
SELECT

BASIC SELECT STATEMENT

- A basic and a very simple select statement shall include the columns that you would like to see and from which table.
- You need to mention two things mandate in a select statement.
  - From which table you would like to see the data.
  - And do you want to see all the columns or you would like to see only limited columns as per your requirement.

•While writing a select statement couple of things that you need to keep in mind:

- Select Statements will not make any changes to the database. It is just a query language and it will just help you to retrieve the data from the database depending on your requirement.
- SQL statements are not case- sensitive so you can write in any case you want.
- SQL statements are not sensitive to spaces so you can write your query in one line or for readability purpose in multiple lines it will not make any difference.
- The only thing that we need to keep in mind is that the keyword like SELECT or other keywords in SQL you cannot split the keywords.
- Every SQL statement needs to end with a semicolon(;)

- Consider the Products Table on the right which has three columns and six rows.

product id	name	PRICE
a	tv	500
b	phone	600
c	ac	700
d	refrigrator	900
e	LED	1100
f	Micorwave	1200

- Let's suppose I want to see the complete data on my screen so I will write the below query.

```
SELECT * from Products;
```

- Now, in another scenario I would like to see only two columns i.e. product_id and name but still want to see all the rows, so I will write the below query

```
SELECT product_id, name from Products;
```

SELECT WITH WHERE CLAUSE

- WHERE Clause is used to restrict the number of rows in the report as only those rows will be shown which matches the condition.
- Consider the Products Table on the right which has three columns and six rows.

product id	name	PRICE
a	tv	500
b	phone	600
c	ac	700
d	refrigrator	900
e	LED	1100
f	Micorwave	1200

- Let's suppose I want to see the data only for those Products where the price of the products is more than 1000, so I will write the below query

```
SELECT * from products
where price > 1000;
```

- Let's suppose I want to see the data only for those Products where the price of the products is more than 1000 and less than 1200, so I will write the below query

```
SELECT * from products
where price > 1000 and price < 1200;
```

NOTE: Whenever you write multiple conditions you need to make use of AND | Or operator.

AND : Both the conditions needs to be satisfied.

OR : Any one condition satisfied it will do the job.

SELECT WITH BETWEEN OPERATOR

- Whenever you are giving a condition on one numerical column where you would like to find the data between a range can make use of BETWEEN operator.
- Consider the Products Table on the right which has three columns and six rows.

product id	name	PRICE
a	tv	500
b	phone	600
c	ac	700
d	refrigrator	900
e	LED	1100
f	Micorwave	1200

- Let's suppose I want to see the data only for those Products where the price of the products is more than 1000 and less than 1200, so I will write the below query

```
SELECT * from products
where price between  1000 and 1500;
```

NOTE: Between Operator the upper and the lower limit is included.

1200

SELECT WITH LIMIT CLAUSE

- Whenever you want to LIMIT the number of rows you want to see in the report or you want to extract from database you can make use of LIMIT statement.
- The LIMIT statement will just limit the rows in your output it will not restrict the rows based on any condition.
  - Let's suppose I want to see only first three rows, so I will write the below query

```
SELECT * from products
LIMIT 3;
```

The above query will just show three rows from the top in the output.

product id	name	PRICE
a	tv	500
b	phone	600
c	ac	700
d	refrigrator	900
e	LED	1100
f	Micorwave	1200

SELECT WITH ORDER BY CLAUSE

- Whenever you want to arrange the data in ascending or descending order we make use of the order by clause
- The default property of order by clause is that it sorts the data in ascending order.

product id	name	PRICE
a	tv	500
b	phone	600
c	ac	700
d	refrigrator	900
e	LED	1100
f	Micorwave	1200

- ASCENDING ORDER query

```
SELECT * from products
order by price;  OFFSET
```

- DESCENDING ORDER query

```
SELECT * from products
                                order by price DESC
Limit 3 offset 2;
```

SELECT WITH IS NULL AND IS NOT NULL

- Whenever you want to arrange extract the data based on null values in a column or non-null values in a column you can make use of IS NULL or ISNULL Operator..
  - Extracting rows where price is NULL

```
SELECT * from products
 where price is null;
```

(Will get last three rows only)

  - Extracting rows where price is NOT NULL

```
SELECT * from products
 where price is not null;
```

(Will get first three rows only)

product id	name	PRICE
a	tv	500
b	phone	600
c	ac	700
d	refrigrator	null
e	LED	null
f	Micorwave	null

SELECT WITH DISTINCT CLAUSE

- Whenever you want to extract only unique rows or data from a column you can make use of DISTINCT clause.
  - Extracting unique rows from product id.

    SELECT distinct product_id from products

    (Will get first six rows only)

NOTE: Distinct keyword will always be next to SELECT keyword.

product id	name	PRICE
a	tv	500
b	phone	600
c	ac	700
d	refrigrator	null
e	LED	null
f	Micorwave	null
f	Micorwave	null

SELECT WITH LIKE OPERATOR (WILDCARD SEARCH)

•Whenever we want to extract data based on a pattern we make use of LIKE operator which is also known was WILDCARD SEARCH.

product id	name	PRICE
a	tv	500
b	phone	600
c	ac	700
d	refrigrator	null
e	LED	null
f	Micorwave	null
f	Micorwave	null

- The LIKE operator is used with WHERE clause to search for a specified pattern in a column.
- Wildcard search makes use of symbol

Symbol	Description	Example
%	Represents zero or more characters	bl% finds bl, black, blue, and blob
_	Represents a single character	h_t finds hot, hat, and hit

- Some examples of WILDCARD search is mentioned below in the screenshot.

LIKE Operator	Description
WHERE CustomerName LIKE 'a%'	Finds any values that starts with "a"
WHERE CustomerName LIKE '%a'	Finds any values that ends with "a"
WHERE CustomerName LIKE '%or%'	Finds any values that have "or" in any position
WHERE CustomerName LIKE '_r%'	Finds any values that have "r" in the second position
WHERE CustomerName LIKE 'a_%_%'	Finds any values that starts with "a" and are at least 3 characters in length
WHERE ContactName LIKE 'a%o'	Finds any values that starts with "a" and ends with "o"

SELECT WITH IN AND NOT IN

- Whenever you want to extract a list of data or everything else apart from a particular list we make use of IN and NOT IN operator.
  - Want to see only products like tv, ac and refrigerator

```
SELECT * from products
 where name in ('tv','ac','refrigerator');
```

  - Extracting every other product but not tv, ac and refrigerator

```
SELECT * from products
where name not in ('tv','ac','refrigerator');
```

product id	name	PRICE
a	tv	500
b	phone	600
c	ac	700
d	refrigrator	null
e	LED	null
f	Micorwave	null