GROUP BY AND HAVING CLAUSE IN MYSQL

Presentation By
Chimirala Kowstubha

AGENDA

1. Group By
2. Having Clause
3. Applications

GROUP BY

Definition:

- GROUP BY clause in MySQL is used to collect data from multiple records and group the result by one or more column.
- It is often used with the aggregation functions like COUNT(), MAX(), MIN(), SUM(), AVG() in order to group results for better analyzation of the data.
- Group by is similar to distinct clause but the only difference is distinct clause is used instead of group by if there is no aggregate function in the select statement.
- If the objective is to remove duplicate rows, the DISTINCT clause should be used. However, if the aim is to group data and apply aggregate functions to each group, the GROUP BY clause is appropriate.

Syntax:

```
SELECT column1, column2, ..., columnN, aggregate_function(columnX)
FROM table_name
WHERE condition GROUP BY column1, column2, ..., columnN;
```

GROUP BY

Example:

Query: select sum(salary) as total_salary_expense, count(employee_id) as total_no_of_employees,department_id from hr.employees group by department_id;

Here this query returns the number of employees and total salary expense according to different departments.

Limitations:

- **Query performance:** When working with large datasets the query performance can be reduced if we use group by because MySQL has to sort and group the data.
- **Null value columns:** When trying to group by columns that contain null values, the GROUP BY clause can cause undesirable results because it considers null value as groupable value. To overcome this issue the COALESCE function is used to replace null items with a default value before grouping.

HAVING CLAUSE

Definition:

- Having clause is used after the group by to filter the aggregated columns based on the given condition.
- It is similar to where clause but the only difference is where clause filters the data individually before grouping whereas having filters the data after grouping.

Syntax:

```
SELECT column1, column2, ..., columnN, aggregate_function(columnX)
FROM table_name
WHERE condition GROUP BY column1, column2, ..., columnN HAVING
aggregate_function(columnX)>condition;
```

HAVING CLAUSE

Example:

Query: select sum(salary) as total_salary_expense, count(employee_id) as total_no_of_employees,department_id from hr.employees group by department_id having total_no_of_employees>100 ;

Here this query returns the number of employees and total salary expense according to different departments which have more than 100 employees.

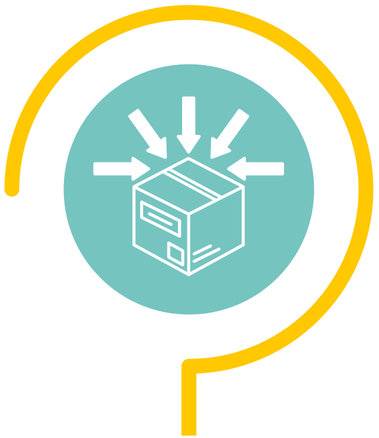
Advantages:

- It overcomes the disadvantage of where clause where aggregate fucnctions can not be included in conditions.
- It helps to use an aggregate function as a filter upon a group.

APPLICATIONS

GROUPING

Group by can be used when there is a need to filter the data according to some particular columns especially while handling large datasets.



AGGREGATION

Group by and having clause are used to filter the data including aggregate functions.

SELECTIVE ANALYSIS

When we have to do selective analysis on particular columns group by and having are used to get fast and better summarization.

KEY INSIGHTS

ROLL UP MODIFIER

Using the WITH ROLLUP modifier with GROUP BY allows to include extra rows that represent higher-level (that is, super-aggregate) summary operations. For example retrieving the total sales of products according to customer id.

ONLY_FULL_GROUP_BY

If the ONLY_FULL_GROUP_BY SQL mode is enabled, MySQL rejects queries for which the select list, HAVING condition, or ORDER BY list refer to nonaggregated columns that are neither named in the GROUP BY clause nor are functionally dependent on them whereas GROUP BY in MySQL permits the select list, HAVING condition, or ORDER BY list to refer to nonaggregated columns even if the columns are not functionally dependent on GROUP BY columns.

THANK YOU