Date : 24/9/25

EXERCISE-14

OTHER DATABASE OBJECTS

Objectives AIM: To create and maintain sequences and indexes

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
MAX VALUE 1000
NOCYCLE;

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');
Select dept_id, dept_name FROM DEPT where dept_name
IN ('Education', 'Administration') ORDER BY dept_id; COMMIT;

4. CREATE INDEX emp_dept_id_fk_idx ON
EMP(dept_id);

5. Select index_name, table_name, uniqueness FROM USER_INDEXES where table_name = 'EMP';

RESULT:

Thus the sequences and indexes of database objects are studied.