていないと、Interactive SQL Classic は対話型モードになります。このモードでは、コマンドをコマンドウィンドウに入力できません。

## Interactive SQL Classic のオプション

コマンドプロンプトから Interactive SQL Classic を起動するときに、以下のオプションを指定します。

表 37 : Interactive SQL Classic のオプション

オプション	説明
<b>-c</b> "key-word=value; ..."	<p>接続パラメータを指定する。接続パラメータを指定しないと、環境変数 SQLCONNECT が使用される。Interactive SQL が接続できない場合は、ダイアログボックスが表示されるので、そこに適切なパラメータを入力する。</p> <p><b>注意：</b> 設定をデフォルトのまま使用するのではなく、必ず <b>Interactive SQL Classic</b> の接続パラメータを指定してください。接続パラメータは、コマンドラインまたは初期化ファイル (UNIX では .odbc.ini、Windows では odbc.ini など) で指定します。たとえば、サーバ上で複数のデータベースを起動する場合は、データベース名を指定し、サブネットが設定されているネットワークの場合は、通信プロトコルパラメータとホスト番号を指定します。</p>
<b>-d</b> <i>delimiter</i>	コマンドデリミタを指定する。デフォルトでは、デリミタはセミコロン。
<b>-q</b>	クワイエットモード - 出力メッセージは表示されない。このオプションは、コマンドまたはコマンドファイルを使用して Interactive SQL Classic を起動したときにのみ役立つ。
<b>-r</b>	ストアードプロシージャの定義済み結果セットが実際の結果セットと一致しない場合に、“Not enough fields allocated in sqllda” というエラーを返す。このオプションは、ストアードプロシージャのクエリを実行するときに役立つことがある。
<b>-x</b>	構文のみをチェックする。コマンドをスキャンするが、実行はしない。このオプションは、長いコマンドファイルの構文エラーをチェックする場合に有用。

## ファンクションキーと特殊キー (UNIX)

ファンクションキーと特殊キーを使用して、データを移動したり、データベーステーブルをリストしたりします。

表 38 : UNIX での Interactive SQL Classic のファンクションキーと特殊キー

ファンクションキー	説明
F5	データウィンドウ内で、データを 1 カラム分左へ移動する。
Shift + F5	データを 1 文字分左へ移動する。
F6	データを 1 カラム分右へ移動する。
Shift + F6	データを 1 文字分右へ移動する。
F7	データベース内のテーブルのリストを表示する。上矢印キーや下矢印キーを使用すると、テーブル名がスクロールされ、強調表示になるテーブル名が変わる。リストが表示されている状態で [Enter] キーを押すと、現在のテーブル名がコマンドウィンドウのカーソル位置に挿入される。リストが表示されている状態で [F7] キーを押すと、強調表示されたテーブルのカラムのリストが表示される。この場合も、[Enter] キーを使って強調表示されるカラム名を選択し、それをコマンドウィンドウのカーソル位置に挿入できる。
Ctrl + PgUp	データの先頭に移動する。
Ctrl + PgDn	データの最後に移動する。

## ファンクションキーと特殊キー (Windows)

ファンクションキーと特殊キーを使用して、データを移動したり、データベーステーブルをリストしたりします。

表 39 : Windows での Interactive SQL Classic のファンクションキーと特殊キー

ファンクションキー	説明
F5	データウィンドウ内で、データを 1 カラム分左へ移動する。
Shift + F5	データを 1 文字分左へ移動する。

ファンクションキー	説明
F6	データを1カラム分右へ移動する。
Shift + F6	データを1文字分右へ移動する。
F7	データベース内のテーブルのリストを表示する。上矢印キーや下矢印キーを使用すると、テーブル名がスクロールされ、強調表示になるテーブル名が変わる。リストが表示されている状態で [Enter] キーを押すと、現在のテーブル名がコマンドウィンドウのカーソル位置に挿入される。リストが表示されている状態で [F7] キーを押すと、強調表示されたテーブルのカラムのリストが表示される。この場合も、[Enter] キーを使って強調表示されるカラム名を選択し、それをコマンドウィンドウのカーソル位置に挿入できる。
F9	コマンドウィンドウに表示されているコマンドを実行する。この操作は、[実行] ボタンをクリックしても実行できる。
F10	ウィンドウの上部にあるメニューを開く。
PgUp	1 ページ上のデータへ移動する。
PgDn	1 ページ下のデータへ移動する。
Ctrl + PgUp	データの先頭に移動する。
Ctrl + PgDn	データの最後に移動する。

