Database Programming with SQL
16-1: Working with Sequences
Practice Activities

Objectives

- List at least three useful characteristics of a sequence
- Write and execute a SQL statement that creates a sequence
- Query the data dictionary using USER_SEQUENCES to confirm a sequence definition
- Apply the rules for using NEXTVAL to generate sequential numbers for use in a table
- List the advantages of caching sequence values
- Name three reasons why gaps can occur in a sequence

Vocabulary

Identify the vocabulary word for each definition below.

	Command that automatically generates sequential numbers
	Generates a numeric value
	Returns the next available sequence value
	Specifies the interval between sequence numbers
	Specifies a maximum value of 10^27 for an ascending sequence and -1 for a descending sequence (default)
	returns the current sequence value
	specifies the minimum sequence value
	specifies whether the sequence continues to generate values after reaching its maximum or minimum values
	specifies a minimum value of 1 for an ascending sequence and –(10^26) for a descending sequence (default)
	specifies a maximum or default value the sequence can generate
	specifies the first sequence number to be generated
	specifies how many values the Server pre-allocates and keeps in memory

Try It / Solve It

1. Using CREATE TABLE AS subquery syntax, create a seq_d_songs table of all the columns in the DJs on Demand database table d_songs. Use the SELECT * in the subquery to make sure that you have copied all of the columns.

2. Because you are using copies of the original tables, the only constraints that were carried over were the NOT NULL constraints. Create a sequence to be used with the primary-key column of the seq_d_songs table. To avoid assigning primary-key numbers to these tables that already exist, the sequence should start at 100 and have a maximum value of 1000. Have your sequence increment by 2 and have NOCACHE and NOCYCLE. Name the sequence seq_d_songs_seq.

3. Query the USER_SEQUENCES data dictionary to verify the seq_d_songs_seq SEQUENCE settings.

4. Insert two rows into the seq_d_songs table. Be sure to use the sequence that you created for the ID column. Add the two songs shown in the graphic.

ID	TITLE	DURATION	ARTIST	TYPE_CODE
	Island Fever	5 min	Hawaiian Islanders	12
	Castle of Dreams	4 min	The Wanderers	77

5. Write out the syntax for seq_d_songs_seq to view the current value for the sequence. Use the DUAL table. (Oracle Application Developer will not run this query.)

6. What are three benefits of using SEQUENCEs?

7. What are the advantages of caching sequence values?

8. Name three reasons why gaps may occur in a sequence?

Extension Exercise

1. Create a table called “students”. You can decide which columns belong in that table and what datatypes these columns require. (The students may create a table with different columns; however, the important piece that must be there is the student_id column with a numeric datatype. This column length must allow the sequence to fit, e.g. a column length of 4 with a sequence that starts with 1 and goes to 10000000 will not work after student #9999 is entered.)

2. Create a sequence called student_id_seq so that you can assign unique student_id numbers for all students that you add to your table.

3. Now write the code to add students to your STUDENTS table, using your sequence “database object.”