Ex. No.: 02
Date : **Foreign Key Constraints**

Aim:

To study and practice to create table, add foreign key constraints and incorporate referential integrity.

Description:

A referential integrity constraint is also known as **foreign key constraint**. A foreign key is a key whose values are derived from the Primary key of another table.

The table from which the values are derived is known as **Master or Referenced** Table and the Table in which values are inserted accordingly is known as **Child or Referencing** Table,

In other words, we can say that the table containing the **foreign key** is called the **child table**, and the table containing the **Primary key/candidate key** is called the **referenced or parent table**.

Procedure:

Step 1: Create the master or referenced table with required fields.
Step 2: Create the child table.
Step 3: Create the primary key in master table.
Step 4: Apply the insert and delete constrains.

CONSTRAINTS:

1) Primary key
2) Foreign key/references
3) Check
4) Unique
5) Not null
6) Null
7) Default

CONSTRAINTS CAN BE CREATED IN THREE WAYS:

1) Column level constraints
2) Table level constraints
3) Using DDL statements-alter table command

OPERATION ON CONSTRAINT:

i) ENABLE
ii) DISABLE
iii) DROP

NOT NULL:

Syntax:

```
Create table tablename(
                    fieldname1 datatype(constraint)not null,
                    fieldname2 datatype,
```

…….
fieldnamen datatype);

Example:

SQL> create table notnull (eno varchar(10) not null, ename varchar(10),esalary number(20));

Table created
SQL>insert into notnull values('1','abdul','20000')
1 row created.

SQL>insert into notnull values('','raj','30000')

*

ERROR at line 1:

ORA-01400: cannot insert NULL into ("SCOTT"."NOTNULL"."ENO")

CHECK:

Check constraint specify conditions that each tuple must satisfy.

Syntax:

Create table tablename(
Fieldname1 datatype(constraint),
Fieldname2 datatype,
…….
Fieldname3 datatype);

Example:

SQL> create table con (empid decimal(10) not null, empname varchar(20),empsalary decimal(10), check(empsalary>10000));

SQL>insert into con values ('1','kumar','20000')
1 row created

SQL>insert into con values('2','raja','9000')

*

ERROR at line 1:

ORA-02290: check constraint (SCOTT.SYS_C0010283) violated

UNIQUE:

Used to set unique constraint to the specified column name which will not allow redundant values

Syntax:

Create table tablename(
fieldname1 datatype(constraint)unique,
fieldname2 datatype,

…….
Fieldname3 datatype);

Example:

SQL> create table conn(eno varchar(10) unique, ename varchar(20));

Table created.
SQL> insert into conn values('1','hello')
1 row created.

SQL>insert into conn values('1','hi')

*

ERROR at line 1:

ORA-00001: unique constraint (SCOTT.SYS_C0010285) violated

PRIMARY KEY:
Primary key is a constraint for both unique and not null.

Syntax:

Create table tablename(
Fieldname1 datatype(constraint)primary key,
fieldname2 datatype,
…….
Fieldname3 datatype);

Example:

SQL> create table con(empid varchar(10),empname varchar(20) primary key);
Table created.

ADDING CONSTRIANT:

Used to set any constraint to the specified column at the last by specifying the constraint type and field name.

Syntax:

Create table tablename(
Fieldname1 datatype(constraint),

fieldname2 datatype,
constraint constraintname constrainttype(fieldname));

Example:

SQL> create table con(empid varchar(10),empname varchar(10),constraint c1 primary key(empid));

Table created.
SQL> insert into con values (‘1’,’anand’)

SQL>insert into con values (‘1’,’vijay’)

*

ERROR at line 1:

ORA-00001: unique constraint (SCOTT.C1) violated

ADD CONSTRAINT (ALTER)

Used to set the constraint for the table already created by using alter command.

Syntax:

Alter table tablename add constraint constraintname (fieldname)datatype,primary key.

Example:

SQL> create table con(empid varchar(10),empname varchar(10));

Table created.

SQL> alter table con add constraint c1 primary key (empid);

Table altered.

SQL> desc con;

Name	Null? Type
EMPID	NOT NULL VARCHAR(10)
EMPNAME	VARCHAR(10)

DROP CONSTRAINT:

Used to drop the constraint.

Syntax:

Alter table table_name drop constraint constraint_name.

Example:

SQL> alter table con drop constraint c1;

Table altered.

SQL> desc con;

Name	Null? Type
EMPID	VARCHAR(10)
EMPNAME	VARCHAR(10)

REFERENTIAL INTEGRITY:
Used to refer the primary key of the parent table from the child table.

Syntax:

a)Create table tablename(
Fieldname1 datatype primary key,
fieldname2 datatype,
…….
Fieldname3 datatype);

b) Create table tablename(Fieldname1 datatype references,
Parent tablename(fieldname)
…….
Fieldname n datatype);

Example:

SQL> create table parent(eno varchar(10),ename varchar(10) primary key);
Table created.

SQL>insert into parent values (‘1’,’ajay’)
1 row created.
SQL>insert into parent values (‘2’,’bala’)
1 row created.

SQL> create table child (eno varchar(10),ename varchar(10) references parent(ename));

Table created.

SQL>insert into child values (‘1’,’ajay’)
1 row created.

SQL>insert into child values (‘2’,’balaji’)

ERROR at line 1:

ORA-02291: integrity constraint (SCOTT.SYS_C0010290) violated - parent key not

Found

ON DELETE CASCADE:
The changes done in parent table is reflected in the child table when references are made.

Syntax:

Create table tablename(
Fieldname1 datatype references,
Parent tablename(fieldname),
On delete cascade);

Example:

SQL> create table parent(eno varchar(10),ename varchar(10) primary key);
Table created.

SQL>insert into parent values (‘1’,’a’)

1 row created.

SQL> create table child(eno varchar(10),ename varchar(10) references parent(ename) on delete cascade);

Table created.

SQL> insert into child values (‘2’,’a’)

1 row created.
SQL> select * from parent;

ENO ENAME
1 a

SQL> select * from child;

ENO ENAME
2 a

SQL> delete from parent where eno=1;

1 row deleted.

SQL> select * from parent;

no rows selected

SQL> select * from child;

no rows selected

Result

Thus the various key constraints based on foreign key where written and executed successfully.