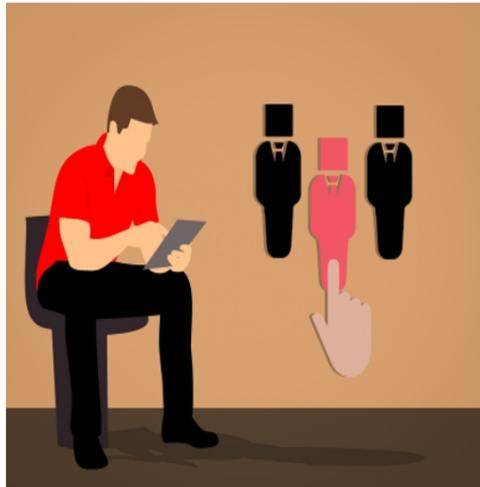


## SELECT Statement

- The **SELECT** statement of SQL is used to retrieve data from one or more tables or views.
- The result returned is stored in a temporary result table, called the result-set.

### **SELECT can be used for retrieving:**

- all rows from a table.
- specific rows from a table.
- specific values from a table.



## SELECT syntax

```

SELECT [distinct] [column_name1,column_name2...] | *
FROM table_name [alias] [,table_name [alias]..]
[WHERE conditions]
[GROUP BY group [HAVING group_conditions]]
[ORDER BY sort_columns[ASC|DESC]];

```

We will discuss more regarding SELECT statement in this session.

## SELECT

SQL statements are **not case sensitive**, but the data stored in the database is **case sensitive**. For example: a member with the name **tom** and **TOM** are different, but **SELECT** and **select** are the same statement.

**Remember!**

## Retrieve all the records



Query:

SELECT \* FROM Customer;

Output:

CID	CNAME	PHONENO	DOB	EMAILID	ADDRESS
1	Tom	9876523190	17-MAR-87	tom@gmail.com	chennai
2	John	8765432190	26-JAN-86	john@yahoo.com	delhi
3	Ram	7654321890	14-DEC-85	ram@gmail.com	pune
4	Tiny	9012365478	28-MAY-86	NULL	chennai

## Retrieve Specific Columns



Query:

SELECT cname, address FROM Customer;

Output:

CNAME	ADDRESS
Tom	chennai
John	delhi
Ram	pune
Tiny	chennai

## Arithmetic Expressions



I want to know the amount of the policy including it's GST of 10%.

You can perform this operation using Arithmetic Operator in SELECT clause.

PID	PNAME	PPERIODINYEARS	MINAMOUNTPERMONTH
MBP	Money Back Plan	20	1000
PP	Personal Protect	15	1500

SELECT pid, Minamountpermonth+(Minamountpermonth\*0.1) FROM policy;

PID	MINAMOUNTPERMONTH +(MINAMOUNTPERMONTH*0.1)
MBP	1100
PP	1650



## Column Alias

Oh! Why is the column name meaningless in the previous query?

Provide an alias name for the column.

```
SELECT pid, Minamountpermonth+(Minamountpermonth*0.1) PolicyAmount FROM policy;
```

PID	POLICYAMOUNT
MBP	1100
PP	1650

Arithmetic expressions containing a Null Value evaluates to Null.

Enclose the alias name in double quotes, if it contains space between words.

## Concatenation Operator

How to display the customer name along with their address in the following format  
"CustomerName" lives in "Address"?

|| operator is used to concatenate columns or character strings.

```
SELECT Cname || ' lives in' || Address as CustomerAddress FROM customer;
```

CUSTOMERADDRESS
Tom lives in Chennai
John lives in Delhi
Ram lives in Pune
Tiny lives in Chennai

## Eliminating Duplicate Rows

I want to display the address of all the customers, but I don't want to duplicate them.

DISTINCT keyword in the SELECT clause is used to eliminate duplicate rows .

```
SELECT DISTINCT address FROM customer;
```

ADDRESS
Chennai
Delhi
Pune

## WHERE CLAUSE



Restrict the rows returned by using the WHERE clause. The '**WHERE**' clause follows the '**FROM**' clause.

```
SELECT * | {[DISTINCT] column|expression [alias],...}
```

```
FROM table {[WHERE condition(s)];}
```



Value in the condition is compared using **Relational operators**.



**Logical operators** are used to specify more than one condition in WHERE clause.

## Where CLAUSE

What is the query to retrieve customers who are only from Chennai?

Use 'WHERE clause', which is used to extract only those records that fulfill a specific condition.

```
SELECT cname, emailid, address FROM customer WHERE address='Chennai';
```

CNAME	EMAILID	ADDRESS
Tom	tom@gmail.com	Chennai
Tiny	tiny@yahoo.co.in	Chennai

## Comparison Operators

Operator	Explanation
=	Equal to.
>	Greater than.
>=	Greater than or equal to.
<	Less than.
<=	Less than or equal to.
!=, <>	Not equal to.
Between ... and	Range of values.
In(set)	Match any of the list values.
Like	Match a character pattern.
Isnull	Is a null value.



## Example - Less than operator



What is the query to retrieve all the policies whose Minimum Amount to be paid is less than 1500?

In the same manner, we can use  
>,>,<=

```
SELECT * FROM Policy WHERE MinAmountPerMonth<1500;
```

PID	PNAME	PPERIODINYEARS	MINAMOUNTPERMONTH
MBP	Money Back Plan	20	1000

## Example - Like operator



What is the query to retrieve the customers who have a yahoo email?

To select rows that match a character pattern by using the LIKE condition. Search conditions can contain either literal characters or numbers:  
'%' denotes zero or many characters.  
'\_' denotes one character.

```
SELECT cid, cname, emailid FROM customer WHERE emailid LIKE '%yahoo%';
```

CID	CNAME	EMAILID
2	John	john@yahoo.com
4	Tiny	tiny@yahoo.co.in

## WHERE CLAUSE



OPERATORS	MEANING	EXAMPLE	DESCRIPTION
Between and Y	X	Range of values between X and Y (X,Y is inclusive ) .....WHERE Salary Between 5000 and 8000'	List the employees who earn salary between 5000 to 8000 i.e 5000,5001,.....7999,8000
IN (set)		Match any of the list values .....WHERE Salary IN (5000,6000,7000)	List the employees who earn salary of 5000 or 6000 or 7000 exactly
IS NULL		.....WHERE Salary IS NULL	List the employees whose salary is empty

A null is not the same as zero or a space. Zero is a number, and a space is a character. If any operation has NULL value, then the result is null.

## Example – Between, In, Is Null



To retrieve policy details where minimum amount to be paid is between 2500 and 5000.

```
SELECT * FROM policy WHERE MinAmountPerMonth BETWEEN 2500 AND 5000;
```

To retrieve customer id and policy id of customers whose customer id is 101,105

```
SELECT cid, pid FROM policyenrollment WHERE cid IN (101,105);
```

To retrieve policy enrollment details of those who have not paid penalty for late payment

```
SELECT * FROM policyenrollment WHERE penalty IS NULL;
```

## Logical operators



OPERATOR	MEANING
AND	Returns true if both the conditions are true.
OR	Returns true if either of the condition is true.
NOT	Returns true if the condition is false.

```
SELECT cid, cname FROM customer WHERE cid=1 AND address='chennai';
```



'NOT' can be used with  
BETWEEN, IN, LIKE and  
IS NULL operators.

## Order by clause



To retrieve names of the customers in ascending order.

```
SELECT Cname FROM customer ORDER BY cname;
```

CNAME
John
Mini
Ram
Tiny
Tom

Default -  
ascending  
order.

For descending order:  
Select cname from  
customer **order by**  
**cname desc;**