Queries with Tables & Constraints

- **Create table clause:**

Syntax - 1: Both Constraint and Key are declared with column definition.

```
CREATE TABLE <table-name> (
<column-name-1> <data-type>,
<column-name-2> <data-type> CONSTRAINT,
<column-name-3> <data-type> KEY,
<column-name-4> <data-type> CONSTRAINT KEY
);
```

Syntax - 2: Key declared at the end.

```
CREATE TABLE <table-name> (
<column-name-1> <data-type>,
<column-name-2> <data-type>,
<column-name-3> <data-type> CONSTRAINT,
KEY_NAME1(<column-name1>),
KEY_NAME2(<column-name2>)
);
```

Syntax - 3: Both Constraint and Key declared at the end.

```
CREATE TABLE <table-name> (
<column-name-> <data-type>,
<column-name-> <data-type>,
KEY_NAME1(<column-name>),
KEY_NAME2(<column-name>),
CONSTRAINT1(<column-name>)
);
```

- **ALTER table clause:**

The various variations in ALTER table clause are explained in the following.

a. Add column

Syntax:

```
ALTER TABLE <table-name>
ADD <column-name> <data-type>;
```

b. Drop column

Syntax:

ALTER TABLE *<table-name>*
DROP COLUMN *<column-name>;*

c. Modify column

Syntax-1:
ALTER TABLE *<table-name>*
MODIFY COLUMN *<column-name> <data-type>;*

Syntax-2:
ALTER TABLE *<table-name>*
ALTER COLUMN *<column-name> <data-type>;*

Below is a customer table with details related to the customer:

id	branch_id	first_name	last_name	DOB	gender

This is the table we want to create with Primary Key as Id, below is the query for it -

CREATE TABLE customer
(
id INT NOT NULL,
branch_id INT NOT NULL,
first_name VARCHAR(20),
last_name VARCHAR(20),
DOB INT,
gender CHAR(6),
PRIMARY KEY(id)
);

In the above query, we have declared id as PRIMARY KEY.

● What is the Primary key?

o Primary Key helps us to uniquely identify tuples of a relation. It cannot be NULL

and have UNIQUE values.

o Each table has only one primary key. But a primary key can have one or more

columns(fields) as part of the Primary Key.

o In case we have multiple fields as Primary key it is called **Composite Key** and all the conditions that apply to a single field Primary Key applies to multiple field Composite Key as well.

o Let's say for the same customer table we want id and branch id as PRIMARY KEY we can use the below-mentioned declaration if we haven't declared the table yet -

```
CREATE TABLE customer
(
 id INT NOT NULL,
 branch_id INT NOT NULL,
 first_name VARCHAR(20),
 last_name VARCHAR(20),
 DOB INT,
 gender CHAR(6),
 PRIMARY KEY(id, branch_id)
);
```

Let's say we have declared table, as mentioned below:

```
CREATE TABLE customer
(
id INT NOT NULL,
branch_id INT NOT NULL,
first_name VARCHAR(20),
last_name VARCHAR(20),
DOB INT,
gender CHAR(6)
);
```

In case we want to make any attribute a Primary Key we need to use ALTER keyword for that as mentioned below -

```
ALTER TABLE CUSTOMER
ADD PRIMARY KEY (ID);
```

Note:- Primary Key column should already have been declared as NOT NULL at the time of creating the table.

Similarly, we can declare the composite primary key (assuming columns or attributes we want to use are declared as NOT NULL) as done below -

ALTER TABLE CUSTOMERS
ADD CONSTRAINT Pkey_Custid PRIMARY KEY (ID, NAME);

We have learned to declare the primary key and all the conditions associated with column(s) to be the Primary key now let us discuss **how to delete a primary key** - Let us say have declared a table as below mentioned -

CREATE TABLE customer
(
 id INT NOT NULL,
 branch_id INT NOT NULL,
 first_name VARCHAR(20),
 last_name VARCHAR(20),
 DOB INT,
 gender CHAR(6),
 PRIMARY KEY(id)
);

Now we want to delete id primary key we will use ALTER and DROP keyword as mentioned below -

ALTER TABLE CUSTOMERS
DROP PRIMARY KEY;

- **What is Foreign Key?**

Foreign Key refers to the Primary Key of another table. Each table can have any number of foreign key(s). Let's us say we have a customer and account table with details mentioned below -

Table : customer

id	branch_id	first_name	last_name	DOB	gender

Table : account

id	balance	customer_id

Declaring customer table -

```
CREATE TABLE customer
(
id INT NOT NULL,
branch_id INT NOT NULL,
first_name VARCHAR(20),
last_name VARCHAR(20),
DOB INT,
gender CHAR(6),
PRIMARY KEY(id)
);
```

Declaring account table with Primary Key and Foreign Key -

```
CREATE TABLE account
(
id INT NOT NULL,
balance INT,
customer_id INT NOT NULL,
PRIMARY KEY(id),
FOREIGN KEY(customer_id) REFERENCES customer(id) );
```

- **What is a UNIQUE constraint?**

UNIQUE constraints make sure that all the values in a column which is declared

UNIQUE are different.

