

How can I use the WHERE option with the SET statement in SAS to perform a SAS (Sequential Access Storage) operation?

Authored by
stats writer

June 23, 2024

RECOMMENDED CITATION

stats writer (2024). *How can I use the WHERE option with the SET statement in SAS to perform a SAS (Sequential Access Storage) operation?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=148241>

The WHERE option in the SET statement of SAS allows for the execution of a SAS operation by specifying conditions for selecting or excluding observations from a dataset. This option is useful for performing a Sequential Access Storage (SAS) operation, where specific observations need to be accessed and processed. By using the WHERE option, users can efficiently filter and manipulate data to meet their specific needs, making data processing more precise and streamlined. This feature enhances the functionality of SAS and enables users to perform complex operations with ease.

SAS: Use WHERE Option with SET

You can use the WHERE option with SET in SAS to create a new dataset that only includes rows from another dataset where certain conditions are met.

Here are two common ways to use this option in practice:

Method 1: Use WHERE and SET with One Condition

```
data new_data;  
set my_data (where = (points>20));  
run;
```

This example creates a new dataset called new_data that only includes rows from my_data where the value in the points column is greater than 20.

Method 2: Use WHERE and SET with Multiple

Conditions

```
data new_data;  
set my_data (where = (points>20 or team="Rockets"));  
run;
```

This example creates a new dataset called `new_data` that only includes rows from `my_data` where the value in the `points` column is greater than 20 or the value in the `team` column is equal to `Rockets`.

The following examples show how to use each method in practice with the following dataset in SAS:

```
/*create dataset*/  
data my_data;  
input team $ points assists;  
datalines;  
Mavs 22 10  
Rockets 12 14  
Spurs 29 8  
Kings 13 10  
Warriors 44 10  
Heat 18 8  
Magic 11 5
```

Pelicans 19 3

Blazers 12 8

;

run;

/*view dataset*/

proc printdata=my_data;

Obs	team	points	assists
1	Mavs	22	10
2	Rockets	12	14
3	Spurs	29	8
4	Kings	13	10
5	Warriors	44	10
6	Heat	18	8
7	Magic	11	5
8	Pelicans	19	3
9	Blazers	12	8

Example 1: Use WHERE and SET with One Condition

We can use the following syntax to create a new dataset called new_data that only includes rows from my_data where the value in the points column is greater than 20:

/*create new dataset*/

data new_data;

```
set my_data (where = (points>20));  
run;
```

```
/*view new dataset*/  
proc print data=new_data;
```

Obs	team	points	assists
1	Mavs	22	10
2	Spurs	29	8
3	Warriors	44	10

Notice that only the rows with a value greater than 20 in the points column are included in this dataset.

Example 2: Use WHERE and SET with Multiple Conditions

```
/*create new dataset*/  
data new_data;  
set my_data (where = (points>20 or team="Rockets"));  
run;
```

```
/*view new dataset*/  
proc print data=new_data;
```

Obs	team	points	assists
1	Mavs	22	10
2	Rockets	12	14
3	Spurs	29	8
4	Warriors	44	10

Notice that only the rows with a value greater than 20 in the points column or a value of "Rockets" in the team column are included in this dataset.

Note: You could use and instead of or to only include rows that meet multiple conditions.

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