**Lesson 1: Select,Create and Insert**

create table Employee

(

EmpID varchar(255),

EmpName varchar(255),

Age int,

PhoneNumber int,

EmailID varchar(255),

Address varchar(255)

);

insert into Employee values(1,"Tom",45,3839202,"tom@gmail.com","hollywood avenue, LA");

insert into Employee values(3,"Dick",40,3839202,"dick@gmail.com","hollywood avenue, LA");

insert into Employee values(2,"Harry",35,3839202,"harry@gmail.com","hollywood avenue, LA");

insert into Employee values(2,"Harry",35,3839202,"harry@gmail.com","hollywood avenue, LA");

insert into Employee values(4,"Harry",35,3839202,"harry@gmail.com","hollywood avenue, LA");

--Select \* from Employee;

--select count(\*) from Employee;

--Select \* from Employee where EmpName="Tom";

--Select \* from Employee where EmpName="Harry" and age="45";

--Select \* from Employee where Age>35;

--select distinct \* from Employee;

select count(distinct EmpID) from Employee;

select count(distinct EmpName) from Employee;

**Lesson 2: Order By, And/Or/Not**

--Select \* from Employee order by EmpID ASC;

--Select \* from Employee order by EmpID DESC;

--Select \* from Employee order by Age DESC;

--Select \* from Employee order by Age;

--Select \* from Employee order by EmpName,Age;

--Select \* from Employee where age>40 or EmpID>2;

Select \* from Employee where age>35 and EmpID>2 and PhoneNumber=3839202;

Create table Customer(

ID int,

Name varchar(255),

PhoneNumber int,

EmailID varchar(255),

City varchar(255),

Country varchar(255)

);

insert into Customer values(10,"Tom",3839200,"tom@gmail.com","Jersey City","USA");

insert into Customer values(20,"Dick",3839202,"dick@gmail.com","Bengaluru","India");

insert into Customer values(30,"Harry",3839209,"harry@gmail.com","Berlin","Germany");

insert into Customer values(40,"Harry",3839206,"harry@gmail.com","Tokyo","Japan");

insert into Customer values(50,"Harry",3839204,"harry@gmail.com","Seoul","South Korea");

insert into Customer values(60,"Tom",3839200,"tom@gmail.com","New York","USA");

--Select \* from Customer where country="USA" OR Country="India";

--Select \* from Customer where Not country="Germany";

--Select \* from Customer where country="USA" AND (City="Jersey City" OR City="New York");

Select \* from Customer where country="USA" AND NOT Country="Germany";

**Lesson 3: Like,Null and Not Null**

--% and \_

--LIKE

--Select \* from customer where Name Like 'Tom%';

--Select \* from customer where Name Like '%h';

--Select \* from customer where Name Like '%ar%';

--Select \* from customer where Name Like '\_i%';

--Select \* from customer where Name Like 'T\_\_%';

--Select \* from customer where Name Like 'T%y';

--IS NULL

--Select \* from customer where City IS NULL;

--IS NOT NULL

--Select \* from customer where PhoneNumber IS NOT NULL;

Select \* from customer where PhoneNumber IS NULL;

**LESSON 4:MAX,LIMIT in sql**

create table Employee

(

EmpID varchar(255),

EmpName varchar(255),

Age int,

PhoneNumber int,

EmailID varchar(255),

Address varchar(255),

salary int

);

insert into Employee values(1,"Tom",45,3839202,"tom@gmail.com","hollywood avenue, LA",10000);

insert into Employee values(3,"Dick",40,3839202,"dick@gmail.com","hollywood avenue, LA",40000);

insert into Employee values(2,"Harry",35,3839202,"harry@gmail.com","hollywood avenue, LA",5000);

insert into Employee values(2,"Harry",35,3839202,"harry@gmail.com","hollywood avenue, LA",8000);

insert into Employee values(4,"Harry",35,3839202,"harry@gmail.com","hollywood avenue, LA",700000);

--Select max(salary) from Employee;

--Inner Query:

--2nd highest salary:

--Select max(salary) from Employee where salary<(select max(salary) from Employee);

--3rd highest salary:

--Select max(salary) from Employee where salary < (select max(salary) from Employee where salary < (select max(salary) from Employee));

--2nd --1 IQ

--3rd --2 IQ

--Nth --N-1 IQ

--LIMIT:

--Select \* from Employee LIMIT 2;

--Select salary from employee order by salary desc Limit 5-1,1;

--TOP ROWNUM

--Select Top 2 from employee;

Select \* from Employee where ROWNUM<=2;

**LESSON 5: Primary Key and Not Null contraints**

**--SQL Constraints**

--Rules --DATA

--NOT NULL

Create Table Employee(

ID int not null,

firstname varchar(255) not null,

lastname varchar(255) not null,

Age int

);

Insert into Employee values(1,"Tom","Harris",25);

Insert into Employee values(2,"Deeps","Gupta",NULL);

Select \* from Employee;

--PRIMARY KEY: CONSTRAINT:

--MUST CONTAIN UNIQUE VALUES

--CAN NOT HOLD ANY NULL DATA

--CAN HAVE ONLY ONE PRIMARY KEY

--BUT CAN CONTAIN SINGLE AND MULTIPLE COLUMNS

--MY SQL:

Create Table Student(

ID int not null,

firstname varchar(255) not null,

lastname varchar(255) not null,

Age int,

Primary key (ID)

);

Insert into Student values(1,"Tom","Singh",28);

Insert into Student values(2,"Tom","Singh",28);

Select \* from Student;

=====================================================================================

--SQL SERVER,ORACLE

Create table Employee(

ID int not null Primary Key,

FIRSTNAME VARCHAR(255) NOT NULL,

LASTNAME VARCHAR(255) NOT NULL,

AGE INT

);

INSERT INTO EMPLOYEE VALUES(1,"MEE","YOU",20);

SELECT \* FROM EMPLOYEE;

=====================================================================================

--PRIMARY KEY ON MULTIPLE COLUMNS

Create table Employee(

ID INT NOT NULL,

FIRSTNAME VARCHAR(255) NOT NULL,

LASTNAME VARCHAR(255) NOT NULL,

AGE INT,

CONSTRAINT PK\_EMPLOYEE PRIMARY KEY(ID, LASTNAME)

);

INSERT INTO EMPLOYEE VALUES(1,"MEE","YOU",20);

INSERT INTO EMPLOYEE VALUES(2,"MEE","YOU",20);

SELECT \* FROM EMPLOYEE;

--DROP PRIMARY KEY ON AN EXISTING TABLE:

ALTER TABLE EMPLOYEE DROP PRIMARY KEY;

====================================================================================

--PRIMARY KEY ON MULTIPLE COLUMNS

Create table Employee(

ID INT NOT NULL,

FIRSTNAME VARCHAR(255) NOT NULL,

LASTNAME VARCHAR(255) NOT NULL,

AGE INT,

CONSTRAINT PK\_EMPLOYEE PRIMARY KEY(ID, LASTNAME)

);

INSERT INTO EMPLOYEE VALUES(1,"MEE","YOU",20);

INSERT INTO EMPLOYEE VALUES(2,"MEE","YOU",20);

SELECT \* FROM EMPLOYEE;

--ADD PRIMARY KEY ON AN EXISTING TABLE:

--ALTER TABLE Employee ADD PRIMARY KEY(ID);