TOPIC: MORE ON SQL
MODULE - 1

DISTANCE LEARNING PROGRAMME

THROUGH E-MODULE BY

ATOMIC ENERGY EDUCATION SOCIETY

MUMBAI

Contents of Module 1

- Order By Clause
- Types of SQL Functions – Scalar and Aggregate
- Aggregate Functions – Count(), Sum(), Avg(), Min() and Max()
- Group By Clause
- Having Clause
- Joins – EQUI and NATURAL

ORDER BY Clause

Syntax:
SELECT expressions
FROM tables
[**WHERE** conditions]
ORDER BY expression [**ASC** | **DESC**];

- **expressions:** It specifies the columns that you want to retrieve.
- **tables:** It specifies the tables, from where you want to retrieve records. There must be at least one table listed in the FROM clause.
- **WHERE conditions:** It is optional. It specifies conditions that must be fulfilled for the records to be selected.
- **ASC:** It is optional. It sorts the result set in ascending order by expression (default, if no modifier is provider).
- **DESC:** It is also optional. It sorts the result set in descending order by expression.

EXAMPLES

1. ORDER BY: without using ASC/DESC attribute

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

2. ORDER BY: with ASC attribute

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

EXAMPLES

3.ORDER BY: with DESC attribute

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

4.ORDER BY: using both ASC and DESC attributes

```
SELECT officer_name, address
FROM officers
WHERE officer_id < 5
ORDER BY officer_name DESC, address ASC;
```

Types of SQL Functions:

1. Single Row(or scalar) Functions: This type of functions work with a single row at a time. A single row function returns a result for every row of a queried table.

2. Multiple Row(or Group or Aggregate)Functions: This type of functions are the group functions that works with data of multiple rows at a time and returns a single result for that group.

Aggregate Functions

Aggregate functions work upon group of rows rather than single rows. That is why, these functions are sometimes called as multiple row functions. Some of aggregate functions are:

1)Count( )

2)Sum( )

3)Avg( )

4)Min( )

5)Max( )

Count() Function

Count() function is used to return the number of rows in a given column or expression.

The following are the syntax of the COUNT() function:

COUNT ({*[DISTINCT | ALL]expr})

- Returns the number of rows in the query.
- If you specify argument expr, this function returns rows where expr is not null. You can count either all rows, or only distinct values of expr.
- If you specify the asterisk (*), this function returns all rows, including duplicates and nulls.

EXAMPLES

1. Execute the following query that uses the COUNT(expression) function to calculates the total number of employees name available in the table:

```
SELECT COUNT(emp_name) FROM employees;
```

2. Execute the following statement that returns all rows from the employee table and WHERE clause specifies the rows whose value in the column emp_age is greater than 32:

```
SELECT COUNT(*) FROM employees
WHERE emp_age>32;
```

Sum() Function

The Sum() function is used to return the total summed value of an expression. It returns NULL if the result set does not have any rows.

Following is the syntax of sum() function in SQL:

SUM([DISTINCT |ALL]n)

Returns the sum of values of n

Example:

Execute the following query that calculates the total number of working hours of all employees in the table:

```
SELECT SUM(working_hours) AS "Total working hours“
FROM employees;
```

Avg() Function

The MySQL avg() is an aggregate function used to return the average value of an expression in various records.

The following are the basic syntax an avg() function in MySQL:

AVG([DISTINCT|ALL]n)

Returns average value of parameter(s) n.

Example

Execute the following query that calculates the **average working hours** of all employees in the table:

```
SELECT AVG(working_hours) Avg_working_hours
FROM employees;
```

MIN() Function

The MIN() function in MySQL is used to return the **minimum value** in a set of values from the table. It is an aggregate function that is useful when we need to find the smallest number, selecting the least expensive product, etc.

The following is the basic syntax of MIN() function in SQL:

MIN ([**DISTINCT|ALL]expr)**

Returns the minimum value of expr.

Example

Execute the following query that uses the MIN function to find the **minimum income** of the employee available in the table:

```
SELECT MIN(income) AS Minimum_Income
FROM employees;
```

MAX() Function

The MAX() function is used to return the maximum value in a set of values of an expression. This aggregate function is useful when we need to find the maximum number, selecting the most expensive product, or getting the largest payment to the customer from your table.

The following is the basic syntax of MAX() function in MySQL:

MAX([**DISTINCT** |ALL]expr)

Returns the maximum value of argument expr.

Example

Execute the following query that uses the MAX function to find the **maximum income** of the employee available in the table:

```
SELECT MAX(income) AS "Maximum Income"
FROM employees;
```

GROUP BY Clause

The GROUP BY Clause is used to collect data from multiple records and group the result by one or more column. It is generally used in a SELECT statement.