## コマンド呼び出しキー (Windows)

キーシーケンスを使用して、以前使用したコマンドを呼び出します。

表 40 : Interactive SQL Classic の呼び出しキー

キーシーケンス	説明
Ctrl + r	command recall window を起動する。
Ctrl + p	1 つ前に実行したコマンドを呼び出す。取得されたコマンドは、コマンドウィンドウに表示される。
Ctrl + n	1 つ後に実行したコマンドを呼び出す。



## 索引

## C

CBSize 接続パラメータ 104

CLEAR 文

構文 51

CommBufferSize 接続パラメータ 104

CONFIGURE 文

構文 52

CONNECT 文

構文 53

CP874toUTF8 ユーティリティ 3

## D

DB 領域ヘッダ 109

dbbackup ユーティリティ 5

dbfhide

説明 9

dbfhide ユーティリティ

構文 9

dbinfo ユーティリティ 11

dbinit ユーティリティ 113

dbisql

オプション 66

データベースへの接続 53

dbisql ユーティリティ 13

dbisqlc ユーティリティ 241

dblocate ユーティリティ 75

dblog ユーティリティ 77

コマンドライン 77

dbping ユーティリティ 81

dbstop ユーティリティ 85

使用状況 85

dbtran ユーティリティ 87

構文 87

終了コード 87

dbvalid ユーティリティ 93

終了コード 96

DEFAULT\_ISQL\_ENCODING オプション

説明 73

DisableMultiRowFetch 接続パラメータ 106

DISCONNECT 文

構文 56

DMRF 接続パラメータ 106

## E

Embedded SQL 141, 142

概要 139

ENC 接続パラメータ

説明 102

Encryption 接続パラメータ

説明 102

## F

File Hiding ユーティリティ

構文 9

説明 9

FIPS 141

## I

INI ファイル

dbfhide による単純暗号の追加 9

Interactive SQL

OUTPUT 文の構文 57

ファイルの読み書き用のコードページを  
指定する 73

Interactive SQL Classic ユーティリティ 241

コマンド呼び出しキー 244

ファンクションキー 243

構文 241

Interactive SQL オプション

DEFAULT\_ISQL\_ENCODING 73

Interactive SQL ユーティリティ 13

インデント 25

オプション 14, 73

キーボードショートカット 31

クエリエディタ 43

クワイエットモード 17

コメント 23

[接続] ウィンドウ 18

## 索引

ソース制御の統合 47  
プランビューア 45  
メインウィンドウ 29  
構文 14  
接続パラメータ 17  
複数の結果セット 33  
文の実行 22  
iqdscp ユーティリティ 97  
iqdsn ユーティリティ  
説明 99  
iqinit ユーティリティ 113  
iqmsgnum 起動スイッチ 176  
iqmsgsz 起動スイッチ 177  
iqnumbercpus  
サーバスイッチ 178  
iqocscfg ユーティリティ 125  
iqpartition 起動スイッチ 179  
iqrlvmem 起動スイッチ 179  
iqsqlpp の長さの制限 142

## L

LivenessTimeout 接続パラメータ 106  
LogFile  
接続パラメータ 104  
LTM  
トランザクションログのオプション 77  
LTO 接続パラメータ 106

## O

ODBC データソース  
iqdsn を使用した作成 99  
ON\_ERROR オプション  
説明 74  
OUTPUT 文  
SQL 構文 57

## P

PARAMETERS 文  
構文 61  
ping  
ユーティリティ 81

## R

READ 文  
構文 62

## S

SET CONNECTION 文  
構文 65  
SET OPTION 文  
構文 66  
SET TEMPORARY OPTION 文  
構文 66  
SQL プリプロセッサ  
コマンドライン 139  
コマンドラインオプション 141  
概要 139  
構文 139  
SQL 文  
OUTPUT 構文 57  
SQL92 141  
sqlpp  
オプション 139  
コマンドライン 139  
コマンドラインオプション 139  
パラメータ 139  
概要 139  
構文 139  
START DATABASE 文  
構文 68  
START ENGINE 文  
構文 69  
start\_iq  
サーバオプション 146  
データベースオプション 222  
構文 145  
STOP DATABASE 文  
構文 71  
STOP ENGINE 文  
構文 72  
stop\_iq  
ユーティリティ 233  
構文 236  
使用状況 233

