Chapter 10
Other Database Objects

Objectives

- Describe some database objects and their uses
- Create, maintain, and use sequences
- Create and maintain indexes
- Create private and public synonyms

What Is a Sequence?

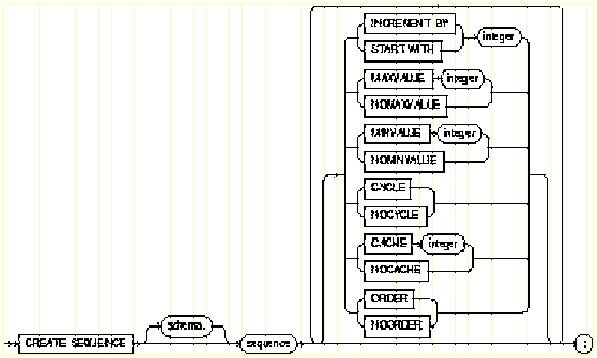
- Automatically generates unique numbers
- Is a sharable object
- Is typically used to create a primary key value
- Replaces application code
- Speeds up the efficiency of accessing sequence values when cached in memory

The CREATE SEQUENCE Statement

- 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}];

INCREMENT BY – integer interval
START WITH – first sequence number (default 1)
[MAXVALUE] NOMAXVALUE – if no max, 10 (default)
[MINVALUE] NOMINVALUE – if no min, 1 for ascending, 10 descending (default)
[CYCLE] NOCYCLE – if no cycle, does not generate additional values (default)
[CACHE] NOCACHE – by default, Oracle Server caches 20 values



Creating a Sequence

- Create a sequence named DEPT_DEPTNO to be used for the primary key of the DEPT table.
- Do not use the CYCLE option.

```
SQL> CREATE SEQUENCE dept_deptno
  2           INCREMENT BY 1
  3           START WITH 91
  4           MAXVALUE 100
  5           NOCACHE
  6           NOCYCLE;
Sequence created.
```

Confirming Sequences

- Verify your sequence values in the USER_SEQUENCES data dictionary table.

```
SQL> SELECT sequence_name, min_value, max_value,
  2           increment_by, last_number
  3  FROM   user_sequences;
```

- The LAST_NUMBER column displays the next available sequence number.

NEXTVAL and CURRVAL Pseudocolumns

- NEXTVAL returns the next available sequence value.
  It returns a unique value every time it is referenced, even for different users.
- CURRVAL obtains the current sequence value.
  NEXTVAL must be issued for that sequence before CURRVAL contains a value.

- You can use NEXTVAL and CURRVAL in the following:
  - The SELECT list of a SELECT that is not part of a subquery
  - The SELECT list of a subquery in an INSERT
  - The VALUES clause of an INSERT
  - The SET clause of an UPDATE

- You cannot use NEXTVAL and CURRVAL in the following:
  - A SELECT of a view
  - A SELECT with the DISTINCT keyword
  - A SELECT with the GROUP BY, HAVING or ORDER BY clauses
  - A subquery in a SELECT, DELETE or UPDATE
  - A DEFAULT expression in a CREATE TABLE or ALTER TABLE

Using a Sequence

– Insert a new department named “MARKETING” in San Diego.–View the current value for the DEPT_DEPTNO sequence.

```
SQL> INSERT INTO dept(deptno, dname, loc)
  2  VALUES      (dept_deptno.NEXTVAL,
  3                    'MARKETING', 'SAN DIEGO');
1 row created.
```

– View the current value for the DEPT_DEPTNO sequence.

```
SQL> SELECT       dept_deptno.CURRVAL
  2  FROM         dual;
```

Using a Sequence

– Caching sequence values in memory allows faster access to those values.
– Gaps in sequence values can occur when:
  o A rollback occurs
  o The system crashes
  o A sequence is used in another table
– View the next available sequence, if it was created with NOCACHE, by querying the USER_SEQUENCES table.

Modifying a Sequence

• Change the increment value, maximum value, minimum value, cycle option, or cache option.

```
SQL> ALTER SEQUENCE dept_deptno
  2              INCREMENT BY 1
  3              MAXVALUE 999999
  4              NOCACHE
  5              NOCYCLE;
Sequence altered.
```

Guidelines for Modifying a Sequence

– You must be the owner or have the ALTER privilege for the sequence.
– Only future sequence numbers are affected.
– The sequence must be dropped and re-created to restart the sequence at a different number.
– Some validation is performed.

Removing a Sequence

– Remove a sequence from the data dictionary by using the DROP SEQUENCE statement.
– Once removed, the sequence can no longer be referenced.

What Is an Index?

- Is a schema object
- Is used by the Oracle Server to speed up the retrieval of rows by using a pointer
- Can reduce disk I/O by using rapid path access method to locate the data quickly
- Is independent of the table it indexes
- Is used and maintained automatically by the Oracle Server

How Are Indexes Created?

- Automatically: A unique index is created automatically when you define a PRIMARY KEY or UNIQUE constraint in a table definition.
- Manually: Users can create nonunique indexes on columns to speed up access time to the rows.

Creating an Index

- Create an index on one or more columns.

```
CREATE INDEX index
ON table (column[, column]...);
```

- Improve the speed of query access on the ENAME column in the EMP table.

```
SQL> CREATE INDEX      emp_ename_idx
  2  ON               emp(ename);
Index created.
```

When to Create an Index

- The column is used frequently in the WHERE clause or in a join condition.
- The column contains a wide range of values.
- The column contains a large number of null values.
- Two or more columns are frequently used together in a WHERE clause or a join condition.
- The table is large and most queries are expected to retrieve less than 2–4% of the rows.

When Not to Create an Index

- The table is small.
- The columns are not often used as a condition in the query.
- Most queries are expected to retrieve more than 2–4% of the rows.
- The table is updated frequently.

Confirming Indexes

- The USER_INDEXES data dictionary view contains the name of the index and its uniqueness.
- The USER_IND_COLUMNS view contains the index name, the table name, and the column name.

```
SQL> SELECT ic.index_name, ic.column_name,
  2           ic.column_position col_pos,ix.uniqueness
  3  FROM   user_indexes ix, user_ind_columns ic
  4  WHERE  ic.index_name = ix.index_name
  5  AND       ic.table_name = 'EMP';
```

Removing an Index

- Remove an index from the data dictionary.

```
SQL> DROP INDEX index;
```

- Remove the EMP_ENAME_IDX index from the data dictionary.

```
SQL> DROP INDEX emp_ename_idx;
Index dropped.
```

- To drop an index, you must be the owner of the index or have the DROP ANY INDEX privilege.

Synonyms

- Simplify access to objects by creating a synonym (another name for an object).
    - Refer to a table owned by another user.
    - Shorten lengthy object names.

```
CREATE [PUBLIC] SYNONYM synonym
FOR    object;
```

Creating and Removing Synonyms

- Create a shortened name for the EAST_ORDER table.

```
SQL> CREATE SYNONYM  eord
  2  FOR             east_order;
Synonym Created.
```

- Drop a synonym

```
SQL> DROP SYNONYM eord;
Synonym dropped.
```

Summary

- Automatically generate sequence numbers by using a sequence generator.
- View sequence information in the USER_SEQUENCES data dictionary table.
- Create indexes to improve query retrieval speed.
- View index information in the USER_INDEXES dictionary table.
- Use synonyms to provide alternative names for objects.