Quick Reference

About this Quick Reference

This Quick Reference shows SQL*Plus command syntax. For detailed information on each command, refer to the SQL*Plus User's Guide and Reference.

This Quick Reference has the following topics:

- Alphabetic List of SQL*Plus Commands
- Starting and Leaving SQL*Plus
- Starting Up and Shutting Down a Database
- Entering and Executing Commands
- Manipulating SQL, SQL*Plus and PL/SQL Commands
- Formatting Query Results
- Accessing Databases
- Miscellaneous

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.
Alphabetic List of SQL*Plus Commands

@{url | file_name[.ext]} [arg ...]
@@ { url | file_name[.ext] } [arg ...]
/ (slash)

ACC[CEPT] variable [NUM[BER] | CHAR | DATE | BINARY_FLOAT | BINARY_DOUBLE] [FOR[MAT] format] [DEF[ault] default] [PROMPT text | NOPR[OMPT]] [HIDE]

A[PPEND] text

ARCHIVE LOG LIST

ATTRIBUTE [type_name.attribute_name [option...]]

BRE[AK] [ON report_element [action [action]]] ...

BTI[TLE] [printspec [text | variable] ...] | [ON | OFF]

C[HANGE] sepchar old [sepchar new [sepchar]]

C[LEAR] option ...

C[OLUMN] [{column | expr} [option ...]]

C[OMP]UTE [function [LAB[EL] text] ... OF {expr | column | alias} ...ON {expr | column | alias | REPORT | ROW} ...]

C[ONN]ECT [{logon | / | proxy} AS {SYSASM | SYSBACKUP | SYSDBA | SYSDG | SYSOPER | SYSRAC | SYSKM} [edition=value] ]

C[OPY] {FROM database | TO database | FROM database TO database} {APPEND | CREATE | INSERT | REPLACE} destination_table{{column, column, column, ...}}

USING query

D[EFIN]E [variable] | [variable = text]

DEL [n | n m | n * | n LAST | * | * n | * LAST | LAST]

D[E]SCRI[B]E {{schema.}object[@connect_identifier]}

D[ISC][O][N][N][N][E][C][T]

E[D]I[T] [file_name[.ext]]

E[XE]C[U][T][E] statement

{EXIT | QUIT} [SUCCESS | FAILURE | WARNING | n | variable | :BindVariable] [COMMIT | ROLLBACK]

G[ET] [FILE] file_name[.ext] [LIST | NOLIST]

H[E]LP | ? [topic]

H[I][S][T][O][R][Y] [n RUN | EDIT | DEL[ETE]] | [CLEAR | LIST]

H[O][S][T] [command]
I[NPUT] [text]
L[ISt] [n | n m | n * | n LAST | * | * n | * LAST | LAST]
PASSW[ORD] [username]
PAU[SE] [text]
PRINT [variable ...]
PRO[MPT] [text]
{QUIT | EXIT} [SUCCESS | FAILURE | WARNING | n | variable | :BindVariable]
[COMMIT | ROLLBACK]
RECOVER {general | managed} [BEGIN BACKUP | END BACKUP]
REM[ARK]
REP[FOOTER] [PAGE] [printspec [text | variable ...]] | [ON | OFF]
REPH[HEADER] [PAGE] [printspec [text | variable ...]] | [ON | OFF]
R[UN]
SAV[E] [FILE] file_name[.ext] [CREATE | REPLACE | APPEND] [FILE]
SET system_variable value
SHO[W] [option]
SHUTDOWN [ABORT | IMMEDIATE | NORMAL | TRANSACTIONAL [LOCAL]]
SPO[OL] [filename[.ext] [CREATE | REPLACE | APPEND]] [OFF | OUT]
STA[RT] {url | file_name[.ext]} [arg ...]
STARTUP db_options | cdb_options | upgrade_options
STORE (SET) file_name[.ext] [CREATE | REPLACE | APPEND]
TIM[ING] [START text | SHOW | STOP]
TTI[TLE] [printspec [text | variable ...]] | [ON | OFF]
UNDEF[INE] variable ...
VAR[IABLE] [variable [type][=value]]
WHENEVER OSERROR {EXIT [SUCCESS | FAILURE | n | variable | :BindVariable]
[COMMIT | ROLLBACK] | CONTINUE[COMMIT | ROLLBACK | NONE]}
WHENEVER SQLERROR {EXIT [SUCCESS | FAILURE | WARNING | n | variable
| :BindVariable] [COMMIT | ROLLBACK] | CONTINUE [COMMIT | ROLLBACK | NONE]}
XQUERY xquery_statement
Starting and Leaving SQL*Plus

Use the following commands to log in to and out of SQL*Plus.