Then, what is the difference between UNIQUE and PRIMARY KEY?

UNIQUE	PRIMARY KEY
UNIQUE constraint column need not be a PRIMARY KEY.	PRIMARY KEY constraint automatically has a UNIQUE constraint.
Multiple columns in a table can have a UNIQUE constraint	There is only one PRIMARY KEY in a table

Below we have mentioned an example of a UNIQUE Constraint –

CREATE TABLE customer

(

id INT NOT NULL,

name VARCHAR(255) NOT NULL,

UNIQUE(id)

);

Using UNIQUE constraint in multiple columns

CREATE TABLE customer
(
id INT NOT NULL,
name VARCHAR(255) NOT NULL,
CONSTRAINT UNIQUE(id, name)
);

- **What is the CHECK constraint?**
The CHECK is used to put limitations on the value range that we can put
in a column.
For example let's say we have the account table as given below,

account_id	balance	customer_id

If we want only customers with a minimum balance of let us say 3000 or more

we can use the CHECK constraint to ensure that.

CREATE TABLE account
(
account_id INT NOT NULL,
balance INT NOT NULL,
customer_id INT NOT NULL,
CHECK(balance >= 3000)
);

- **What is the DEFAULT constraint?**

DEFAULT constraint sets a default value for a column. This default value is added to all the new records unless other value is specified.

Let us say for all the accounts we want to have a balance of 100 when an account is added we can make sure of this by using the DEFAULT constraint

CREATE TABLE account
(
account_id INT NOT NULL,
balance INT DEFAULT 100,
customer_id INT NOT NULL,
);
We can drop the default constraint by using DROP keyword –

ALTER TABLE account
ALTER balance DROP DEFAULT;

Also, if suppose we haven't initialised the balance we can later modify as done below-

ALTER TABLE account
MODIFY balance DEFAULT 100;

Below mentioned table summarises the constraint -

Constraint	Description
CHECK	Determine whether the value is valid or not from a logical expression
FOREIGN KEY	Link between two tables by one specific column of both tables. The specified column in one table must be a PRIMARY KEY and referred by the column of another table known as a FOREIGN KEY.
UNIQUE	Maintains the uniqueness of a column in a table. More than one UNIQUE column can be used in a table.
NOT NULL	column can not contain any NULL value
PRIMARY KEY	Enforces the table to accept unique data for a specific column and is a unique index for accessing the table faster.

- ### What is TRUNCATE? How is it different from DROP and DELETE?

TRUNCATE removes all the records from the table. Below is the given general form -

TRUNCATE TABLE account;

After running this query all the data from the account table will be cleared but the table will still exist.

Whereas in the case of DROP the whole table will be dropped from the database i.e table record and schema will be cleared from the database.

Also, DELETE is used to delete the record of a particular row or conditionals -

DELETE FROM account where account_id = XYZ;

- **Difference between Delete, Drop and Truncate:**

Delete	Drop	Truncate
DML command	DDL command	DDL command
Removes one, some or all the records in the table.	Removes the entire table structure.	Removes all the records from the table.
Is a slow operation	Relatively faster	Fastest of all.

- **ON DELETE CASCADE:**

ON DELETE CASCADE is an option that can be added to the table while creating constraints. It is used to delete rows from the child table automatically when similar rows in the parent table get deleted.

- **ON UPDATE CASCADE:**

ON UPDATE CASCADE is an option that can be added to the table while creating constraints. Suppose there is a situation where we have two tables such that primary key of one table is the foreign key for another table. if we update the primary key of the first table then using the ON UPDATE CASCADE foreign key of the second table automatically get updated.

- **How to rename the table and column?**

Rename a table from cust to customers –

ALTER TABLE cust
RENAME TO customers;

Rename column name to the surname in customers table -

ALTER TABLE customers
RENAME name TO surname;

Information Schema

We have an in-built view in MySQL called information_schema, which holds all details about the constraints on a particular table. INFORMATION_SCHEMA provides access to information about the MySQL server, such as database metadata, database or table names, column data types, and permissions.

General form:

```
SELECT COLUMN_NAME, CONSTRAINT_NAME, REFERENCED_COLUMN_NAME,
REFERENCED_TABLE_NAME
FROM information_schema.KEY_COLUMN_USAGE
WHERE TABLE_NAME = 'table_name';
```