

Can the SAS language's "SAS: Use HAVING Clause Within PROC SQL" feature be implemented by using the HAVING clause within PROC SQL?

Authored by
stats writer

June 25, 2024

RECOMMENDED CITATION

stats writer (2024). *Can the SAS language's "SAS: Use HAVING Clause Within PROC SQL" feature be implemented by using the HAVING clause within PROC SQL?*.

PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=152916>

The "SAS: Use HAVING Clause Within PROC SQL" feature allows users of the SAS language to utilize the HAVING clause within the PROC SQL procedure. This feature enables users to filter and group data within their SQL queries, similar to how the HAVING clause is used in other programming languages. By implementing the HAVING clause within PROC SQL, users can efficiently manipulate and analyze data to meet their specific needs. This feature is a valuable addition to the SAS language and enhances its capabilities for data analysis and manipulation.

SAS: Use HAVING Clause Within PROC SQL

You can use the HAVING clause within PROC SQL in SAS to filter for rows that meet a certain condition.

Note the subtle difference between the WHERE and HAVING clause:

WHERE filters rows before any grouping occurs. HAVING filters rows after any grouping occurs.

The following example shows how to use the HAVING clause in practice.

Example: How to Use HAVING Clause in PROC SQL

Suppose we have the following dataset in SAS that contains information about various basketball players:

```
/*create dataset*/  
data my_data;  
input team $ position $ points;
```

datalines;

A Guard 22

A Guard 20

A Guard 30

A Forward 14

A Forward 11

B Guard 12

B Guard 22

B Forward 30

B Forward 9

B Forward 12

B Forward 25

C Guard 22

C Guard 19

C Guard 10

;

run;

/*view dataset*/

proc printdata=my_data;

Obs	team	position	points
1	A	Guard	22
2	A	Guard	20
3	A	Guard	30
4	A	Forward	14
5	A	Forward	11
6	B	Guard	12
7	B	Guard	22
8	B	Forward	30
9	B	Forward	9
10	B	Forward	12
11	B	Forward	25
12	C	Guard	22
13	C	Guard	19
14	C	Guard	10

We can use the following code to calculate the sum of points scored by players on each team WHERE the players are Guards and then filter the results to only show teams HAVING a sum greater than 50:

```
proc sql;  
select team, sum(points) as sum_points  
from my_data  
where position='Guard'  
group by team  
having sum_points>50;  
quit;
```

team	sum_points
A	72
C	51

Here is exactly how this code worked:

**First, we used SELECT to select team and the sum of points
Then, we used WHERE to filter for rows where position was 'Guard'
Then, we used GROUP to group the results by team
Then we used HAVING to filter for teams with sum of points > 50**

Here is what the results of this query would have looked like if we didn't include the HAVING statement:

```
proc sql;  
select team, sum(points) as sum_points  
from my_data  
where position='Guard'  
group by team;  
quit;
```

team	sum_points
A	72
B	34
C	51

Notice that the sum of points for team B was not greater than 50.

The following tutorials explain how to perform other common tasks in SAS:

ARABPSYCHOLOGY.COM