SQLPLUS [option] [logon | / NOLOG] [start]

where option has the following syntax:

-HELP | VERSION | [-C[OMPATIBILITY] x.y[z]] [-F[ast]] [-M[ARKUP]
markup_option] [-L[OGON]] [-NOLOGINTIME] [-R[ESTRICT] {1 | 2 | 3}] [-S[ILENT]] [-AC]

where markup_option consists of:

- csv_option
- html_option

csv_option has the following syntax:

CSV {ON|OFF} [DELIMITER character] [QUOTE {ON|OFF}]

html_option has the following syntax:

HTML {ON|OFF} [HEAD text] [BODY text] [TABLE text] [ENTMAP {ON|OFF}] [SPOOL {ON|OFF}] [PRE[FORMAT] {ON|OFF}]

where logon has the following syntax:

{username[/password [@connect_identifier] | /} [AS {SYSASM |SYSBACKUP | SYSDBA |SYSDG |SYSOPER |SYSRAC |SYSKM}] [edition=value]

and where start has the following syntax:

@{url | file_name[.ext]} [arg ...]

(EXIT | QUIT) [SUCCESS | FAILURE | WARNING | n | variable | :BindVariable] [COMMIT | ROLLBACK]

Commits or rolls back all pending changes, logs out of Oracle, terminates SQL*Plus and returns control to the operating system.

Starting Up and Shutting Down a Database

Starting up and shutting down a database requires DBA privileges.

STARTUP db_options | cdb_options | upgrade_options

where db options has the following syntax:

[FORCE] [RESTRICT] [PFILE=filename] [QUIET] [ MOUNT [dbname] | [ OPEN [open_db_options] [dbname] ] | NOMOUNT ]

where open_db_options has the following syntax:
where \( cdb\_options \) has the following syntax:

\( root\_connection\_options \) | \( pdb\_connection\_options \)

where \( root\_connection\_options \) has the following syntax:

\( \text{PLUGGABLE DATABASE} \ pdb\_name \ [\text{FORCE}] \ [\text{RESTRICT}] \ [\text{OPEN} \ \{\text{open\_pdb\_options}\}] \)

where \( pdb\_connection\_options \) has the following syntax:

\( [\text{FORCE}] \ [\text{RESTRICT}] \ [\text{OPEN} \ \{\text{open\_pdb\_options}\}] \)

where \( \text{open\_pdb\_options} \) has the following syntax:

\( \text{READ WRITE} \) | \( \text{READ ONLY} \)

and where \( \text{upgrade\_options} \) has the following syntax:

\( [\text{PFILE}=\text{filename}] \ \{\text{UPGRADE} \ | \ \text{DOWNGRADE} \} \ [\text{QUIET}] \)

Starts an Oracle Database instance with several options, including mounting and opening a database.

\( \text{SHUTDOWN} \ [\text{ABORT} \ | \ \text{IMMEDIATE} \ | \ \text{NORMAL} \ | \ \text{TRANSACTIONAL} \ [\text{LOCAL}]] \)

Shuts down a currently running Oracle instance, optionally closing and dismounting a database.

### Entering and Executing Commands

Use the following commands to execute and collect timing statistics on SQL commands and PL/SQL blocks.

\( / \) (slash)

Executes the most recently executed SQL command or PL/SQL block which is stored in the SQL buffer. Does not list the command. Use slash (/) at the command prompt or line number prompt in SQL*Plus command line.

\( \text{EXECUTE} \ \text{statement} \)

Executes a single PL/SQL statement or runs a stored procedure.

\( \text{RUN} \)

Lists and executes the most recently executed SQL command or PL/SQL block which is stored in the SQL buffer.

\( \text{TIMING} \ [\text{START} \ \text{text} \ | \ \text{SHOW} \ | \ \text{STOP}] \)

Records timing data for an elapsed period of time, lists the current timer’s name and timing data, or lists the number of active timers.
Use the following command to access the help system.

