Ex.No. : 03
Date : **Where and Aggregate Functions**

Aim:

To study and execute various database queries using where clause and aggregate functions.

Description:

The WHERE clause is used to filter records. It is used to extract only those records that fulfill a specified condition.

Where clause's

Operator	**Description**
=	Equal
>	Greater than
<	Less than
>=	Greater than or equal
<=	Less than or equal
<>	Not equal.(Note: In some versions of SQL this operator may be written as !=)
BETWEEN	Between a certain range
LIKE	Search for a pattern
IN	To specify multiple possible values for a column

The WHERE clause is not only used in SELECT statements, it is also used in UPDATE, DELETE, etc.!

Equal

SQL> select * from table_name where field=condition
Example
SQL> select * from student1 where stud_rollno=101;

STUD_NAME	STUD_ID	STUD_DEPT	STUD_ROLLNO
Saddiq	103	CSE	101

Greater Than

SQL> select * from student1 where stud_rollno >101;

STUD_NAME	STUD_ID	STUD_DEPT	STUD_ROLLNO
Ram	101	MECH	104
Vicky	102	EEE	105
David	104	EEE	103

Less Than

SQL> select * from student1 where stud_rollno <105;

STUD_NAME	STUD_ID	STUD_DEPT	STUD_ROLLNO
Ram	101	MECH	104

Saddiq	103	CSE	101
David	104	EEE	103

Between

SQL> select * from student1 where stud_rollno between 103 AND 105;

STUD_NAME	STUD_ID	STUD_DEPT	STUD_ROLLNO
Ram	101	MECH	104
Vicky	102	EEE	105
David	104	EEE	103

Like

Syntax

SELECT *column1, column2, ...*
FROM *table_name*
WHERE *columnN* LIKE *pattern*;

LIKE Operator	**Description**
WHERE CustomerName LIKE 'a%'	Finds any values that start with "a"
WHERE CustomerName LIKE '%a'	Finds any values that end 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 start with "a" and are at least 2 characters in length
WHERE CustomerName LIKE 'a__%'	Finds any values that start with "a" and are at least 3 characters in length
WHERE ContactName LIKE 'a%o'	Finds any values that start with "a" and ends with "o"

Example

SQL > Select * from student1 where stud_name like ‘d%’;

STUD_NAME	STUD_ID	STUD_DEPT	STUD_ROLLNO
David	104	EEE	103

Other Where Clauses

Union:

The UNION operator is used to combine the result-set of two or ore SELECT statements.

- Every SELECT statement within UNION must have the same number of columns
- The columns must also have similar data types
- The columns in every SELECT statement must also be in the same order

Syntax:

select column_name(s) from table1
union
select column_name(s) from table2*;*

SQL> select subject from student union select subject from staff order by subject;

Union with where
Syntax

Select <fieldlist> from <tablename1> where (condition)
union
select<fieldlist> from<tablename2> where (condition);

Example
SQL> SELECT dept, subject FROM student
WHERE subject='DBMS'
UNION
SELECT dept, subject FROM staff
WHERE subject='DBMS'
ORDER BY City;

Union All
select column_name(s) from table1
union all
select column_name(s) from table2;

Example
select subject from student
union all
select subject from staff
order by subject;

Intersect:

Syntax:
Select <fieldlist> from <tablename1> where (condition) intersect select<fieldlist>

from<tablename2> where (condition);

Example

SQL> select stud_id from student intersect select staff_id from staff;

In:
The `IN` operator allows you to specify multiple values in a `WHERE` clause.
The `IN` operator is a shorthand for multiple `OR` conditions.

