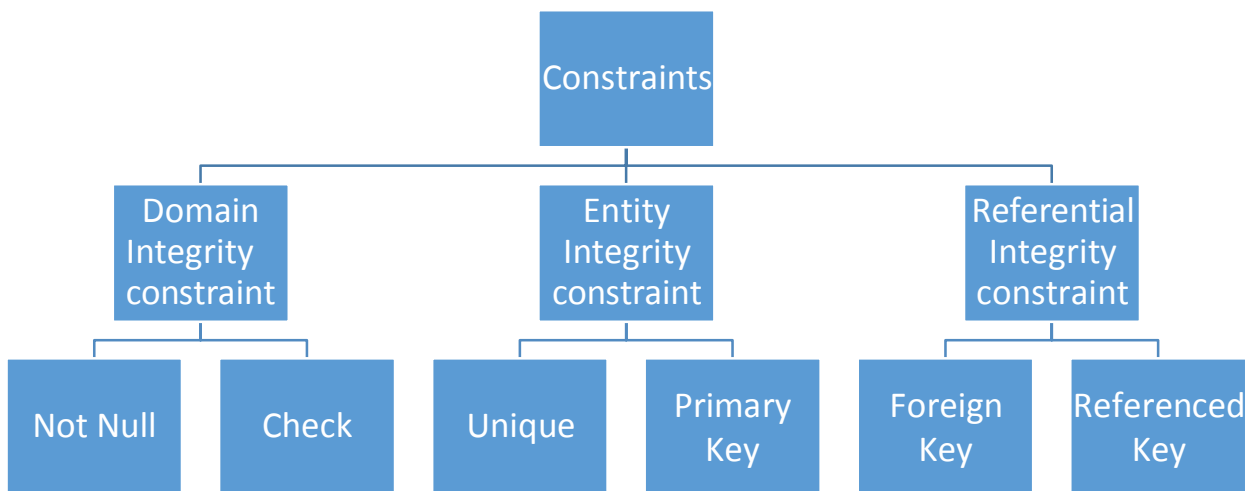


3. CONSTRAINTS

❖ DEFINATION

- **“Constraints are rules which are enforced on data that is being entered and prevents the user from entering invalid data into tables”**
- The different types of constraints are:



3.1 Domain Integrity constraint

❖ NOT NULL

- When a column is defined as NOT NULL, it means that a value must be entered into the column if the record is to be accepted for storage in table
- Syntax: columnname datatype(size) NOT NULL
- Example:
Create table client_master
 (client_no varchar2(6) NOT NULL,
 Name varchar2(20) NOT NULL,
 Address varchar2(30) NOT NULL,

```
City varchar2(15),  
State varchar2(15),  
Pincode number(6));
```

- In the above query the user has to specify values for the client_no, name, and address fields, otherwise the record will not be inserted into the table.
- NOTE: NOT NULL constraint can be defined at column level only.

❖ **CHECK Constraint**

- A check constraint is used to apply business rule validations to a table column

- **Check constraint at column level:**

- Syntax: columnname datatype(size) CHECK(logical expression)

- Example:

Create table client_master

```
(client_no varchar2(6) CHECK ( client_no like 'C%'),  
Name varchar2(20) CHECK (name = upper(name)),  
Address varchar2(30),  
City varchar2(15) CHECK (city IN ( 'delhi', 'mumbai',  
'chennai')),  
State varchar2(15),  
Pincode number(6));
```

- **Check constraint at table level:**

- Syntax: CHECK(logical expression)

- Example:

Create table client_master

```
(client_no varchar2(6), Name varchar2(20),  
Address varchar2(30), City varchar2(15),  
State varchar2(15), Pincode number(6),  
CHECK ( client_no like 'C%'),  
CHECK (name = upper(name)),  
CHECK (city IN ( 'delhi', 'mumbai', 'chennai')));
```

- In the above queries, the following rules are applied using the check constraint :
 - Data values inserted into the client_no column must start with the capital letter 'C'
 - Data values inserted into name column must be in upper case only
 - Data values in city column can be either 'delhi', 'mumbai' or 'chennai'
 - If the check constraint evaluates to FALSE then the processing stops and error message is displayed

- **Limitations of CHECK constraint**

- The condition must be a Boolean expression that can be evaluated using the values in the row being inserted or updated
- The condition cannot contain subqueries

- The condition cannot include SYSDATE, UID or USER

3.2 Entity Integrity constraint

❖ **UNIQUE constraint**

- The purpose of UNIQUE key is to ensure that information in the column is UNIQUE.
- A value entered in column defined in the UNIQUE constraint must not be repeated across the column
- A table may have many UNIQUE keys

- **UNIQUE constraint defined at the column level:**

- Syntax: columnname datatype(size)UNIQUE

- Example:

```
Create table client_master
(client_no varchar2(6)UNIQUE,
Name varchar2(20),
Address varchar2(30),
City varchar2(15),
State varchar2(15),
Pincode number(6));
```

- **UNIQUE constraint defined at the table level:**

- Syntax: UNIQUE (columnname , columnname..)