HELP | ? [topic]

Accesses the command-line help system. Enter HELP INDEX or ? INDEX for a list of topics. You can view the Oracle Database Library at http://www.oracle.com/technology/documentation.

Use the following command to execute operating system commands.

HOST [command]

Executes an operating system command without leaving SQL*Plus. Enter HOST without command to display an operating system prompt. You can then enter multiple operating system commands.

With some operating systems, you can use another character instead of HOST such as "!" (UNIX) and "$" (Windows). See the Oracle installation and user's manuals provided for your operating system for details.

Manipulating SQL, SQL*Plus and PL/SQL Commands

Use the following commands to edit SQL commands and PL/SQL blocks.

APPEND text

Adds specified text to the end of the current line in the SQL buffer. To separate text from the preceding characters with a space, enter two spaces. To append text that ends with a semicolon, end the command with two semicolons (a single semicolon is interpreted as a command terminator).

CHANGE sepchar old [sepchar new [sepchar]]

Changes first occurrence of old on the current line of the SQL buffer. You can use any non-alphanumeric character such as "/" or "!" as a sepchar. You can omit the space between CHANGE and the first sepchar.

DEL [n | n m | n LAST | * | * n | * LAST | LAST]

Deletes one or more lines of the SQL buffer ("*" indicates the current line). You can omit the space between DEL and n or *, but not between DEL and LAST. Enter DEL with no clauses to delete the current line of the buffer.

INPUT [text]

Adds one or more new lines of text after the current line in the SQL buffer.

LIST [n | n m | n LAST | * | * n | * LAST | LAST]
Lists one or more lines of the most recently executed SQL command or PL/SQL block which is stored in the SQL buffer. Asterisk (*) indicates the current line. You can omit the space between LIST and $n$ or *, but not between LIST and LAST. Enter LIST with no clauses to list all lines.

Use the following commands to run scripts.

@ { url | file_name[.ext] } [arg ...]

Runs the SQL*Plus statements in the specified script. The script can be called from the local file system or a web server. You can pass values to script variables in the usual way.

@@ { url | file_name[.ext] } [arg ...]

Runs the SQL*Plus statements in the specified script. This command is almost identical to the @ command. It is useful for running nested scripts because it has the additional functionality of looking for the specified script in the same path or url as the calling script.

START { url | file_name[.ext] } [arg ...]

Runs the SQL*Plus statements in the specified script. The script can be called from the local file system or a web server. You can pass values to script variables in the usual way.

Use the following commands to create and modify scripts.

EDIT [file_name[.ext]]

Invokes an operating system text editor on the contents of the specified file or on the contents of the SQL buffer. To edit the buffer contents, omit the file name.

GET file_name[.ext] [LIST | NOLIST]

Loads a SQL statement or PL/SQL block from a file into the SQL buffer.

REM[ARK]

Begins a comment in a script. The REMARK command must appear at the beginning of a line, and the comment ends at the end of the line (a line cannot contain both a comment and a command). SQL*Plus does not interpret the comment as a command.

SAVE file_name[.ext] [CREATE | REPLACE | APPEND]

Saves the contents of the SQL buffer in a file.

STORE {SET} file_name[.ext] [CREATE | REPLACE | APPEND]

Saves attributes of the current SQL*Plus environment in a file.

WHENEVER OSERROR {EXIT [SUCCESS | FAILURE | n | variable | :BindVariable] [COMMIT | ROLLBACK]} | CONTINUE [COMMIT | ROLLBACK | NONE]}

Performs the specified action (exits SQL*Plus by default) if an operating system error occurs (such as a file writing error).
WHENEVER SQLERROR {EXIT [SUCCESS | FAILURE | WARNING | n | variable |
  :BindVariable] [COMMIT | ROLLBACK] | CONTINUE [COMMIT | ROLLBACK | NONE]}

Performs the specified action (exits SQL*Plus by default) if a SQL command or
PL/SQL block generates an error.

Use the following commands to write interactive
commands.

ACC[CEPT] variable [NUM[BER] | CHAR | DATE | BINARY_FLOAT |
BINARY_DOUBLE] [FOR[MAT] format] [DEF[AULT] default] [PROMPT text | 
NOPR[OMPT]] [HIDE]

Reads a line of input and stores it in a given substitution variable.