You can also use some aggregate functions like COUNT, SUM, MIN, MAX, AVG etc. on the grouped column.

Syntax:

SELECT expression1, expression2, ... expression_n,

aggregate_function (expression)

FROM tables

[**WHERE** conditions]

GROUP BY expression1, expression2, ... expression_n;

expression1, expression2, ... expression_n: It specifies the expressions that are not encapsulated within an aggregate function and must be included in the GROUP BY clause.

aggregate_function: It specifies a function such as SUM, COUNT, MIN, MAX, or AVG etc. tables: It specifies the tables, from where you want to retrieve the records. There must be at least one table listed in the FROM clause.

WHERE conditions: It is optional. It specifies the conditions that must be fulfilled for the records to be selected.

EXAMPLES

(i) GROUP BY Clause with COUNT function

```
SELECT address, COUNT(*)
FROM   officers
GROUP BY address;
```

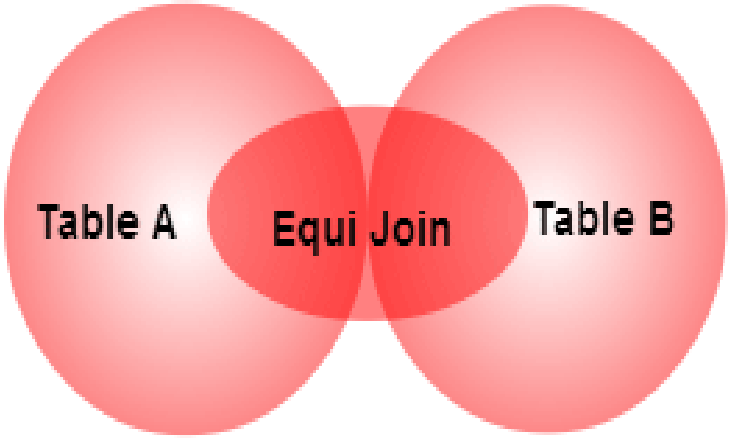
(ii) GROUP BY Clause with SUM function

```
SELECT emp_name, SUM(working_hours)
AS "Total working hours"
FROM employees
GROUP BY emp_name;
```

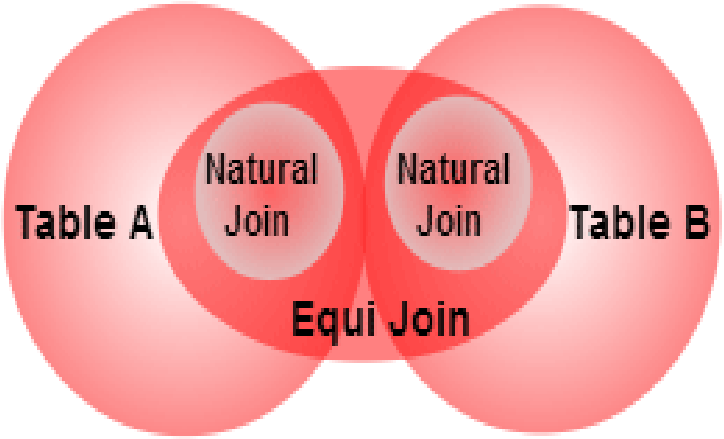
It can also be used with MIN, MAX and AVG functions.

JOINS

- A join is a query that combines rows from two or more tables. In join query, more than one tables are listed in FROM Clause. Some of the types of joins are:



Equi Join



Natural Join

EQUI-JOIN

SQL EQUI JOIN performs a JOIN against equality or matching column(s) values of the associated tables. An equal sign (=) is used as comparison operator in the where clause to refer equality.

You may also perform EQUI JOIN by using JOIN keyword by ON keyword and then specifying names of the columns along with their associated tables to check equality.

Syntax:

```
SELECT column_list
FROM table1,table2.....WHERE
table1.column_name=table2.column_name;
```

OR

```
SELECT * FROM table1 JOIN table2
[ON(join_condition)];
```

EXAMPLE:

```
SELECT  *
FROM Emp  ,Dept
where  Emp.Dno=Dept.Dno;
```

NATURAL-JOIN

The SQL NATURAL JOIN is a type of EQUI JOIN and is structured in such a way that, columns with the same name of associated tables will appear only once.

OR

A SQL NATURAL JOIN is a type of EQUI JOIN which occurs implicitly by comparing all the same names columns in both tables. The join result has only one column for each pair of equally named columns.

Syntax:

SELECT * FROM table1

NATUTAL JOIN table 2 ;

EXAMPLE:

SELECT * FROM Emp Natural join Dept;

Bibliography

I acknowledge that
the content and images are taken from Wikipedia, and Self generated SQL outputs

THANK YOU

Rani Sasty
PGT(ss)
AECS-4,Mumbai