# SQL DAY 3

**Q.1 Create table Customers with schema (ID, name, age, address, salary)**
**Q.2 Create table Orders with Schema(O_ID, o_date, customer_id, amount)**

```
MySQL  localhost:33060+ ssl  training  SQL > Create table Customers( ID int NOT NULL AUTO_INCREMENT PRIMARY KEY, Name varchar(255) NOT NULL, Age int NOT NULL, Address
varchar(255) NOT NULL, Salary int NOT NULL);
Query OK, 0 rows affected (0.0501 sec)
MySQL  localhost:33060+ ssl  training  SQL > Create table Orders(O_ID int NOT NULL AUTO_INCREMENT PRIMARY KEY, O_date DATE NOT NULL, cust_ID int NOT NULL, Amount int N
OT NULL);
Query OK, 0 rows affected (0.0683 sec)
```

**Q.3 Insert 5 records to each table keeping few customer ids common to both the tables**

```
MySQL  localhost:33060+ ssl  training  SQL > insert into Orders(O_date, ID, Amount) values ("2022-07-08", 1, 2000), ("2022-07-09",2, 2500), ("2022-07-10", 3, 3700), ("
2022-07-11", 4, 2700), ("2022-07-12", 5, 4000);
ERROR: 1054: Unknown column 'ID' in 'field list'
MySQL  localhost:33060+ ssl  training  SQL > insert into Orders(O_date,Cust_ID, Amount) values ("2022-07-08", 1, 2000), ("2022-07-09",2, 2500), ("2022-07-10", 3, 3700)
, ("2022-07-11", 4, 2700), ("2022-07-12", 5, 4000);
Query OK, 5 rows affected (0.0112 sec)

Records: 5  Duplicates: 0  Warnings: 0
```

```
MySQL  localhost:33060+ ssl  training  SQL > Select * from Customers;
+----+----------+-----+----------+--------+
| ID | Name     | Age | Address  | Salary |
+----+----------+-----+----------+--------+
|  1 | Abhishek |  21 | Mumbai   |  70000 |
|  2 | Swaraj   |  22 | Pune     |  56000 |
|  3 | Anuj     |  23 | Delhi    |  60000 |
|  4 | Devyansh |  24 | Kanpur   |  60000 |
|  5 | Aditya   |  27 | Dehradun |  26000 |
+----+----------+-----+----------+--------+
5 rows in set (0.0009 sec)
MySQL  localhost:33060+ ssl  training  SQL > Select * from Orders;
+------+------------+---------+--------+
| O_ID | O_date     | cust_ID | Amount |
+------+------------+---------+--------+
|    1 | 2022-07-08 |       1 |   2000 |
|    2 | 2022-07-09 |       2 |   2500 |
|    3 | 2022-07-10 |       3 |   3700 |
|    4 | 2022-07-11 |       4 |   2700 |
|    5 | 2022-07-12 |       5 |   4000 |
+------+------------+---------+--------+
5 rows in set (0.0007 sec)
```

**Q.4 Perform the inner join on customers and orders table to enlist the id, name, amount and o date**

```
MySQL  localhost:33060+ ssl  training  SQL > Select ID, Name, Amount, O_date FROM Customers INNER JOIN Orders ON Customers.ID = Orders.Cust_ID;
+----+----------+--------+------------+
| ID | Name     | Amount | O_date     |
+----+----------+--------+------------+
|  1 | Abhishek |   2000 | 2022-07-08 |
|  2 | Swaraj   |   2500 | 2022-07-09 |
|  3 | Anuj     |   3700 | 2022-07-10 |
|  4 | Devyansh |   2700 | 2022-07-11 |
|  5 | Aditya   |   4000 | 2022-07-12 |
+----+----------+--------+------------+
5 rows in set (0.0007 sec)
```

**Q.5 Perform the left outer join on customers and orders table to enlist the id, name, amount and o date**

```
MySQL  localhost:33060+ ssl  training  SQL > Select ID, Name, Amount, O_date FROM Customers LEFT JOIN Orders ON Customers.ID = Orders.Cust_ID;
+----+---------+--------+------------+
| ID | Name    | Amount | O_date     |
+----+---------+--------+------------+
|  1 | Abhishek |   2000 | 2022-07-08 |
|  2 | Swaraj   |   2500 | 2022-07-09 |
|  3 | Anuj     |   3700 | 2022-07-10 |
|  4 | Devyansh |   2700 | 2022-07-11 |
|  5 | Aditya   |   4000 | 2022-07-12 |
+----+---------+--------+------------+
5 rows in set (0.0009 sec)
```

