**EXERCISE 4 : INTEGRITY CONSTRAINTS**

***Aim :*** • Study & Implementation of different types of constraints

##### Theory:

**CONSTRAINTS:**

Constraints are used to specify rules for the data in a table. If there is any violation between the constraint and the data action, the action is aborted by the constraint. It can be specified when the table is created (using CREATE TABLE statement) or after the table is created (using ALTER TABLE statement).

1. **NOT NULL:** When a column is defined as NOTNULL, then that column becomes a mandatory column. It implies that a value must be entered into the column if the record is to be accepted for storage in the table.

##### Syntax:

**CREATE TABLE** Table\_Name (column\_name data\_type (*size*) **NOT NULL,** );

##### Example:

**CREATE TABLE** student (sno **NUMBER(3)NOT NULL,** name **CHAR**(**10**));

1. **UNIQUE:** The purpose of a unique key is to ensure that information in the column(s) is unique i.e. a value entered in column(s) defined in the unique constraint must not be repeated across the column(s). A table may have many unique keys.

##### Syntax:

**CREATE TABLE** Table\_Name(column\_name data\_type(*size*) **UNIQUE, ….**);

##### Example:

**CREATE TABLE** student (sno **NUMBER(3) UNIQUE,** name **CHAR**(**10**));

1. **CHECK:** Specifies a condition that each row in the table must satisfy. To satisfy the constraint, each row in the table must make the condition either TRUE or unknown (due to a null).

##### Syntax:

**CREATE TABLE** Table\_Name(column\_name data\_type(*size*) **CHECK(*logical expression*), ….**);

##### Example:

**CREATE TABLE** student (sno **NUMBER (3),** name **CHAR**(**10**),class **CHAR(5),CHECK**(class **IN**(‘CSE’,’CAD’,’VLSI’));

1. **PRIMARY KEY:** A field which is used to identify a record uniquely. A column or combination of columns can be created as primary key, which can be used as a reference from other tables. A table contains primary key is known as Master Table.

##### Syntax:

**CREATE TABLE** Table\_Name(column\_name data\_type(*size*) **PRIMARY KEY,**

**….**);

##### Example:

**CREATE TABLE** faculty (fcode **NUMBER(3) PRIMARY KEY,** fname

**CHAR**(**10**));

1. **FOREIGN KEY:** It is a table level constraint. We cannot add this at column level. To reference any primary key column from other table this constraint can be used. The table in which the foreign key is defined is called a **detail table**. The table that defines the primary key and is referenced by the foreign key is called the **master table**.

***Syntax:* CREATE TABLE** Table\_Name(column\_name data\_type(*size*)

**FOREIGN KEY**(column\_name) **REFERENCES** table\_name);

##### Example:

**CREATE TABLE** subject (scode **NUMBER (3) PRIMARY KEY,** subname

**CHAR**(**10**),fcode **NUMBER(3), FOREIGN KEY**(fcode) **REFERENCE** faculty );

#### Defining integrity constraints in the alter table command:

***Syntax:* ALTER TABLE** Table\_Name **ADD PRIMARY KEY** (column\_name);

***Example:* ALTER TABLE** student **ADD PRIMARY KEY** (sno); (Or)

***Syntax:* ALTER TABLE** table\_name **ADD CONSTRAINT** constraint\_name

**PRIMARY KEY**(colname)

***Example:* ALTER TABLE** student **ADD CONSTRAINT** SN **PRIMARY KEY(**SNO**)**

#### Dropping integrity constraints in the alter table command:

***Syntax:* ALTER TABLE** Table\_Name **DROP** constraint\_name;

***Example:* ALTER TABLE** student **DROP PRIMARY KEY**;

(or)

***Syntax:* ALTER TABLE** student **DROP CONSTRAINT** constraint\_name**;**

***Example:* ALTER TABLE** student **DROP CONSTRAINT** SN**;**

1. **DEFAULT** : The DEFAULT constraint is used to insert a default value into a column. The default value will be added to all new records, if no other value is specified.

#### Syntax:

**CREATE TABLE** Table\_Name(col\_name1,col\_name2,col\_name3 DEFAULT ‘<value>’);

Example:

**CREATE TABLE** student (sno **NUMBER(3) UNIQUE,** name **CHAR**(**10**),address **VARCHAR(20) DEFAULT** ‘Chennai’);