## Assignment No.: 4

Name: Sourav Shailesh Toshniwal
Class: TY CSE-8 AIEC-1
Roll no.: 2213047
Batch: A

**Aim:** Database Querying – Simple queries, Queries using aggregate functions, GROUP BY and HAVING clauses. (https://learnsql.com/blog/examples-of-sql-group-by/)

A. Write a Group-by query for one/two columns in Manufacturing industry / **Hospital**/ Company table
B. Write a Having clause query for Manufacturing industry / **Hospital**/ Company table
C. Write a queries to make use of aggregate functions Count(),Sum(),Avg(),Min(),Max()

**Software required: MySQL**

**Theory:** In a database management system (DBMS), the GROUP BY clause and the HAVING clause are used in conjunction with the SELECT statement to perform advanced data analysis and filtering on groups of rows. Here's a brief introduction to each clause:

**GROUP BY Clause:**

> The GROUP BY clause is used to group rows in a result set based on one or more columns. It is commonly used in combination with aggregate functions like COUNT, SUM, AVG, MAX, or MIN to perform calculations on groups of data. The result of a GROUP BY query is a set of rows where each row represents a unique combination of values in the specified column(s) and the aggregate function(s) are applied to the corresponding groups.

For example, if you have a "Sales" table with columns like "Product", "Category", and "Revenue", you can use the GROUP BY clause to calculate the total revenue per product category. The query might look like this:

```
SELECT Category, SUM(Revenue) AS TotalRevenue
FROM Sales
GROUP BY Category;
```

**HAVING Clause:**

> The HAVING clause is used to filter the results of a GROUP BY query based on specified conditions. It allows you to apply filtering criteria to the grouped data after the aggregation has taken place. This clause operates similarly to the WHERE clause, but while the WHERE clause filters individual rows, the HAVING clause filters groups of rows.

Continuing with the previous example, if you want to retrieve only the product categories with total revenue greater than a certain value, you can use the HAVING clause. For instance:

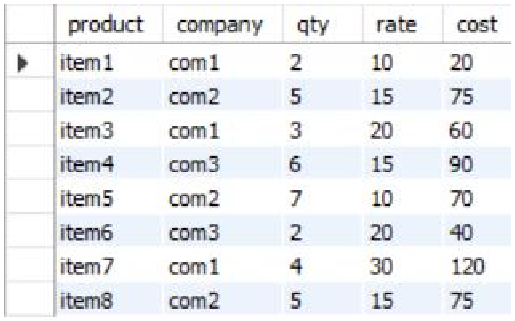
```
SELECT Category, SUM(Revenue) AS TotalRevenue
FROM Sales
GROUP BY Category
HAVING SUM(Revenue) > 100000;
```

This query will return the product categories with a total revenue greater than 100,000.

**SQL Commands:**

```
use TYAIEC;

create table assignment4(product TEXT, company TEXT, qty INT, rate INT, cost INT);

insert into assignment4 values('item1', 'com1', 2, 10, 20);
insert into assignment4 values('item2', 'com2', 5, 15, 75);
insert into assignment4 values('item3', 'com1', 3, 20, 60);
insert into assignment4 values('item4', 'com3', 6, 15, 90);
insert into assignment4 values('item5', 'com2', 7, 10, 70);
insert into assignment4 values('item6', 'com3', 2, 20, 40);
insert into assignment4 values('item7', 'com1', 4, 30, 120);
insert into assignment4 values('item8', 'com2', 5, 15, 75);

select product, count(qty) as qty from assignment4 group by cost;
select rate from assignment4 having count(rate) > 2;
select count(*), avg(rate), sum(cost), min(rate), max(rate) from assignment4;
select count(distinct company), avg(distinct rate) from assignment4;

select * from assignment4;
```

Output:



| product | company | qty | rate | cost |
|---|---|---|---|---|
| item1 | com1 | 2 | 10 | 20 |
| item2 | com2 | 5 | 15 | 75 |
| item3 | com1 | 3 | 20 | 60 |
| item4 | com3 | 6 | 15 | 90 |
| item5 | com2 | 7 | 10 | 70 |
| item6 | com3 | 2 | 20 | 40 |
| item7 | com1 | 4 | 30 | 120 |
| item8 | com2 | 5 | 15 | 75 |

**Example:**

**:**

**Conclusion:** In summary, the GROUP BY clause helps to group rows based on specific columns, while the HAVING clause allows you to filter the grouped data based on aggregate conditions. Together, they provide powerful tools for data analysis and summarization in DBMS.