## Q.6 Perform the right outer join on customers and orders table to enlist the id, name, amount and o_date

```
MySQL  localhost:33060+ ssl  training  SQL > Select ID, Name, Amount, O_date FROM Customers RIGHT JOIN Orders ON Customers.ID = Orders.Cust_ID;
+----+---------+--------+------------+
| ID | Name    | Amount | O_date     |
+----+---------+--------+------------+
|  1 | Abhishek |   2000 | 2022-07-08 |
|  2 | Swaraj   |   2500 | 2022-07-09 |
|  3 | Anuj     |   3700 | 2022-07-10 |
|  4 | Devyansh |   2700 | 2022-07-11 |
|  5 | Aditya   |   4000 | 2022-07-12 |
+----+---------+--------+------------+
5 rows in set (0.0007 sec)
```

## Q.7 Perform the full outer join on customers and orders table to enlist the id, name, amount and o date by using 'union all set operation

```
MySQL  localhost:33060+ ssl  training  SQL > Select *
                                        -> FROM Customers
                                        -> LEFT JOIN Orders
                                        -> On Customers.ID = Orders.Cust_ID
                                        -> Union
                                        -> Select *
                                        -> FROM Customers
                                        -> RIGHT JOIN Orders
                                        -> ON Customers.ID = Orders.Cust_ID;
+----+---------+-----+---------+--------+------+------------+---------+--------+
| ID | Name    | Age | Address | Salary | O_ID | O_date     | cust_ID | Amount |
+----+---------+-----+---------+--------+------+------------+---------+--------+
|  1 | Abhishek |  21 | Mumbai  |  70000 |    1 | 2022-07-08 |       1 |   2000 |
|  2 | Swaraj   |  22 | Pune    |  56000 |    2 | 2022-07-09 |       2 |   2500 |
|  3 | Anuj     |  23 | Delhi   |  60000 |    3 | 2022-07-10 |       3 |   3700 |
|  4 | Devyansh |  24 | Kanpur  |  60000 |    4 | 2022-07-11 |       4 |   2700 |
|  5 | Aditya   |  27 | Dehradun |  26000 |    5 | 2022-07-12 |       5 |   4000 |
```

## Q.8 Perform the self join on customers table to enlist the pair of customers belonging to same Address

```
MySQL  localhost:33060+ ssl  training  SQL > Select * from Customers as T1 , Customers as T2 Where t1.address = t2.address and t1.id <> t2.id;
+----+---------+-----+---------+--------+----+---------+-----+---------+--------+
| ID | Name    | Age | Address | Salary | ID | Name    | Age | Address | Salary |
+----+---------+-----+---------+--------+----+---------+-----+---------+--------+
|  6 | Raj     |  23 | Mumbai  |  30000 |  1 | Abhishek |  21 | Mumbai  |  70000 |
|  1 | Abhishek |  21 | Mumbai  |  70000 |  6 | Raj     |  23 | Mumbai  |  30000 |
+----+---------+-----+---------+--------+----+---------+-----+---------+--------+
2 rows in set (0.0010 sec)
```

## Q.9 Perform the Cross/ Cartesian join on customers and orders table to enlist the id, name, amount and o_date

```
MySQL  localhost:33060+ ssl  training  SQL > SELECT ID, Name, Amount, O_date from
                                          -> Customers , Orders;
+----+----------+--------+------------+
| ID | Name     | Amount | O_date     |
+----+----------+--------+------------+
|  1 | Abhishek |   4000 | 2022-07-12 |
|  1 | Abhishek |   2700 | 2022-07-11 |
|  1 | Abhishek |   3700 | 2022-07-10 |
|  1 | Abhishek |   2500 | 2022-07-09 |
|  1 | Abhishek |   2000 | 2022-07-08 |
|  2 | Swaraj   |   4000 | 2022-07-12 |
|  2 | Swaraj   |   2700 | 2022-07-11 |
|  2 | Swaraj   |   3700 | 2022-07-10 |
|  2 | Swaraj   |   2500 | 2022-07-09 |
|  2 | Swaraj   |   2000 | 2022-07-08 |
|  3 | Anuj     |   4000 | 2022-07-12 |
|  3 | Anuj     |   2700 | 2022-07-11 |
|  3 | Anuj     |   3700 | 2022-07-10 |
|  3 | Anuj     |   2500 | 2022-07-09 |
|  3 | Anuj     |   2000 | 2022-07-08 |
|  4 | Devyansh |   4000 | 2022-07-12 |
|  4 | Devyansh |   2700 | 2022-07-11 |
|  4 | Devyansh |   3700 | 2022-07-10 |
|  4 | Devyansh |   2500 | 2022-07-09 |
|  4 | Devyansh |   2000 | 2022-07-08 |
|  5 | Aditya   |   4000 | 2022-07-12 |
|  5 | Aditya   |   2700 | 2022-07-11 |
|  5 | Aditya   |   3700 | 2022-07-10 |
|  5 | Aditya   |   2500 | 2022-07-09 |
|  5 | Aditya   |   2000 | 2022-07-08 |
|  6 | Raj      |   4000 | 2022-07-12 |
|  6 | Raj      |   2700 | 2022-07-11 |
|  6 | Raj      |   3700 | 2022-07-10 |
|  6 | Raj      |   2500 | 2022-07-09 |
|  6 | Raj      |   2000 | 2022-07-08 |
+----+----------+--------+------------+
```