- Example:

```
Create table client_master
(client_no varchar2(6),
Name varchar2(20),
Address varchar2(30),
City varchar2(15),
State varchar2(15),
Pincode number(6),UNIQUE(client_no));
```

- In the above queries the client_no column can take only UNIQUE values
- If a value is repeated, an error message is displayed and the record is not inserted

❖ **PRIMARY KEY constraint**

- A primary key is one or more column in a table that is used to uniquely identify each row in the table
- The column in which primary key is set cannot be left blank
- The data stored in the column must be Unique

- There are two types of primary key :
 - **Simple Primary Key** – A single column primary key is called Simple Primary Key
 - **Composite Primary Key** – A Primary Key set on multiple columns is called Composite Primary Key

- **Primary Key defined at the column level:**

- Syntax : columnlevel datatype(size) PRIMARY KEY

- Example:

```
Create table sales_order
(Order_no varchar2(6) PRIMARY KEY,
Order_date date,
Client_no varchar2(6),
Dely_addr varchar2(25),
Salesman_no varchar2(6),
Dely_type varchar2(1),
Dely_date date, order_status varchar2(10));
```

- In the above query, the order_no field of the sales+order table has been assigned the Primary Key
- It means that this field cannot take Null values and all the values should be unique
- This is also an example of a simple Primary Key

- **Primary Key defined at the table level:**

- Syntax : PRIMARY KEY (columnname, columnname,...)

- Example:

```
Create table sales_order_details
(detlorder_no varchar2(6),
Product_no varchar2(6),
Qty_ordered varchar2(8),
Qty_disp number(8),
Product_rate number(8,2),
PRIMARY KEY ( detlorder_no,product_no));
```

- In the above query a composite Primary Key has been set on detlorder_no and product_no columns of the sales_order_details table
- This means that both these columns together uniquely identify each data value

3.3 Referential Integrity constraint

❖ FOREIGN KEY constraint

- A Foreign key represents a relationship between tables
- A foreign key is column or a group of columns whose values are derived from the primary key of other table
- Foreign table - The table in which the foreign key is defined is called the foreign table or detail table
- Master table – The table that defines the primary key and is referenced by the foreign key is called the primary table or master table
- The master table can be referenced in the foreign key definition by using REFERENCES keyword
- If an insert or update operation is performed in the foreign table, the corresponding data value must exist in the primary table, otherwise the operation is not allowed
- If a delete operation is performed on the primary table, then the data value must first be deleted from the foreign table. Otherwise the operation is not allowed
- If data from the primary table and the foreign table is to be deleted then the ON DELETE CASCADE option should be specified

➤ **Foreign key constraint defined at the column level:**

- Syntax: columnname datatype(size) REFERENCES tablename
 (columnname) ON DELETE CASCADE

► Example:

Create table sales_order_details

```
( detlorder_no varchar2(6) REFERENCES sales_order(order_no)
,product_no varchar2(6), qty_ordered number(8), qty_disp number(8),
product_rate number(8,2));
```

- The REFERENCES keyword points to the table sales_order
- The table sales_order has the column order_no as Primary Key

➤ **Assigning user defined names to constraints:**

- A constraint can be given a user-defined name by preceding the constraint definition with keyword **CONSTRAINT** and a user-defined name
- Syntax: **CONSTRAINT** constraintname constraintdefinition

▪ Example

3.3.1 Create table client_master

```
(client_no varchar2(6) CONSTRAINT p_clientkey("constraintname")  
PRIMARY KEY, Name varchar2(20),  
Address varchar2(30),  
City varchar2(15), State  
varchar2(15), Pincode  
number(6));
```

3.3.2 create table sales_order_details

```
( detloredr_no varchar2(6) REFERENCES  
Sales_order(order_no),  
Product_no varchar2(6), qty_ordered number(8),  
Qty_disp number(8),product_rate number(8,2),  
CONSTRAINT f_orderkey ("constraintname")  
FOREIGN KEY (detlorder_no)  
REFERENCES sales_order);
```

REVIEW QUESTION

1) Explain Domain Integrity constraints.

OR

Not null, Check Constraints

2) Explain Entity Integrity constraints

OR

Unique, primary key Constraints.

3) Explain Referential integrity constraints.

OR

Foreign key referenced key Constraints.