DEF[INE] [variable] | [variable = text]

Specifies a substitution variable and assigns a CHAR value to it, or lists the value and
variable type of a single variable or all variables.

PAU[SE] [text]

Displays the specified text then waits for the user to press RETURN.

PRO[MPT] [text]

Sends the specified message or a blank line to the user's screen.

UNDEF[INE] variable ...

Deletes one or more substitution variables that you defined either explicitly (with the
DEFINE command) or implicitly (with a START command argument).

Use the following commands to create and display bind
variables.

PRINT [variable ...]

Displays the current values of bind variables, or lists all bind variables.

VAR[IBLE] [variable [type][=value]]

Declares a bind variable that can be referenced in PL/SQL, or lists the current display
characteristics for a single variable or all variables.

type represents one of the following:

  NUMBER
  CHAR
  CHAR (n [CHAR | BYTE])
  NCHAR
  NCHAR (n)
VARCHAR2 (n [CHAR | BYTE])
NVARCHAR2 (n)
CLOB
NCLOB
REFCURSOR
BINARY_FLOAT
BINARY_DOUBLE

_value_ allows you to assign a value to a variable for input binding.

Use the following symbols to create substitution variables and parameters for use in scripts.

\&n

Specifies a parameter in a script you run using the START command. START substitutes values you list after the script name as follows: the first for \&1, the second for \&2, and so on.

\&user_variable, \&\&user_variable

Indicates a substitution variable in a SQL or SQL*Plus command. SQL*Plus substitutes the value of the specified substitution variable for each substitution variable it encounters. If the substitution variable is undefined, SQL*Plus prompts you for a value each time an "\&" variable is found, and the first time an "\&\&" variable is found.

. (period)
Terminates a substitution variable followed by a character that would otherwise be part of the variable name.

Formatting Query Results

Use the following commands to format, store and print your query results.

ATTRIBUTE [type_name.attribute_name [option...]]

Specifies display characteristics for a given attribute of an Object Type column, such as the format of NUMBER data. Columns and attributes should not have the same names as they share a common namespace. Lists the current display characteristics for a single attribute or for all attributes.

Where _option_ represents one of the following clauses:

\ ALI[AS] alias
\ CLE[AR]
\ FOR[MAT] format
\ LIKE {type_name.attribute_name|alias}
\ ON|OFF
BREAK [ON report_element [action [action]]] ...  
Specifies where changes occur in a report and the formatting action to perform, such as:

- suppressing the display of duplicate values for a given column
- skipping a line each time a given column value changes
- printing computed figures each time a given column value changes or at the end of the report

Enter BREAK with no clauses to list the current BREAK definition.

Where report_element has the following syntax:

{column | expr | ROW | REPORT}

and where action has the following syntax:

[SKIP n | [SKIP] PAGE] [NODUPLICATES] | DUPLICATES]

BTITLE [printspec [text | variable] ...] | [ON | OFF]

Places and formats a title at the bottom of each report page, or lists the current BTITLE definition. Use one of the following clauses in place of printspec:

- BOLD
- CENTER
- COL n
- FORMAT text
- LEFT
- RIGHT
- SKIP n
- TAB n
- CLEAR option ...

Resets or erases the current value or setting for the specified option.

Where option represents one of the following clauses:

- BRE[AKS]
- BUFF[ER]
- COL[UMNS]
- COMP[UTES]
- SCR[EEN]
- SQL
- TIM[ING]
- COL[UMN] [[column | expr] [option ...]]

Specifies display attributes for a given column, such as:

- text for the column heading
- alignment for the column heading
- format for NUMBER data
• wrapping of column data

Also lists the current display attributes for a single column or all columns.

Where option represents one of the following clauses:

```
ALI[AS] alias
CLE[AR]
ENTMAP {ON | OFF}
FOLD_A[FTER]
FOLD_B[EFORE]
FOR[MAT] format
HEA[DING] text
JUS[TIFY] {L[EFT] | C[ENTER] | R[IGHT]}
LIKE {expr | alias}
NEWL[INE]
NEW_V[ALUE] variable
NOPRI[NT] | PRI[NT]
NUL[L] text
OLD_V[ALUE] variable ON | OFF
WRA[PPED] | WOR[D_WRAPPED] | TRU[NCATED]
```

Enter COLUMN [{column | expr} FORMAT format] where the format element specifies the display format for the column.

