

## EXERCISE-14

### OTHER DATABASE OBJECTS

#### Objectives

After the completion of this exercise, the students will be able to do the following:

- Create, maintain, and use sequences
- Create and maintain indexes

#### Database Objects

Many applications require the use of unique numbers as primary key values. You can either build code into the application to handle this requirement or use a sequence to generate unique numbers. If you want to improve the performance of some queries, you should consider creating an index. You can also use indexes to enforce uniqueness on a column or a collection of columns. You can provide alternative names for objects by using synonyms.

#### **What Is a Sequence?**

A sequence:

- Automatically generates unique numbers
- Is a sharable object
- Is typically used to create a primary key value
- Replaces application code
- Speeds up the efficiency of accessing sequence values when cached in memory

#### **The CREATE SEQUENCE Statement Syntax**

Define a sequence to generate sequential numbers automatically:

```
CREATE SEQUENCE sequence
[INCREMENT BY n]
[START WITH n]
[{:MAXVALUE n | NOMAXVALUE}]
[{:MINVALUE n | NOMINVALUE}]
[{:CYCLE | NOCYCLE}]
[{:CACHE n | NOCACHE}];
```

#### **In the syntax:**

*sequence* is the name of the sequence generator

INCREMENT BY *n* specifies the interval between sequence numbers where *n* is an integer (If this clause is omitted, the sequence increments by 1.)

START WITH *n* specifies the first sequence number to be generated (If this clause is omitted, the sequence starts with 1.)

MAXVALUE *n* specifies the maximum value the sequence can generate

NOMAXVALUE specifies a maximum value of  $10^{27}$  for an ascending sequence and -1 for a descending sequence (This is the default option.)

DROP INDEX index;

Find the Solution for the following:

1. Create a sequence to be used with the primary key column of the DEPT table. The sequence should start at 200 and have a maximum value of 1000. Have your sequence increment by ten numbers. Name the sequence DEPT\_ID\_SEQ.

2. Write a query in a script to display the following information about your sequences: sequence name, maximum value, increment size, and last number

3. Write a script to insert two rows into the DEPT table. Name your script lab12\_3.sql. Be sure to use the sequence that you created for the ID column. Add two departments named Education and Administration. Confirm your additions. Run the commands in your script.

4. Create a nonunique index on the foreign key column (DEPT\_ID) in the EMP table.

5. Display the indexes and uniqueness that exist in the data dictionary for the EMP table.

- 1) Create sequence dept-id-seq, start with 200 increment by 10 maxvalue 1000;
- 2) Select sequence-name, max-value, increment-by, last-number from user\_sequences where sequence-name = "DEPT\_ID\_SEQ";
- 3) Insert into dept(dept-id, dept-name) values (DEPT\_ID-SEQ.NEXTVAL, 'Education');  
Insert into dept (dept-id, dept-name) values (DEPT\_ID-SEQ.NEXTVAL, 'Administration');  
Commit;  
Select \* from dept;
- 4) Create index emp-dept-id-idx on emp(dept-id);
- 5) Select index-name, uniqueness from user\_indexes where table-name = 'EMP';