## Q.10 Design the sub query with select statement for displaying all the details of the customers having salary greater than 20000

```
MySQL  localhost:33060+ ssl  training  SQL > Select * from Customers
                                          -> Where Customers.Salary>20000;
+----+----------+-----+----------+--------+
| ID | Name     | Age | Address  | Salary |
+----+----------+-----+----------+--------+
|  1 | Abhishek |  21 | Mumbai   |  70000 |
|  2 | Swaraj   |  22 | Pune     |  56000 |
|  3 | Anuj     |  23 | Delhi    |  60000 |
|  4 | Devyansh |  24 | Kanpur   |  60000 |
|  5 | Aditya   |  27 | Dehradun |  26000 |
|  6 | Raj      |  23 | Mumbai   |  30000 |
+----+----------+-----+----------+--------+
6 rows in set (0.0009 sec)
```

## Q.11 Create a backup table- cust_bkp' of the table customers by using insert statement with the subquery

```
MySQL localhost:33060+ ssl training SQL > CREATE TABLE cust_bkp AS SELECT * from Customers;\
Query OK, 6 rows affected (0.0630 sec)

Records: 6  Duplicates: 0  Warnings: 0
MySQL localhost:33060+ ssl training SQL > CREATE TABLE cust_bkp AS SELECT * from Customers;
ERROR: 1050: Table 'cust_bkp' already exists
MySQL localhost:33060+ ssl training SQL > Insert into cust_bkp(Name, Age, Address, Salary) values ("Raj" , 23, "Mumbai", 30000);
Query OK, 1 row affected (0.0099 sec)
MySQL localhost:33060+ ssl training SQL > Select * from cust_bkp;
+----+----------+-----+----------+--------+
| ID | Name     | Age | Address  | Salary |
+----+----------+-----+----------+--------+
|  1 | Abhishek |  21 | Mumbai   |  70000 |
|  2 | Swaraj   |  22 | Pune     |  56000 |
|  3 | Anuj     |  23 | Delhi    |  60000 |
|  4 | Devyansh |  24 | Kanpur   |  60000 |
|  5 | Aditya   |  27 | Dehradun |  26000 |
|  6 | Raj      |  23 | Mumbai   |  30000 |
|  0 | Raj      |  23 | Mumbai   |  30000 |
+----+----------+-----+----------+--------+
7 rows in set (0.0006 sec)
```

## Q.12 Update the salaries by 10% of all the customers(in customers table) having age greater than or equals to 24 by using subquery with update clause( by using backup table cust_bkp)

```
MySQL localhost:33060+ ssl training SQL > Update Cust_bkp SET Salary = Salary +(Salary*10/100) WHERE Age >= 24;
Query OK, 2 rows affected (0.0105 sec)

Rows matched: 2  Changed: 2  Warnings: 0
```

## Q.13 Delete all the customers having age greater than 26 by using delete clause with the subquery

```
MySQL localhost:33060+ ssl training SQL > DELETE FROM Customers WHERE Age > 26;
Query OK, 1 row affected (0.0105 sec)
MySQL localhost:33060+ ssl training SQL > Select * from Customers;
+----+----------+-----+----------+--------+
| ID | Name     | Age | Address  | Salary |
+----+----------+-----+----------+--------+
|  1 | Abhishek |  21 | Mumbai   |  70000 |
|  2 | Swaraj   |  22 | Pune     |  56000 |
|  3 | Anuj     |  23 | Delhi    |  60000 |
|  4 | Devyansh |  24 | Kanpur   |  66000 |
|  6 | Raj      |  23 | Mumbai   |  30000 |
+----+----------+-----+----------+--------+
```