To change the display format of a NUMBER column, use FORMAT followed by one of the elements in the following table:

<table>
<thead>
<tr>
<th>Element</th>
<th>Examples</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>. (comma)</td>
<td>9,999</td>
<td>Displays a comma in the specified position.</td>
</tr>
<tr>
<td>. (period)</td>
<td>99.99</td>
<td>Displays a period (decimal point) to separate the integral and fractional parts of a number.</td>
</tr>
<tr>
<td>$</td>
<td>$9999</td>
<td>Displays a leading dollar sign.</td>
</tr>
<tr>
<td>0</td>
<td>0999</td>
<td>Displays leading zeros</td>
</tr>
<tr>
<td></td>
<td>9990</td>
<td>Displays trailing zeros</td>
</tr>
<tr>
<td>9</td>
<td>9999</td>
<td>Displays a value with the number of digits specified by the number of 9s. Value has a leading space if positive, a leading minus sign if negative. Blanks are displayed for leading zeroes. A zero (0) is displayed for a value of zero.</td>
</tr>
<tr>
<td>B</td>
<td>B9999</td>
<td>Displays blanks for the integer part of a fixed-point number when the integer part is zero, regardless of zeros in the format model.</td>
</tr>
<tr>
<td>C</td>
<td>C999</td>
<td>Displays the ISO currency symbol in the specified position.</td>
</tr>
<tr>
<td>D</td>
<td>D9D99</td>
<td>Displays the decimal character to separate the integral and fractional parts of a number.</td>
</tr>
<tr>
<td>Element</td>
<td>Examples</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>----------</td>
<td>-------------</td>
</tr>
<tr>
<td>EEEE</td>
<td>9.999EEE</td>
<td>Displays value in scientific notation (format must contain exactly four “E”s).</td>
</tr>
<tr>
<td>G</td>
<td>9G999</td>
<td>Displays the group separator in the specified positions in the integral part of a number.</td>
</tr>
<tr>
<td>L</td>
<td>L999</td>
<td>Displays the local currency symbol in the specified position.</td>
</tr>
<tr>
<td>MI</td>
<td>9999MI</td>
<td>Displays a trailing minus sign after a negative value. Display a trailing space after a positive value.</td>
</tr>
<tr>
<td>PR</td>
<td>9999PR</td>
<td>Displays a negative value in &lt;angle brackets&gt;. Displays a positive value with a leading and trailing space.</td>
</tr>
<tr>
<td>RN m</td>
<td>RN rn</td>
<td>Displays uppercase Roman numerals. Displays lowercase Roman numerals. Value can be an integer between 1 and 3999.</td>
</tr>
<tr>
<td>S</td>
<td>S9999</td>
<td>Displays a leading minus or plus sign. Displays a trailing minus or plus sign.</td>
</tr>
</tbody>
</table>

| TM      | TM       | Displays the smallest number of decimal characters possible. The default is TM9. Fixed notation is used for output up to 64 characters, scientific notation for more than 64 characters. Cannot precede TM with any other element. TM can only be followed by a single 9 or E |
| U      | U9999    | Displays the dual currency symbol in the specified position. |

Comp[ute] [function [LABEL|text]] ... OF {expr | column | alias} ...ON {expr | column | alias | REPORT | ROW} ...]

In combination with the BREAK command, calculates and prints summary lines using various standard computations. It also lists all COMPUTE definitions. The following table lists valid functions. All functions except NUMBER apply to non-null values only. COMPUTE functions are always executed in the following sequence AVG, COUNT, MINIMUM, MAXIMUM, NUMBER, SUM, STD, VARIANCE.

