CERTIFICATION OBJECTIVE 1.04

Describe the Purpose of DDL

Data Definition Language consists of those SQL statements that are used to build database objects. Specifically, DDL statements are used to

- Create, alter, and drop tables and other database objects
- Add comments on a particular object to be stored in the database and associated with that object
- Issue privileges to users to perform various tasks in the database
- Initiate performance analysis on objects using built-in tools

The following list briefly describes DDL statements that are tested by the exam:

- **CREATE** Used to create a user, table, view, index, synonym, or other object in the database.
- **ALTER** Used on an existing object in the database to modify that object's structure, name, or some other attribute. (Two exceptions are the uses of ALTER with the reserved words SESSION and SYSTEM. ALTER SESSION and ALTER SYSTEM are not technically considered DDL statements but fall under a different category. Neither is included on this exam.)
- **DROP** Used to remove a database object from the database that has already been created with the CREATE statement.
- **RENAME** Changes the name of an existing database object.
- **TRUNCATE** Removes all the rows—in other words, data—from an existing table in the database. TRUNCATE is

something of a brute-force alternative to the DELETE statement, in that TRUNCATE gives up recovery options offered by DELETE in exchange for faster performance. These differences in approach are the reason TRUNCATE is categorized as DDL while DELETE is DML.

- **GRANT** Provides *privileges*, or rights, to user objects to enable them to perform various tasks in the database.
- **REVOKE** Removes privileges that have been issued with the GRANT statement.
- **FLASHBACK** Restores an earlier version of a table or database.
- **PURGE** Irrevocably removes database objects from the recycle bin.
- **COMMENT** Adds comments to the data dictionary for an existing database object.

Each DDL statement is rich with options and clauses. I will review and provide examples of most of these statements as you progress through the book.

SQL keywords, such as CREATE, are not really "commands" or "statements" by themselves but become commands when combined with other reserved words, as in CREATE TABLE or CREATE SEQUENCE, which are commands or statements. In practice, CREATE can be called a statement or command by professionals in the field, and even by Oracle Corporation in various forms of documentation. Technically there is a difference. However, this isn't an issue on the exam. Similarly, the terms command and statement tend to be used interchangeably by Oracle's documentation. If you conduct some searches in the SQL Language Reference Manual, you'll

find plenty of examples of SQL statements being referred to as commands. Either is fine. And none of these issues is a concern on the exam.

CERTIFICATION OBJECTIVE 1.05

Describe the Purpose of DML

DML refers to those statements in SQL that are used to work with data in the objects. DML statements are used to add, modify, view, and delete data in a database object, such as a table.

The following list briefly describes each DML statement that is tested by the exam:

- **SELECT** Displays data of a database table or view
- **INSERT** Adds data to a database table, either directly or, in some situations, through a view
- **UPDATE** Modifies existing data in a table, either directly or, in some situations, through a view
- **DELETE** Removes existing data from a table, either directly or, in some situations, through a view
- **MERGE** Performs a combination of INSERT, UPDATE, and DELETE statements in a single statement

The SELECT statement is rather involved and will get several chapters' worth of review. The other DML statements are reviewed in this chapter and in various sections that follow.

Transaction Control Language

In addition to these DML statements, three more SQL statements are important for working with DML. These statements are not part of DML but instead are categorized as TCL. These statements are specifically identified by Oracle within the certification objectives for DML, so I'll discuss them in this chapter. These are the statements you need to study:

- **COMMIT** Saves a set of DML modifications performed in the current database session
- **ROLLBACK** Undoes a set of DML modifications performed during the current database session
- **SAVEPOINT** Marks a position in a session to prepare for a future ROLLBACK to enable that ROLLBACK to restore data at a selected point in a session other than the most recent commit event

In addition to the explicit COMMIT command, any DDL statement will cause an implied commit action to occur. In other words, if you issue an UPDATE statement on table A and then issue a GRANT statement to user B for accessing table C, an implied commit event will occur, and the data you updated to table A will be committed, even though you haven't issued an explicit COMMIT.