

Summary – Day 6

SQL

Constraints:

- The constraint in MySQL is used to specify the rule that allows or restricts what values/data will be stored in the table. They provide a suitable method to ensure data accuracy and integrity inside the table. It also helps to limit the type of data that will be inserted inside the table. If any interruption occurs between the constraint and data action, the action is failed.

Types of Constraints:

- Column Level Constraints: These constraints are applied only to the single column that limits the type of particular column data.
- Table Level Constraints: These constraints are applied to the entire table that limits the type of data for the whole table.

How to create the constraints ?

```
CREATE TABLE new_table_name (  
    col_name1 datatype constraint,  
    col_name2 datatype constraint,  
    col_name3 datatype constraint,  
    .....  
);
```

Types:

NOT NULL:

This constraint specifies that the column cannot have NULL or empty values. The below statement creates a table with NOT NULL constraint.

```
CREATE TABLE Student(Id INTEGER, LastName TEXT NOT NULL, FirstName TEXT NOT NULL, City VARCHAR(35));
```

UNIQUE:

This constraint ensures that all values inserted into the column will be unique. It means a column cannot store duplicate values. MySQL allows us to use more than one column with UNIQUE constraint in a table. The below statement creates a table with a UNIQUE constraint:

```
CREATE TABLE ShirtBrands(Id INTEGER, BrandName VARCHAR(40) UNIQUE, Size VARCHAR(30));
```

CHECK:

It controls the value in a particular column. It ensures that the inserted value in a column must be satisfied with the given condition. In other words, it determines whether the value associated with the column is valid or not with the given condition.

```
mysql> CREATE TABLE Persons (  
    ID int NOT NULL,  
    Name varchar(45) NOT NULL,  
    Age int CHECK (Age >= 18)  
);
```

DEFAULT:

This constraint is used to set the default value for the particular column where we have not specified any value. It means the column must contain a value, including NULL.

```
mysql> CREATE TABLE Persons (  
    ID int NOT NULL,  
    Name varchar(45) NOT NULL,  
    Age int,  
    City varchar(25) DEFAULT 'New York'  
);
```

PRIMARY KEY:

This constraint is used to identify each record in a table uniquely. If the column contains primary key constraints, then it cannot be null or empty. A table may have duplicate columns, but it can contain only one primary key. It always contains unique value into a column.

```
CREATE TABLE Persons (  
    ID int NOT NULL PRIMARY KEY,  
    Name varchar(45) NOT NULL,  
    Age int,  
    City varchar(25));
```

ENUM:

The ENUM data type in MySQL is a string object. It allows us to limit the value chosen from a list of permitted values in the column specification at the time of table creation. It is short for enumeration, which means that each column may have one of the specified possible values. It uses numeric indexes (1, 2, 3...) to represent string values.

```
mysql> CREATE TABLE Shirts (  
    id INT PRIMARY KEY AUTO_INCREMENT,  
    name VARCHAR(35),  
    size ENUM('small', 'medium', 'large', 'x-large')  
);
```

FOREIGN KEY:

This constraint is used to link two tables together. It is also known as the referencing key. A foreign key column matches the primary key field of another table. It means a foreign key field in one table refers to the primary key field of another table.

```
CREATE TABLE Orders (  
    Order_ID int NOT NULL PRIMARY KEY,  
    Order_Num int NOT NULL,  
    Person_ID int,  
    FOREIGN KEY (Person_ID) REFERENCES Persons(Person_ID)  
);
```

What is AUTO_INCREMENT ?

This constraint automatically generates a unique number whenever we insert a new record into the table. Generally, we use this constraint for the primary key field in a table.

```
mysql> CREATE TABLE Animals(  
    id int NOT NULL AUTO_INCREMENT,  
    name CHAR(30) NOT NULL,  
    PRIMARY KEY (id));
```