Syntax:
Select <fieldlist> from <tablename1> where (condition) in select<fieldlist> from<tablename2>

where (condition);

Example
SQL>select * from student1 where stud_dept in ('cse', 'mech');

STUD_NAME	STUD_ID	STUD_DEPT	STUD_ROLLNO
Ram	101	MECH	104
Saddiq	103	CSE	101

SQL>select * from customers where stud_dept not in ('cse', 'mech', 'it');

STUD_NAME	STUD_ID	STUD_DEPT	STUD_ROLLNO
Vicky	102	EEE	105
David	104	EEE	103

Not like:

Syntax:

Select <fieldlist> from <tablename> where <fieldname> not like <expression>;

All:

The ALL operator:

- returns a boolean value as a result
- returns TRUE if ALL of the subquery values meet the condition
- is used with SELECT, WHERE and HAVING statements

ALL means that the condition will be true only if the operation is true for all values in the range.

Syntax:

Select <fieldlist> from <tablename1>where <fieldname> all Select <fieldlist> from

<tablename2> where (condition);

Example
SQL> select stud_name from student where stud_id = all (select subject_id from subject where sem = 4);

Any:
The ANY operator:

- returns a Boolean value as a result
- returns TRUE if ANY of the sub query values meet the condition

ANY means that the condition will be true if the operation is true for any of the values in the range.

Syntax:

```
SELECT column_name(s)
FROM table_name
WHERE column_name operator ANY
  (SELECT column_name
  FROM table_name
  WHERE condition);
```

The operator must be a standard comparison operator (=, <>, !=, >, >=, <, or <=).

Example

SQL> select stud_name from student where stud_id = any (select subject_id from subject where sem = 4);

The above SQL statement lists the student name if it finds ANY records in the subject table has sem equal to 4 (this will return TRUE if the sem column having value 4)

Aggregate Functions

MySQL's aggregate function is used to perform calculations on multiple values and return the result in a single value like the average of all values, the sum of all values, and maximum & minimum value among certain groups of values. We mostly use the aggregate functions with SELECT statements in the data query languages.

Aggregate Function	Descriptions
count()	It returns the number of rows, including rows with NULL values in a group.
sum()	It returns the total summed values (Non-NULL) in a set.
average()	It returns the average value of an expression.
min()	It returns the minimum (lowest) value in a set.
max()	It returns the maximum (highest) value in a set.
groutp_concat()	It returns a concatenated string.
first()	It returns the first value of an expression.
last()	It returns the last value of an expression.

```
Sql > create table student(rollno decimal, sname varchar(15), mark1 decimal, mark2 decimal);
Table created
Sql> insert into student values(101, 'kareem',95,90);
Sql> insert into student values(102, 'kaasim',92,97);
Sql> insert into student values(103, 'ram',85,95);
Sql> insert into student values(104, 'sai',93,91);
```

```
Sql> select * from student;
```

Rollno	sname	mark1	mark2
101	kareem	95	90
102	kaasim	92	97

103	ram	85	95
104	sai	93	91

Count

The COUNT () function returns the number of rows that matches a specified criterion.

Syntax

SELECT COUNT (column_name) FROM table_name WHERE condition;

Example

SELECT COUNT (rollno) FROM student;

4

Sum

The SUM () function returns the total sum of a numeric column.

Syntax

SELECT SUM(column_name) FROM table_name WHERE condition;

Example

SELECT SUM(mark1) FROM student;

365

SELECT SUM(mark1) FROM student WHERE mark2>=95;

192

Average

The AVG() function returns the average value of a numeric column.

Syntax

SELECT AVG(column_name) FROM table_name WHERE condition;

Example

SELECT AVG(mark1) FROM student;

91

SELECT AVG(mark2) FROM student WHERE mark2>=95;

96

Min

The MIN() function returns the smallest value of the selected column.

SELECT MIN(column_name) FROM table_name WHERE condition;

Example

SELECT MIN(mark1) FROM student;

85

Max

The MAX() function returns the largest value of the selected column.

SELECT MAX(column_name) FROM table_name WHERE condition;
Example
SELECT MAX(mark1) FROM student;
95

Result

Thus the where clause function quires and aggregate function quires are executed successfully.