## U

UTF8 照合 3

## え

エスケープ文字  
OUTPUT SQL 文 57

## お

オプション  
dbisql オプションの設定 52  
DBISQL オプションの設定 52  
DEFAULT\_ISQL\_ENCODING 73  
ON\_ERROR 74  
temporary の設定 66

## か

カタログストア  
キャッシュサイズの設定 149  
検証 93

## き

キャッシュサイズ  
カタログストア用の設定 149

## く

クライアント側  
Encryption [ENC] 接続パラメータ 102

## こ

コードページ  
DEFAULT\_ISQL\_ENCODING オプション  
73  
コマンドファイル  
パラメータ 61  
コマンドライン  
データベースサーバ 145  
設定ファイル内 148  
コマンドラインユーティリティ  
dbfhide 構文 9  
ファイル隠蔽構文 9  
コマンドリミッタ 15

## さ

サーバ  
デフォルト 213

ライブバックアップ 5

再利用 233

サーバの再利用 233

## す

スタックサイズ  
SQL Anywhere スレッド 168  
スレッド  
スタックサイズの制限 168

## せ

セキュリティ  
dbfhide ユーティリティ 9  
Encryption [ENC] 接続パラメータ 102  
File Hiding ユーティリティ 9  
設定ファイルへの単純暗号の追加 9

## そ

ソフトウェア  
dbinfo 11  
dblog 77

## た

タイ語 3

## て

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 57  
データソース  
iqdsn を使用した ODBC データソースの作  
成 99  
データのエキスポート  
テーブルからファイルへのデータのエク  
スポート 57  
データのロード 3  
データベース  
ページの使用状況 11  
起動 68  
作成 113  
情報 11

## 索引

- 停止 71
- データベースサーバ
  - コマンドライン 145
  - 起動 69, 145
  - 停止 72, 85, 233
- データベースの停止 71
- データベースページ
  - サイズの表示 11
- データベースユーティリティ
  - dbfhide 9
  - File Hiding 9
  - iqdsn 99
  - データソース 99
  - ログの変換 87
- データベース情報 11
- テーブル
  - テーブルからファイルへのデータのエク  
スポート 57
- デフォルトサーバ 213

## と

- トランザクションログ
  - dblog 77
  - バックアップ 5
  - ユーティリティ 77

## は

- ハイパースレディング
  - サーバスイッチ 178
- バックアップ
  - ライブ 5
  - 実行中のサーバ上 5
- バックアップユーティリティ 5
- バッファキャッシュ
  - パーティショニング 179

## ふ

- ファイル
  - テーブルからファイルへのデータのエク  
スポート 57

## へ

- ページ
  - データベースファイル内での使用状況の  
表示 11

## め

- メッセージログ
  - iqmsgnum サーバオプション 176
  - iqmsgsz サーバオプション 177
  - アーカイブの数の指定 176
  - サイズの指定 177

## ゆ

- ユーティリティ
  - CP874toUTF8 3
  - DB 領域ヘッダ 109
  - dbfhide 9
  - dbfhide 構文 9
  - dbinfo 11
  - dbinit 113
  - dblocate 75
  - dblog 77
  - dbping 81
  - dbstop 85
  - dbvalid 93
  - File Hiding 9
  - Interactive SQL 13
  - Interactive SQL Classic 241
  - iqdscp 97
  - iqinit 113
  - iqocscfg 125
  - isql 127
  - SQL プリプロセッサ 139
  - stop\_iq 233
  - サーバ検出ユーティリティ 75
  - データソース 99
  - トランザクションログユーティリティ 77
  - バックアップ 5
  - ファイル隠蔽構文 9
  - 検証ユーティリティ 93
  - 情報 11

## ら

- ライブバックアップ 5

## り

- リソース計画
  - iqnumbercpus スイッチ 178
- リターンコード
  - 検証ユーティリティ (dbvalid) 96

## れ

- レプリケーション
  - Replication Server 77

## ろ

- ロードパフォーマンス
  - iqpartition サーバオプション 179
- ログファイル 77
- ロック競合
  - iqpartition サーバオプション 179

