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.

CREATE SEQUENCE	Command that automatically generates sequential numbers
SEQUENCE	Generates a numeric value
NEXTVAL	Returns the next available sequence value
INCREMENT BY	Specifies the interval between sequence numbers
NOMAXVALUE	Specifies a maximum value of 10^27 for an ascending sequence and -1 for a descending sequence (default)
CURRVAL	returns the current sequence value
MINVALUE	specifies the minimum sequence value
CYCLE / NOCYCLE	specifies whether the sequence continues to generate values after reaching its maximum or minimum values
NOMINVALUE	specifies a minimum value of 1 for an ascending sequence and –(10^26) for a descending sequence (default)
MAXVALUE	specifies a maximum or default value the sequence can generate
START WITH	specifies the first sequence number to be generated
CACHE	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.

```
CREATE TABLE seq_d_songs AS
SELECT * FROM d_songs;
```

ID	TITLE	DURATION	ARTIST	TYPE_CODE
45	Its Finally Over	5 min	The Hobbits	12
46	Im Going to Miss My Teacher	2 min	Jane Pop	12
47	Hurrah for Today	3 min	The Jubilant Trio	77
48	Meet Me At the Altar	6 min	Bobby West	1
49	Lets Celebrate	8 min	The Celebrants	77
50	All These Years	10 min	Diana Crooner	88

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.

```
CREATE SEQUENCE seq_d_songs_seq
INCREMENT BY 2
START WITH 100
MAXVALUE 1000
NOCACHE
NOCYCLE;
```

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

```
SELECT sequence_name, min_value, max_value,
increment_by, last_number
FROM user_sequences
WHERE sequence_name = 'SEQ_D_SONGS_SEQ';
```

SEQUENCE_NAME	MIN_VALUE	MAX_VALUE	INCREMENT_BY	LAST_NUMBER
SEQ_D_SONGS_SEQ	1	1000	2	100

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.

```
INSERT INTO seq_d_songs (id, title, duration, artist, type_code)
VALUES (seq_d_songs_seq.NEXTVAL, 'Island Fever', '5 min', 'Hawaiian Islanders', 12);
```

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

```
INSERT INTO seq_d_songs (id, title, duration, artist, type_code)
VALUES (seq_d_songs_seq.NEXTVAL, '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.)

```
SELECT seq_d_songs_seq.CURRVAL
FROM dual;
```

6. What are three benefits of using SEQUENCEs?

Ensures unique numeric values for primary keys.
Simplifies manual key generation.
Reduces human errors by automating ID management.

7. What are the advantages of caching sequence values?

Improves performance by pre-allocating values.
Reduces access to disk for sequence numbers.
Useful for high-volume transactions.

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

Rollback of a transaction using NEXTVAL.
System crashes while caching sequence values.
Using the same sequence across multiple tables.

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.)

```
CREATE TABLE students_kb (
    student_id NUMBER(6) PRIMARY KEY,
    first_name VARCHAR2(30),
    last_name VARCHAR2(30),
    enrollment_date DATE
);
```

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.

```
CREATE SEQUENCE student_id_seq
START WITH 1
INCREMENT BY 1
NOCACHE
NOCYCLE;
```

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

```
INSERT INTO students_kb (student_id, first_name, last_name, enrollment_date)
VALUES (student_id_seq.NEXTVAL, 'John', 'Doe', SYSDATE);

