MYSQL CLAUSES

1) MySQL WHERE Clause

MySQL WHERE Clause is used with SELECT, INSERT, UPDATE and DELETE clause to filter the results. It specifies a specific position where you have to do the operation.

Let's take an example to retrieve data from a table "officers".

```
mysql> SELECT*FROM officers;
+------------+--------------+-----------+
| officer_id | officer_name | address   |
+------------+--------------+-----------+
|          1 | Ajeet        | Mau       |
|          2 | Deepika      | Lucknow   |
|          3 | Vimal        | Faizabad  |
|          4 | Rahul        | Lucknow   |
+------------+--------------+-----------+
4 rows in set (0.00 sec)
```

- **MySQL WHERE Clause with single condition**

  Example: mysql> SELECT * FROM officers WHERE address = 'Mau';

  Output:

```
mysql> SELECT *
    -> FROM officers
    -> WHERE address = 'Mau';
+------------+--------------+---------+
| officer_id | officer_name | address |
+------------+--------------+---------+
|          1 | Ajeet        | Mau     |
+------------+--------------+---------+
1 row in set (0.00 sec)
```

- **MySQL WHERE Clause with AND condition**

  In this example, we are retrieving data from the table "officers" with AND condition.

  Example:

  SELECT * FROM officers WHERE address = 'Lucknow' AND officer_id < 5;

  Output:

```
mysql> SELECT *
    -> FROM officers
    -> WHERE address = 'Lucknow'
    -> AND officer_id < 5;
+------------+--------------+---------+
| officer_id | officer_name | address |
+------------+--------------+---------+
|          2 | Deepika      | Lucknow |
|          4 | Rahul        | Lucknow |
+------------+--------------+---------+
2 rows in set (0.06 sec)
```

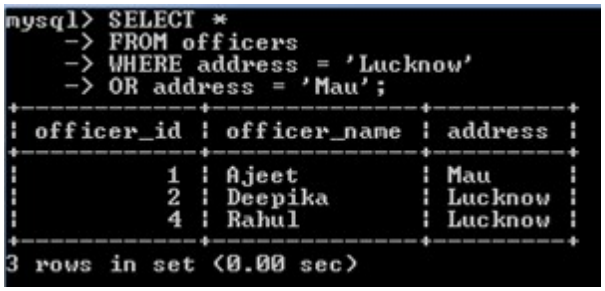
- **MySQL WHERE Clause with OR condition**

  In this example, we are retrieving data from the table "officers" with OR condition.

  Example:

  SELECT * FROM officers WHERE address = 'Lucknow' OR address = 'Mau';

  Output:



- **MySQL WHERE Clause with combination of AND & OR conditions**

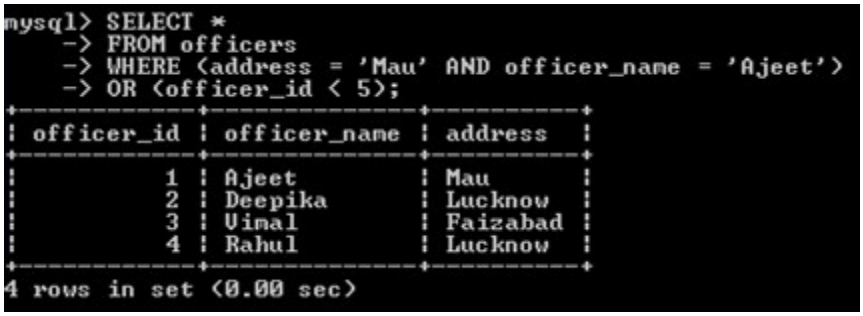
  You can also use the AND & OR conditions altogether with the WHERE clause.

  Example:

  SELECT * FROM officers WHERE (address = 'Mau' AND officer_name = 'Ajeet')

  OR (officer_id < 5);

  Output:



2) MySQL DISTINCT Clause

MySQL DISTINCT clause is used to remove duplicate records from the table and fetch only the unique records. The DISTINCT clause is only used with the SELECT statement.

Syntax:

```
SELECT DISTINCT expressions
FROM tables
[WHERE conditions];
```

- **MySQL DISTINCT Clause with single expression**

  If you use a single expression then the MySQL DISTINCT clause will return a single field with unique records (no duplicate record).