**FAQs: (Answer all FAQs using suitable examples)**

I. What is the difference between the WHERE clause and the HAVING clause?

II. Can I use the GROUP BY clause without any aggregate functions?

III. Can I use the HAVING clause without the GROUP BY clause?

IV. Can I include columns in the SELECT statement that are not part of the GROUP BY clause?

V. Can I use multiple aggregate functions in the HAVING clause?

Ans:I. The main difference between the WHERE clause and the HAVING clause in SQL is their purpose and when they are applied:

- WHERE clause: The WHERE clause is used in SQL queries to filter rows from the result set before any grouping or aggregation is performed. It operates on individual rows, and you use it with SELECT, UPDATE, DELETE, and other DML statements to filter which rows are included in the operation. It does not work with aggregated values.

- HAVING clause: The HAVING clause is used in SQL queries that involve grouping using the GROUP BY clause. It filters the result set after grouping and aggregation have been performed. It operates on aggregated values, allowing you to filter groups of rows based on aggregate conditions (e.g., SUM, COUNT, AVG). You cannot use HAVING without GROUP BY.

II. Yes, you can use the GROUP BY clause without any aggregate functions. When you use GROUP BY without aggregate functions, it is typically used to group rows in the result set by one or more columns. This is useful when you want to see distinct values for specific columns and don't need to perform any aggregation on the grouped data. For example:

```sql
SELECT column1, column2

FROM table_name

GROUP BY column1, column2;
```

This query groups the rows by the values in `column1` and `column2` and returns distinct combinations of these columns.

III. No, you cannot use the HAVING clause without the GROUP BY clause. The HAVING clause is specifically designed to filter groups of rows that result from the use of the GROUP BY clause. It operates on aggregated values computed for each group. If you attempt to use HAVING without GROUP BY, you will encounter a syntax error in most SQL database systems.

IV. Yes, you can include columns in the SELECT statement that are not part of the GROUP BY clause, but their behavior may vary depending on the database system you are using and whether the column is part of an aggregate function or not. In standard SQL, when you include a column in the SELECT clause that is not part of the GROUP BY clause or an aggregate function, it should be in the GROUP BY clause, or it should be part of an aggregate function. However, some database systems, like MySQL, allow you to include non-aggregated columns in the SELECT clause without including them in the GROUP BY clause, but the result might not be what you expect, as it can lead to unpredictable results.

V. Yes, you can use multiple aggregate functions in the HAVING clause to filter groups based on multiple aggregate conditions. For example, you can filter groups based on having a minimum count of rows and a maximum sum of a specific column's values. Here's an example:

```sql

SELECT column1, SUM(column2) as total

FROM table_name

GROUP BY column1

HAVING COUNT(*) >= 2 AND SUM(column2) > 100;

```

In this example, the HAVING clause uses both COUNT(*) and SUM(column2) as aggregate conditions to filter the groups.

**Additional problem statements:**

Write an SQL query to analyze sales data by grouping it by product category and cI.alculate the total revenue for each category. Display only the categories with total revenue greater than $10,000.

II. Develop an SQL query to retrieve the department names and the count of employees in each department. Display only the departments that have more than 50 employees.

III. Create an SQL query to group the products by their suppliers and calculate the average stock quantity for each supplier. Display only the suppliers whose average stock quantity exceeds 500 units.

IV. Write an SQL query to group customers by their age range (e.g., 18-25, 26-35, etc.) and calculate the count of customers in each age range. Display only the age ranges with more than 100 customers.

V. Develop an SQL query to analyze exam scores by grouping them by the subject and calculate the average score for each subject. Display only the subjects with an average score above 80.