MySQL - Group By Clause

The SQL **GROUP BY** clause is used in conjunction with the SELECT statement to arrange identical data into groups. This clause follows the WHERE clause in a SELECT statement and precedes the ORDER BY and HAVING clauses (if they exist).

The main purpose of grouping the records of a table based on particular columns is to perform calculations on these groups. Therefore, The GROUP BY clause is typically used with aggregate functions such as SUM(), COUNT(), AVG(), MAX(), or MIN() etc.

Syntax :

```
SELECT column_name(s)

FROM table_name

GROUP BY column_name(s);
```

Example :

```
SELECT country

FROM customers

GROUP BY country;
```

GROUP BY Clause with Aggregate Functions

Typically, we group the record of a table to perform calculations on them. Therefore, the SQL GROUP BY clause is often used with the aggregate functions such as SUM(), AVG(), MIN(), MAX(), COUNT(), etc.

Example 1 :

```
SELECT country, COUNT(country)

FROM customers

GROUP BY country;
```

Example 2 :

```
SELECT department, MIN(salary) as min_salary

FROM employee

GROUP BY department ;
```

Example 3 :

```
SELECT department, MAX(salary) as max_salary

FROM employee

GROUP BY department ;
```

Example 4 :

```
SELECT department, AVG(salary) as avg_salary

FROM employee

GROUP BY department ;
```

Example 5 :

```
SELECT department, SUM(salary) as sum_salary

FROM employee

GROUP BY department ;
```

GROUP BY with ORDER BY Clause

We can use the ORDER BY clause with GROUP BY in SQL to sort the grouped data by one or more columns.

Syntax :

```
SELECT column_name(s)..., aggregate_functions (column_name)

FROM table_name

GROUP BY column_name(s)

ORDER BY column1_name ASC/DESC, column2_name ASC/DESC,...;
```

Example 5 :

```
SELECT department, SUM(salary) as sum_salary

FROM employee

GROUP BY department

ORDER BY sum_salary DESC ;
```

GROUP BY with HAVING Clause

We can also use the GROUP BY clause with the HAVING clause to filter the grouped data in a table based on specific criteria.

Syntax :

```
SELECT column_name(s)..., aggregate_functions (column_name)

FROM table_name

GROUP BY column_name(s)

HAVING  condition ;
```

Example 1 :

```
SELECT department, SUM(salary) as sum_salary

FROM employee

GROUP BY department

HAVING sum_salary >= 60000;
```

Example 2 :

```
SELECT department, SUM(salary) as sum_salary

FROM employee

GROUP BY department

HAVING sum_salary >= 60000

ORDER BY department DESC ;
```

The SQL HAVING Clause

The SQL HAVING clause is similar to the WHERE clause; both are used to filter rows in a table based on specified criteria. However, the HAVING clause is used to filter grouped rows instead of single rows. These rows are grouped together by the GROUP BY clause, so the HAVING clause must always be followed by the GROUP BY clause.

Moreover, the HAVING clause can be used with aggregate functions such as COUNT(), SUM(), AVG(), etc., whereas the WHERE clause cannot be used with them.

The following code block shows the position of the HAVING Clause in a query –

```
SELECT

FROM

WHERE

GROUP BY

HAVING

ORDER BY
```