  Example:

  ```
  SELECT DISTINCT address  FROM officers;
  ```

  Output:

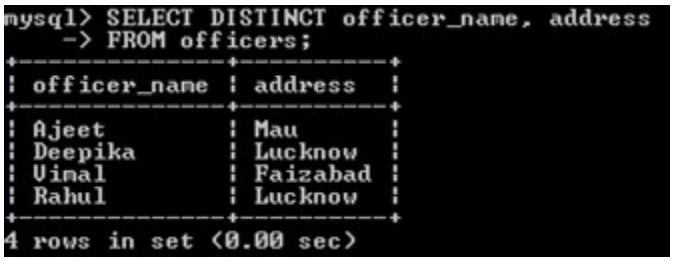
- **MySQL DISTINCT Clause with multiple expressions**

  If you use multiple expressions with DISTINCT Clause then MySQL DISTINCT clause will remove duplicates from more than one field in your SELECT statement.

  Example:

  ```
  SELECT DISTINCT officer_name, address  FROM officers;
  ```

  Output:



3) MySQL ORDER BY Clause

The MYSQL ORDER BY Clause is used to sort the records in ascending or descending order.

Syntax:

```
SELECT expressions FROM tables
[WHERE conditions]  ORDER BY expression [ ASC | DESC ];
```

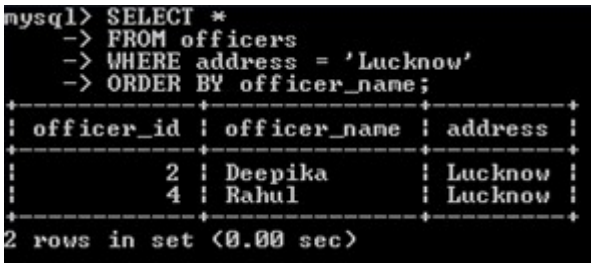
- **MySQL ORDER BY: without using ASC/DESC attribute**

  If you use MySQL ORDER BY clause without specifying the ASC and DESC modifier then by default you will get the result in ascending order.

  Example:

  ```
  SELECT *   FROM officers   WHERE address = 'Lucknow'  ORDER BY officer_name;
  ```

  Output:



```
mysql> SELECT *
    -> FROM officers
    -> WHERE address = 'Lucknow'
    -> ORDER BY officer_name;
+------------+--------------+---------+
| officer_id | officer_name | address |
+------------+--------------+---------+
|          2 | Deepika      | Lucknow |
|          4 | Rahul        | Lucknow |
+------------+--------------+---------+
2 rows in set (0.00 sec)
```

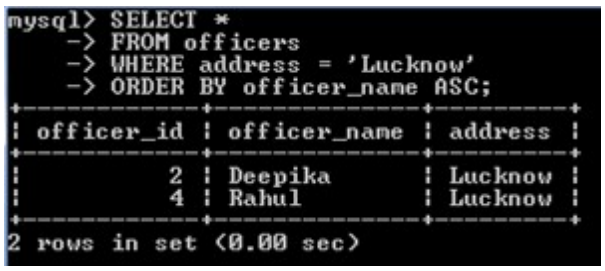
- **MySQL ORDER BY: with ASC attribute**

  Let's take an example to retrieve the data in ascending order.

  Example:

  ```
  SELECT *   FROM officers   WHERE address = 'Lucknow'
  ORDER BY officer_name ASC;
  ```

  Output:



```
mysql> SELECT *
    -> FROM officers
    -> WHERE address = 'Lucknow'
    -> ORDER BY officer_name ASC;
+------------+--------------+---------+
| officer_id | officer_name | address |
+------------+--------------+---------+
|          2 | Deepika      | Lucknow |
|          4 | Rahul        | Lucknow |
+------------+--------------+---------+
2 rows in set (0.00 sec)
```

- **MySQL ORDER BY: with DESC attribute**

  Let's take an example to retrieve the data in descending order.

  Example:

  ```
  SELECT *   FROM officers
  ```

WHERE address = 'Lucknow' ORDER BY officer_name DESC;

Output: