HAVING Clause in SQL

The HAVING clause places the condition in the groups defined by the GROUP BY clause in the SELECT statement.

This SQL clause is implemented after the 'GROUP BY' clause in the 'SELECT' statement.

This clause is used in SQL because we cannot use the WHERE clause with the SQL aggregate functions. Both WHERE and HAVING clauses are used for filtering the records in SQL queries.

Difference between HAVING and WHERE Clause

The difference between the WHERE and HAVING clauses in the database is the most important question asked during an IT interview.

The following table shows the comparisons between these two clauses, but the main difference is that the WHERE clause
uses condition for filtering records before any groupings are made, while HAVING clause uses condition for filtering values from a group.

HAVING	WHERE
1. The HAVING clause is used in database systems to fetch the data/values from the groups according to the given condition.	1. The WHERE clause is used in database systems to fetch the data/values from the tables according to the given condition.
2. The HAVING clause is always executed with the GROUP BY clause.	2. The WHERE clause can be executed without the GROUP BY clause.

3. The HAVING clause can include SQL aggregate functions in a query or statement.	3. We cannot use the SQL aggregate function with WHERE clause in statements.
4. We can only use SELECT statement with HAVING clause for filtering the records.	4. Whereas, we can easily use WHERE clause with UPDATE, DELETE, and SELECT statements.
5. The HAVING clause is used in SQL queries after the GROUP BY clause.	5. The WHERE clause is always used before the GROUP BY clause in SQL queries.
6. We can implements this SQL clause in column operations.	6. We can implements this SQL clause in row operations.
7. It is a post-filter.	7. It is a pre-filter.
8. It is used to filter groups.	8. It is used to filter the single record of the table.

Syntax of HAVING clause in SQL

1. **SELECT** column_Name1, column_Name2, ..., column_NameN aggregate_function_name(column_Name) **FROM** table_name **GROUP BY** column_Name1 **HAVING** condition;

Examples of HAVING clause in SQL

In this article, we have taken the following four different examples which will help you how to use the HAVING clause with different SQL aggregate functions:

Example 1: Let's take the following **Employee** table, which helps you to analyze the HAVING clause with SUM aggregate function:

Emp_Id	Emp_Name	Emp_Salary	Emp_City
201	Abhay	2000	Goa
202	Ankit	4000	Delhi
203	Bheem	8000	Jaipur
204 Ram	2000	Goa	
205	Sumit	5000	Delhi

If you want to add the salary of employees for each city, you have to write the following query:

```
1. SELECT SUM(Emp_Salary), Emp_City FROM Employee GROUP BY Emp_City;
```

The output of the above query shows the following output:

SUM(Emp_Salary)	Emp_City
4000	Goa
9000	Delhi
8000	Jaipur

Now, suppose that you want to show those cities whose total salary of employees is more than 5000. For this case, you have to type the following query with the HAVING clause in SQL:

```
1. SELECT SUM(Emp_Salary), Emp_City FROM Employee GROUP BY Emp_City
   HAVING SUM(Emp_Salary)>5000;
```

The output of the above SQL query shows the following table in the output:

SUM(Emp_Salary)	Emp_City
9000	Delhi
8000	Jaipur

Example 2: Let's take the following **Student_details** table, which helps you to analyze the HAVING clause with the COUNT aggregate function:

Roll_No	Name	Marks	Age
1	Rithik	91	20
2	Kapil	60	19
3	Arun	82	17
4	Ram	92	18
5	Anuj	50	20
6	Suman	88	18
7	Sheetal	57	19
8	Anuj	64	20

Suppose, you want to count the number of students from the above table according to their age. For this, you have to write the following query:

1. **SELECT** COUNT(Roll_No), Age **FROM** Student_details **GROUP BY** Age ;

The above query will show the following output:

Count(Roll_No)	Age
3	20
2	19
1	17
2	18

Now, suppose that you want to show the age of those students whose roll number is more than and equals 2. For this case, you have to type the following query with the HAVING clause in SQL:

1. **SELECT** COUNT(Roll_No), Age **FROM** Student_details **GROUP BY** Age **HAVING** COUNT(Roll_No) >= 2 ;

The output of the above SQL query shows the following table in the output:

Count(Roll_No)	Age
3	20
2	19
2	18

Example 3: Let's take the following **Employee** table, which helps you to analyze the HAVING clause with MIN and MAX aggregate function:

Emp_ID	Name	Emp_Salary	Emp_Dept
1001	Anuj	9000	Finance
1002	Saket	4000	HR
1003	Raman	3000	Coding
1004	Renu	6000	Coding
1005	Seenu	5000	HR
1006	Mohan	10000	Marketing
1007	Anaya	4000	Coding
1008	Parul	8000	Finance

MIN Function with HAVING Clause:

If you want to show each department and the minimum salary in each department, you have to write the following query:

1. **SELECT MIN**(Emp_Salary), Emp_Dept **FROM** Employee **GROUP BY** Emp_Dept;

The output of the above query shows the following output:

MIN(Emp_Salary)	Emp_Dept
8000	Finance

4000	HR
3000	Coding
10000	Marketing

Now, suppose that you want to show only those departments whose minimum salary of employees is greater than 4000. For this case, you have to type the following query with the HAVING clause in SQL:

1. **SELECT MIN**(Emp_Salary), Emp_Dept **FROM** Employee **GROUP BY** Emp_Dept **HAVING MIN**(Emp_Salary) > 4000 ;

The above SQL query shows the following table in the output:

MIN(Emp_Salary)	Emp_Dept
8000	Finance
10000	Marketing

MAX Function with HAVING Clause:

In the above employee table, if you want to list each department and the maximum salary in each department. For this, you have to write the following query:

1. **SELECT MAX**(Emp_Salary), Emp_Dept **FROM** Employee **GROUP BY** Emp_Dept;

The above query will show the following output:

MAX(Emp_Salary)	Emp_Dept
9000	Finance

5000	HR
6000	Coding
10000	Marketing

Now, suppose that you want to show only those departments whose maximum salary of employees is less than 8000. For this case, you have to type the following query with the HAVING clause in SQL:

1. **SELECT MAX**(Emp_Salary), Emp_Dept **FROM** Employee **GROUP BY** Emp_Dept **HAVING MAX**(Emp_Salary) < 8000 ;

The output of the above SQL query shows the following table in the output:

MAX(Emp_Salary)	Emp_Dept
5000	HR
6000	Coding

Example 4: Let's take the following **Employee_Dept** table, which helps you to analyze the HAVING clause with AVG aggregate function:

Emp_ID	Name	Emp_Salary	Emp_Dept
1001	Anuj	8000	Finance
1002	Saket	4000	HR
1003	Raman	3000	Coding
1004	Renu	6000	Coding

1005	Seenu	5000	HR
1006	Mohan	10000	Marketing
1007	Anaya	4000	Coding
1008	Parul	6000	Finance

If you want to find the average salary of employees in each department, you have to write the following query:

```
1. SELECT AVG(Emp_Salary), Emp_Dept FROM Employee_Dept GROUP BY
   Emp_Dept;
```

The above query will show the following output:

AVG(Emp_Salary)	Emp_Dept
7000	Finance
4500	HR
6500	Coding
10000	Marketing

Now, suppose that you want to show those departments whose average salary is more than and equals 6500. For this case, you have to type the following query with the HAVING clause in SQL:

```
1. SELECT AVG(Emp_Salary), Emp_Dept FROM Employee_Dept GROUP BY
   Emp_Dept HAVING AVG(Emp_Salary) > 6500 ;
```

The above SQL query will show the following table in the output:

AVG(Emp_Salary)	Emp_Dept
7000	Finance
6500	Coding
10000	Marketing

https://www.javatpoint.com/having-clause-in-sql