Database Programming with SQL

14-3: Managing Constraints

Practice Activities

Objectives
- List four different functions that the ALTER statement can perform on constraints
- Write ALTER TABLE statements to add, drop, disable, and enable constraints
- Name a business function that would require a DBA to drop, enable, and/or disable a constraint or use the CASCADE syntax
- Query the data dictionary for USER_CONSTRAINTS and interpret the information returned

Vocabulary
Identify the vocabulary word for each definition below.

	To deactivate an integrity constraint
	Disables dependent integrity constraints
	To add, modify, or drop columns from a table
	To activate an integrity constraint currently disabled
	Removes a constraint from a table
	Allows user to delete a column from a table
	Defines the actions the database server takes when a user attempts to delete or update a key to which existing foreign keys point

Try It / Solve It

Using Oracle Application Express, click the SQL Workshop tab in the menu bar. Click the Object Browser and verify that you have a table named copy_d_clients and a table named copy_d_events. If you don't have these tables in your schema, create them before completing the exercises below. Here is how the original tables are related. The d_clients table has a primary key client_number. This has a primary-key constraint and it is referenced in the foreign-key constraint on the d_events table.

1. What are four functions that an ALTER statement can perform on constraints?

2. Since the tables are copies of the original tables, the integrity rules are not passed onto the new tables; only the column datatype definitions remain. You will need to add a PRIMARY KEY constraint to the copy_d_clients table. Name the primary key copy_d_clients_pk . What is the syntax you used to create the PRIMARY KEY constraint to the copy_d_clients.table?

3. Create a FOREIGN KEY constraint in the copy_d_events table. Name the foreign key copy_d_events_fk. This key references the copy_d_clients table client_number column. What is the syntax you used to create the FOREIGN KEY constraint in the copy_d_events table?

4. Use a SELECT statement to verify the constraint names for each of the tables. Note that the tablenames must be capitalized.

   a. The constraint name for the primary key in the copy_d_clients table is _____.

   b. The constraint name for the foreign key in the copy_d_events table is _____.

5. Drop the PRIMARY KEY constraint on the copy_d_clients table. Explain your results.

6. Add the following event to the copy_d_events table. Explain your results.

ID	NAME	EVENT_DATE	DESCRIPTION	COST	VENUE_ID	PACK-AGE_CODE	THEME_CODE	CLI-ENT_NUMBER
140	Cline Bas Mitzvah	15-Jul-2004	Church and Private Home formal	4500	105	87	77	7125

7. Create an ALTER TABLE query to disable the primary key in the copy_d_clients table. Then add the values from #5 to the copy_d_events table. Explain your results.

8. Repeat question 5: Insert the new values in the copy_d_events table. Explain your results.

9. Enable the primary-key constraint in the copy_d_clients table. Explain your results.

10. If you wanted to enable the foreign-key column and reestablish the referential integrity between these two tables, what must be done?

11. Why might you want to disable and then re-enable a constraint?

12. Query the data dictionary for some of the constraints that you have created. How does the data dictionary identify each constraint type?