INSERT INTO students_kb (student_id, first_name, last_name, enrollment_date)
VALUES (student_id_seq.NEXTVAL, 'Jane', 'Smith', SYSDATE);
```

STUDENT_ID	FIRST_NAME	LAST_NAME	ENROLLMENT_DATE
1	John	Doe	12/17/2024
2	Jane	Smith	12/17/2024

Database Programming with SQL
16-2: Indexes and Synonyms
Practice Activities

Objectives

- Define an index and its use as a schema object
- Name the conditions that cause an index to be created automatically
- Create and execute a CREATE INDEX and DROP INDEX statement
- Construct and execute a function-based index
- Construct a private and public synonym

Vocabulary

Identify the vocabulary word for each definition below.

Confirming Indexes	Confirms the existence of indexes from the USER_INDEXES data dictionary view
Index	Schema object that speeds up retrieval of rows
Synonym	To refer to a table by another name to simplify access
Composite Index	An index that you create on multiple columns in a table
Unique Index	The Oracle Server automatically creates this index when you define a column in a table to have a PRIMARY KEY or a UNIQUE KEY constraint
Function-Based Index	Stores the indexed values and uses the index based on a SELECT statement to retrieve the data
DROP INDEX	Removes an index
Synonym	Gives alternative names to objects

Try It / Solve It

1. What is an index and what is it used for?

   ```
   An index is a schema object that provides direct and fast access to rows in a table. It
   speeds up data retrieval by avoiding full table scans.
   ```

2. What is a ROWID, and how is it used?

   ```
   A ROWID is a unique base-64 string representation of a row's physical location in the database. It
   contains the block identifier, row location, and file identifier. ROWIDs are used because they
   provide the fastest way to access a row.
   ```

3. When will an index be created automatically?

   ```
   A PRIMARY KEY constraint is defined.
   A UNIQUE KEY constraint is defined
   ```

4. Create a nonunique index (foreign key) for the DJs on Demand column (cd_number) in the D_TRACK_LISTINGS table. Use the Oracle Application Developer SQL Workshop Data Browser to confirm that the index was created.

```
CREATE INDEX d_track_cd_idx
ON d_track_listings(cd_number);
```

5. Use the join statement to display the indexes and uniqueness that exist in the data dictionary for the DJs on Demand D_SONGS table.

```
SELECT ic.index_name, ic.column_name, ic.column_position, id.uniqueness
FROM user_indexes id
JOIN user_ind_columns ic
ON id.index_name = ic.index_name
WHERE id.table_name = 'D_SONGS';
```

6. Use a SELECT statement to display the index_name, table_name, and uniqueness from the data dictionary USER_INDEXES for the DJs on Demand D_EVENTS table.

```
SELECT index_name, table_name, uniqueness
FROM user_indexes
WHERE table_name = 'D_EVENTS';
```

7. Write a query to create a synonym called dj_tracks for the DJs on Demand d_track_listings table.

```
CREATE SYNONYM dj_tracks
FOR d_track_listings;
```

8. Create a function-based index for the last_name column in DJs on Demand D_PARTNERS table that makes it possible not to have to capitalize the table name for searches. Write a SELECT statement that would use this index.

```
CREATE INDEX d_partners_lastname_idx
ON d_partners(UPPER(last_name));
```

9. Create a synonym for the D_TRACK_LISTINGS table. Confirm that it has been created by querying the data dictionary.

```
CREATE SYNONYM d_track_syn
FOR d_track_listings;


SELECT * FROM user_synonyms WHERE synonym_name = 'D_TRACK_SYN';
```

10. Drop the synonym that you created in question 9.

```
DROP SYNONYM d_track_syn;
```

Database Programming with SQL
17-1: Controlling User Access
Practice Activities

Objectives

- Compare the difference between object privileges and system privileges
- Construct the two commands required to enable a user to have access to a database
- Construct and execute a GRANT… ON …TO statement to assign privileges to objects in a user's schema to other users and/or PUBLIC
- Query the data dictionary to confirm privileges granted

Try It / Solve It

1. What are system privileges concerned with?

```
System privileges are concerned with the ability to perform actions on the database
itself, such as creating users, tables, views, and sequences.
```

2. What are object privileges concerned with?

```
Object privileges are concerned with the ability to perform actions on specific database
objects, such as tables, views, sequences, or procedures.
```

3. What is another name for object security?

```
Another name for object security is data security.
```

4. What commands are necessary to allow Scott access to the database with a password of tiger?

