

1) ORDER BY

SELECT * FROM tablename

ORDER BY column-1, column-2... ASC | DESC; (Default ASC)

```
SELECT * FROM customers
```

```
ORDER BY country, customer_name;
```

→ Select all records from customers and sort them by country, if they are from the same country then sort by customer name.

```
SELECT * FROM customers
```

```
ORDER BY country ASC, customer_name DESC;
```

→ Select all records from customers and sort them by country in ascending order, if they are from the same country then sort by customer name in descending order.

Example:

- ```
SELECT * FROM `ccc_product` ORDER BY product_created_at;
```

## 2) LIMIT

SELECT \* FROM tablename

LIMIT number;

```
SELECT * FROM customers
```

```
LIMIT 3;
```

→ Select first three records from customers.

```
SELECT * FROM customers
```

```
LIMIT 3 OFFSET 3;
```

→ Select three records starting from row 4 customers.

```
SELECT * FROM customers
```

```
WHERE country = 'India'
```

```
LIMIT 3;
```

→ Select first three records from customers where the country is India.

Example:

- ```
SELECT * FROM `ccc_product` LIMIT 10;
```

- ```
SELECT * FROM `ccc_product` LIMIT 10 OFFSET 9;
```

### 3) WHERE

SELECT \* FROM tablename WHERE condition;

```
SELECT * FROM customers
WHERE country = 'India';
```

→ Select all records from customers where the country is India.

```
SELECT * FROM customers
WHERE customer_id = 1;
```

→ Select records from customers where the customer id is 1.

Operators Used:

- = Equal
- > Greater than
- < Less than
- >= Greater than or equal
- <= Less than or equal
- <> or != Not equal
- BETWEEN (between certain range)
- LIKE (search for a pattern)
- IN (to specify multiple value for a column)

Example:

- SELECT \* FROM `ccc\_product` WHERE product\_type = 'simple';
- SELECT \* FROM `ccc\_category` WHERE `cat\_name` = 'Jeremy';

### 4) JOIN

4 Types:

- Inner Join
  - SELECT \* FROM tablename\_1  
INNER JOIN tablename\_2  
ON tablename\_1.foreign\_key = tablename\_2.foreign\_key;
  - SELECT \* FROM (  
tablename\_1  
INNER JOIN tablename\_2  
ON tablename\_1.foreign\_key = tablename\_2.foreign\_key)  
INNER JOIN tablename\_3  
ON tablename\_1.foreign\_key = tablename\_3.foreign\_key);
- Left Join
- Right Join
- Cross Join

Example:

- ```
SELECT ccc_product.`product_name`, ccc_category.`cat_name`  
FROM ccc_product  
INNER JOIN ccc_category  
ON ccc_product.`cat_id`=ccc_category.`cat_id`;
```

5) GROUP BY

SELECT * FROM tablename

WHERE condition

GROUP BY column_names;

```
SELECT COUNT(customer_id), country FROM customers  
GROUP BY country;
```

→ List the number of records in each country.

```
SELECT shippers.shipper_name, COUNT(orders.order_id) FROM orders  
INNER JOIN shippers  
ON orders.shipper_id = shippers.shipper_id  
GROUP BY shipper_name;
```

→ List the number of orders shipped by each shipper.

Example:

- ```
SELECT COUNT(ccc_product.`product_name`) AS product_count, ccc_category.`cat_name`
FROM ccc_product
INNER JOIN ccc_category
ON ccc_product.`cat_id`=ccc_category.`cat_id`
GROUP BY cat_name;
```
- ```
SELECT COUNT(product_name), product_sku FROM ccc_product  
GROUP BY product_status;
```

6) UPDATE with JOIN

UPDATE table1

INNER JOIN table2

ON table1.join_column = table2.join_column

SET table1.column = new_value, table2.column = new_value;

```
UPDATE customers  
INNER JOIN orders  
ON customers.id = orders.customer_id  
SET customers.salary = customers.salary + 1000, orders.amount = orders.amount + 500;
```

→ Increase the salary of customers by 1000 and order amount by 500.

```
UPDATE customers
INNER JOIN orders
ON customers.id = orders.customer_id
SET customers.salary = customers.salary + 1000, orders.amount = orders.amount + 500
WHERE orders.customer_id = 3;
```

➡ Increase the salary of customers by 1000 and order amount by 500 where customer_id is 3 in the orders table.

Example:

- UPDATE ccc_product
INNER JOIN ccc_category
ON ccc_product.`cat_id`=ccc_category.`cat_id`
SET ccc_product.product_name = 'Updated', ccc_category.cat_name = 'Updated Cate'
WHERE product_id = 1;

7) INSERT on Duplicate

Example:

- INSERT INTO ccc_category (cat_id, cat_name)
VALUES (1, 'Electronics')
ON DUPLICATE KEY UPDATE cat_id = 1;

➡ This query inserts a new record into the ccc_category table with category_id 1 and category_name 'Electronics'. If a record with category_id 1 already exists, it updates the category_name to 'Electronics' instead of throwing a duplicate key error.