<table>
<thead>
<tr>
<th>Function</th>
<th>Computes</th>
<th>Applies to Datatypes</th>
</tr>
</thead>
<tbody>
<tr>
<td>AVG</td>
<td>Average of non-null values</td>
<td>NUMBER</td>
</tr>
<tr>
<td>COUNT</td>
<td>Count of non-null values</td>
<td>All types</td>
</tr>
<tr>
<td>MINIMUM</td>
<td>Minimum value</td>
<td>NUMBER, CHAR, NCHAR, VARCHAR2 (VARCHAR), NVARCHAR2 (NCHAR VARYING)</td>
</tr>
<tr>
<td>MAXIMUM</td>
<td>Maximum value</td>
<td>NUMBER, CHAR, NCHAR, VARCHAR2 (VARCHAR), NVARCHAR2 (NCHAR VARYING)</td>
</tr>
<tr>
<td>NUMBER</td>
<td>Count of rows</td>
<td>All types</td>
</tr>
<tr>
<td>SUM</td>
<td>Sum of non-null values</td>
<td>NUMBER</td>
</tr>
<tr>
<td>Function</td>
<td>Computes</td>
<td>Applies to Datatypes</td>
</tr>
<tr>
<td>----------</td>
<td>----------</td>
<td>---------------------</td>
</tr>
<tr>
<td>STD</td>
<td>Standard deviation of non-null values</td>
<td>NUMBER</td>
</tr>
<tr>
<td>VAR[IANCE]</td>
<td>Variance of non-null values</td>
<td>NUMBER</td>
</tr>
</tbody>
</table>

REPF[OOTER] [PAGE] [printspec [text | variable] ..] | [ON | OFF]
Places and formats a footer at the bottom of a report, or lists the current REPFOOTER definition.
Where *printspec* represents one or more of the following clauses:
- BOLD
- CE[NTER]
- COL n
- FORMAT text
- LE[FT]
- R[IGHT]
- S[KIP] [n]
- TAB n

REPF[EADER] [PAGE] [printspec [text | variable] ..] | [ON | OFF]
Places and formats a header at the top of a report, or lists the current REPHEADER definition.
Where *printspec* represents one or more of the clauses shown for REPFOOTER.

SPO[OL] [filename[.ext] [CRE[ATE] | REP[PLACE] | APP[END]] | OFF | OUT]
Stores query results in a file, or optionally sends the file to a printer. OFF stops spooling, OUT stops spooling and sends the file to your computer's default printer. Enter SPOOL with no clauses to list the current spooling status. If no file extension is given, the default extension, .lst or .lis, is used.

TTI[TLE] [printspec [text | variable] ..] | [ON | OFF]
Places and formats a specified title at the top of each report page, or lists the current TTITLE definition. The old form of TTITLE is used if only a single word or a string in quotes follows the TTITLE command.
Where *printspec* represents one or more of the following clauses:
- BOLD
- CE[NTER]
- COL n
- FORMAT text
- LE[FT]
- R[IGHT]
- S[KIP] [n]
- TAB n
Accessing Databases

Use the following commands to access and copy data between tables on different databases.

```
CONN[ECT] [{logon | / | proxy} [AS {SYSASM | SYSBACKUP | SYSDBA | SYSDG | SYSOPER | SYSRAC | SYSKM} [edition=value] ]
```

where `logon` requires the following syntax:

```
username[password] [@connect_identifier]
```

where `proxy` requires the following syntax:

```
proxyuser[username] [password] [@connect_identifier]
```

Note:
The brackets around `username` in `proxy` are required syntax.

Connects a given username to the Oracle Database. If you omit `connect_identifier`, SQL*Plus connects you to the default database. If you omit `username` and/or `password`, SQL*Plus prompts you for them. CONNECT followed by a slash (/) connects you using a default (OPS$) logon.

When you run a CONNECT command, the site profile, glogin.sql, and the user profile, login.sql, are processed in that order. CONNECT does not reprompt for `username` or `password` if the initial connection does not succeed.

```
DISC[ONNECT]
```

Commits pending changes to the database and logs the current user out of Oracle, but does not exit SQL*Plus. In SQL*Plus command line, use EXIT or QUIT to log out of Oracle and return control to your computer's operating system.

```
COPY {FROM database | TO database | FROM database TO database} {APPEND | CREATE | INSERT | REPLACE} destination_table[(column, column, column, ...)] USING query
```

where `database` has the following syntax:

```
username[password]@connect_identifier
```

Copies data from a query to a table in the same or another database. APPEND, CREATE, INSERT or REPLACE specifies how COPY treats the existing copy of the destination table (if it exists). USING `query` identifies the source table and determines which rows and columns COPY copies from it. COPY supports CHAR, DATE, LONG, NUMBER and VARCHAR2 datatypes.

```
PASSW[ORD] [username]
```
Allows you to change a password without displaying it on an input device.

**XQUERY** *xquery_statement*

Allows you to run an XQuery from SQL*Plus.

**Miscellaneous**

**ARCHIVE LOG LIST**
Displays information about redo log files.

```
DESC[RISE] { [schema.]object[@connect_identifier] }
```

Lists the column definitions for a table, view or synonym, or the specifications for a function or procedure.

```
RECOVER {general | managed | BEGIN BACKUP | END BACKUP}
```

where the `general` clause has the following syntax:

```
[AUTOMATIC] [FROM location] { {full_database_recovery | partial_database_recovery
 | LOGFILE filename} { [TEST | ALLOW integer CORRUPTION | parallel_clause ]
 [TEST [ALLOW integer CORRUPTION | parallel_clause ]...][CONTINUE [DEFAULT] | CANCEL}
```

where the `full_database_recovery` clause has the following syntax:

```
[STANDBY] DATABASE { {UNTIL {CANCEL | TIME date | CHANGE integer} | USING
 BACKUP CONTROLFILE} {UNTIL {CANCEL | TIME date | CHANGE integer} | USING
 BACKUP CONTROLFILE | SNAPSHOT TIME date}...}
```

where the `partial_database_recovery` clause has the following syntax:

```
{TABLESPACE tablespace [, tablespace]... | DATAFILE {filename | filenumber} [, filename | filenumber]... | STANDBY {TABLESPACE tablespace [, tablespace]... |
 DATAFILE {filename | filenumber} [, filename | filenumber]...} UNTIL [CONSISTENT
 WITH] CONTROLFILE }
```

where the `parallel` clause has the following syntax:

```
{ NOPARALLEL | PARALLEL [ integer ] }
```

where the `managed` clause has the following syntax:

```
MANAGED STANDBY DATABASE recover_clause | cancel_clause | finish_clause
```

where the `recover_clause` has the following syntax:

```
{ { DISCONNECT [ FROM SESSION ] } | { TIMEOUT integer | NOTIMEOUT } }
{ NODELAY | DEFAULT DELAY | DELAY integer } | NEXT integer | { EXPIRE integer |
 NO EXPIRE } | parallel_clause | USING CURRENT LOGFILE | UNTIL CHANGE
 integer | THROUGH { [ THREAD integer | SEQUENCE integer | ALL ARCHIVELOG |
 { ALL | LAST | NEXT } SWITTOVER ] | { DISCONNECT [ FROM SESSION ] } |
 { TIMEOUT integer | NOTIMEOUT } | { NODELAY | DEFAULT DELAY | DELAY
 integer } | NEXT integer | { EXPIRE integer | NO EXPIRE } | parallel_clause | USING
```
CURRENT LOGFILE | UNTIL CHANGE integer | THROUGH { [ THREAD integer ]
SEQUENCE integer | ALL ARCHIVELOG | { ALL | LAST | NEXT } SWITchoVER} ]...

where the cancel_clause has the following syntax:

CANCEL [IMMEDIATE] [WAIT | NOWAIT]

where the finish_clause has the following syntax:

[ DISCONNECT [ FROM SESSION ] ] [ parallel_clause ] FINISH [ SKIP [ STANDBY
LOGFILE ] ] [ WAIT | NOWAIT ]

where the parallel_clause has the following syntax:

{ NOPARALLEL | PARALLEL [ integer ] }

Performs media recovery on one or more tablespaces, one or more datafiles, or the
entire database.

SET system_variable value

Sets a system variable to alter the SQL*Plus environment settings for your current
session. For example, to:

• set the display width for data
• customize HTML formatting
• enable or disable printing of column headings
• set the number of lines per page

Enter a system variable followed by a value as shown below. The default value for a
system variable is underlined.

