

Can you explain how to use the PROC SORT statement with the KEEP statement in SAS?

Authored by
stats writer

June 25, 2024

RECOMMENDED CITATION

stats writer (2024). *Can you explain how to use the PROC SORT statement with the KEEP statement in SAS?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=151663>

The PROC SORT statement is a powerful tool in SAS that allows users to sort data sets based on specific variables or criteria. The KEEP statement, when used in conjunction with the PROC SORT statement, allows users to select and retain only specific variables from the data set while sorting. This means that the user can control which variables are included in the final sorted data set, making it easier to analyze and work with. To use the PROC SORT statement with the KEEP statement, simply specify the desired variables after the KEEP statement and they will be retained in the sorted data set. This combination of statements is particularly useful when dealing with large data sets, as it can expedite the sorting process and reduce the amount of unnecessary data being processed.

SAS: Use PROC SORT with KEEP Statement

You can use PROC SORT with the KEEP statement in SAS to sort the rows in a dataset and only keep specific columns after sorting.

You can use the following basic syntax to do so:

```
proc sort data=my_data out=sorted_data (keep=var1  
var2);  
by var2;  
run;
```

This particular example sorts the rows in the dataset based on the values in the var2 column and then only keeps the var1 and var2 columns after sorting.

The following example shows how to use this syntax in

practice.

Example: Use PROC SORT with KEEP Statement in SAS

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

```
/*create dataset*/  
data my_data;  
input team $ points assists;  
datalines;  
Mavs 113 22  
Pacers 95 19  
Cavs 100 34  
Lakers 114 20  
Heat 123 39  
Kings 100 22  
Raptors 105 11  
Hawks 95 25  
Magic 103 26  
Spurs 119 29  
;  
run;  
  
/*view dataset*/
```

```
proc printdata=my_data;
```

Obs	team	points	assists
1	Mavs	113	22
2	Pacers	95	19
3	Cavs	100	34
4	Lakers	114	20
5	Heat	123	39
6	Kings	100	22
7	Raptors	105	11
8	Hawks	95	25
9	Magic	103	26
10	Spurs	119	29

We could use the following syntax to sort the rows in the dataset based on the values in the points column:

```
/*sort rows in dataset based on values in points column*/
```

```
proc sortdata=my_data out=sorted_data;
```

```
by points;
```

```
run;
```

```
/*view sorted dataset*/
```

```
proc printdata=sorted_data;
```

Obs	team	points	assists
1	Pacers	95	19
2	Hawks	95	25
3	Cavs	100	34
4	Kings	100	22
5	Magic	103	26
6	Raptors	105	11
7	Mavs	113	22
8	Lakers	114	20
9	Spurs	119	29
10	Heat	123	39

Notice that the rows are now sorted in ascending order based on the values in the points column.

By default, SAS keeps all of the columns in the dataset after sorting.

However, you can use the KEEP statement to specify which columns to keep after sorting.

For example, we can use the following syntax to sort the rows in the dataset based on the values in the points column and then only keep the team and points columns:

```
/*sort rows in dataset based on values in points column  
and only keep team and points*/
```

```
proc sort data=my_data out=sorted_data (keep=team
points);
by points;
run;
```

```
/*view sorted dataset*/
```

```
proc printdata=sorted_data;
```

Obs	team	points
1	Pacers	95
2	Hawks	95
3	Cavs	100
4	Kings	100
5	Magic	103
6	Raptors	105
7	Mavs	113
8	Lakers	114
9	Spurs	119
10	Heat	123

Once again, the rows are sorted in ascending order based on the values in the points column but this time we used the KEEP statement to only keep the team and points columns after sorting.

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