```
CREATE USER scott IDENTIFIED BY tiger;
GRANT CREATE SESSION TO scott;
```

5. What are the commands to allow Scott to SELECT from and UPDATE the d_clients table?

```
GRANT SELECT, UPDATE ON d_clients TO scott;
```

6. What is the command to allow everybody the ability to view the d_songs table?

```
GRANT SELECT ON d_songs TO PUBLIC;
```

7. Query the data dictionary to view the object privileges granted to you the user.

```
SELECT * FROM user_tab_privs_recd;
```

8. What privilege should a user be given to create tables?

```
The CREATE TABLE privilege should be granted to the user.
```

9. If you create a table, how can you pass along privileges to other users just to view your table?

```
GRANT SELECT ON your_table_name TO other_user;
```

10. What syntax would you use to grant another user access to your copy_employees table?

```
GRANT SELECT ON copy_employees TO other_user;
```

11. How can you find out what privileges you have been granted for columns in the tables belonging to others?

```
SELECT * FROM user_col_privs_recd;
```

Database Programming with SQL
17-2: Creating and Revoking Object Privileges
Practice Activities

Objectives

- Explain what a ROLE is and what its advantages are
- Construct a statement to create a ROLE and GRANT privileges to it
- Construct a GRANT ON TO WITH GRANT OPTION statement to assign privileges to objects in a user's schema to other users and/or PUBLIC
- Construct and execute a statement to REVOKE object privileges from other users and/or from PUBLIC
- Distinguish between privileges and roles
- Explain the purposes of a database link

Try It / Solve It

1. What is a role?

```
A role is a named group of related privileges that can be granted to users.
It simplifies the management of privileges by grouping them into one entity
```

2. What are the advantages of a role to a DBA?

```
Simplifies privilege management.
Allows the DBA to grant and revoke multiple privileges at once.
Roles can be reused across users.
Makes maintaining security easier in a large database environment
```

3. Give the ability to another user in your class to look at one of your tables. Give him the right to let other students have that ability.

```
GRANT SELECT ON your_table TO user_name WITH GRANT OPTION;
```

4. You are the DBA. You are creating many users who require the same system privileges. What should you use to make your job easier?

```
Use a ROLE to group the privileges and grant the role to the users
```

5. What is the syntax to accomplish the following?
   a. Create a role of manager that has the privileges to select, insert, and update and delete from the employees table

```
CREATE ROLE manager;
GRANT SELECT, INSERT, UPDATE, DELETE ON employees TO manager;
```

   b. Create a role of clerk that just has the privileges of select and insert on the employees table

```
CREATE ROLE clerk;
GRANT SELECT, INSERT ON employees TO clerk;
```

   c. Grant the manager role to user scott

```
GRANT manager TO scott;
```

   d. Revoke the ability to delete from the employees table from the manager role

```
REVOKE DELETE ON employees
FROM manager;
```

6. What is the purpose of a database link?

```
A database link enables a connection from one Oracle database to another, allowing a user to query or manipulate data in the
remote database as if it were local. This provides seamless access to distributed data
```

Database Programming with SQL
17-3: Regular Expressions
Practice Activities

Objectives

- Describe regular expressions
- Use regular expressions to search, match, and replace strings in SQL statements
- Construct and execute regular expressions and check constraints

Try It / Solve It

1. Working with the employees table, and using regular expressions, write a query that returns employees whose first names start with a “S” (uppercase) followed by either a “t” (lowercase) or “h” (lowercase).

```
SELECT first_name, last_name
FROM employees
WHERE REGEXP_LIKE(first_name, '^S[th]');
```

2. Investigate the LOCATIONS table.

   a. Describe the table.

```
DESC LOCATIONS;
```

   b. Perform a select that returns all rows and all columns of that table.

```
SELECT *
FROM LOCATIONS;
```

   c. Write a query using regular expressions that removes the spaces in the street_address column in the LOCATIONS table.

```
SELECT street_address,
       REGEXP_REPLACE(street_address, ' ', '') AS no_spaces
FROM LOCATIONS;
```