SET APPI[NFO]{ON | OFF | text}
SET ARRAY[SIZE] {15 | n}
SET AUTO[COMMIT] {ON | OFF | IMM[EDIATE] | n}
SET AUTOP[RINT] {ON | OFF}
SET AUTORECOVERY {ON | OFF}
SET AUTO[T]RACE {ON | OFF | TRACE[ONLY]} [EXP[LAIN]] [STATISTICS]
SET BLO[CKTERMINATOR] {., | c | ON | OFF}
SET CMDS[EP] {.; | c | ON | OFF}
SET COLINVI[SIBLE] {ON | OFF}
SET COLSEP {., | text}
SET CON[CAT] {., | c | ON | OFF}
SET COPYC[OMMIT] {0 | n}
SET COPYTYPECHECK {ON | OFF}
SET DEFINE {& | c | ON | OFF}
SET DESCRIBE [DEPTH {1 | n | ALL}] [LINENUM {ON | OFF}] [INDENT {ON | OFF}]
SET ECHO {ON | OFF}
SET EDITF[ILE] file_name[.ext]
SET EMB[EDDED] {ON | OFF}
SET ERRORLOGGING {ON | OFF} [TABLE [schema.]tablename] [TRUNCATE]
[IDENTIFIER identifier]
SET ESCAPE {1 | c | ON | OFF}
SET ESCCHAR {@ | ? | % | $ | OFF}
SET EXIT_COMMIT {ON | OFF}
SET FEEDBACK {OFF | ENTRY | INTERMEDIATE | FULL}
SET FLUSH {ON | OFF}
SET HEADING {ON | OFF}
SET HEADERS {1 | c | ON | OFF}
SET HISTORY {ON | OFF | n}
SET INSTANCE [instance_path | LOCAL]
SET LINCSIZE {80 | n | WINDOW}
SET LOB_OFFSET {n | OFF | 1}
SET LOB_PREFETCH {0 | n}
SET LOGSOURCE [pathname]
SET LONG {80 | n}
SET LONGCHUNKSIZE {80 | n}
SET MARK CSV {ON | OFF} [DELIMITER character] [QUOTE {ON | OFF}]
SET MARKUP HTML {ON | OFF} [HEAD text] [BODY text] [TABLE text] [ENTMAP {ON | OFF}] [SPOOL {ON | OFF}] [PRE_FORMAT {ON | OFF}]
SET NEWPAGE {1 | n | NONE}
SET NULL text
SET NUMFORMAT format
SET NUMWIDTH {10 | n}
SET PAGESIZE {14 | n}
SET PAUSE {ON | OFF | text}
SET RECSEP {WRAPPED | EACH | OFF}
SET RECSEPCHAR {1 | c}
SET ROWLIMIT {n | OFF}
SET ROW_PREFETCH {1 | OFF}
SET SECUREDCOLUMN {ON | OFF} [UNAUTHORIZED text][UNKNOWN text]
SET SERVER_OUTPUT_PUT {ON | OFF} [SIZE {n | UNLIMITED}] [FORMAT {WRAPPED | WORD_WRAPPED | TRUNCATED}]
SET SHIFT_INOUT {VISIBLE | INVISIBLE}
SET SHOWMODE {ON | OFF}
SET SQLBLANKLINES {ON | OFF}
SET SQLCASE {MIXED | LOWER | UPPER}
SET SQLCONTINUE {> | text}
SET SQL_COMMENT {ON | OFF}
SET SQL_NOUNIQUE {y | text}
SET SQL_NUMBER {ON | OFF}
SET SQLPLUS_COMPATIBILITY {x.y.z}
SET SQL_PREFERRED {x | c}
SET SQLPrompt {SQL> | text}
SET SQLTERMINATOR {; | c | ON | OFF}
SET STATEMENT_CACHE {0 | n}
SET SUFFIX {SQL | text}
SET TAB {OFF | OFF}
SHOW [option]

Shows the value of a SQL*Plus system variable, or the current SQL*Plus environment. Enter any system variable set by the SET command in place of system_variable.
SHOW SGA requires a DBA privileged login. Use one of the following terms or clauses in place of option:

system_variable
ALL
CON_ID
CON_NAME
BTI[TLE]
EDITION
ERR[ORS] [ {ANALYTIC VIEW | ATTRIBUTE DIMENSION | HIERARCHY | FUNCTION | PROCEDURE | PACKAGE | PACKAGE BODY | TRIGGER | VIEW | TYPE | TYPE BODY | DIMENSION | JAVA CLASS} [schema.]name]
HIST[ORY]
LNO
LOBPREF[ETCH]
PARAMETER[S] [parameter_name]
PDBS
PNO
RECYC[LEBIN] [original_name]
REL[EASE]
REPF[OOTER]
REPH[EADER]
ROWPREF[ETCH]
SGA
SPOO[L]
SPPARAMETER[S] [parameter_name]
SQLCODE
STATEMENTC[ACHE]
TTI[TLE]
USER
XQUERY