

Can we use the KEEP and DROP statements in SAS to implement the SAS approach known as “SAS: Use the KEEP and DROP Statements”?

Authored by
stats writer

July 1, 2024

RECOMMENDED CITATION

stats writer (2024). *Can we use the KEEP and DROP statements in SAS to implement the SAS approach known as “SAS: Use the KEEP and DROP Statements”?*.

PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=164059>

The "SAS: Use the KEEP and DROP Statements" approach is a technique used in SAS programming to selectively choose or exclude variables from a dataset. This approach can be implemented by utilizing the KEEP and DROP statements in SAS. The KEEP statement allows for the selection of specific variables to be included in the dataset, while the DROP statement excludes specified variables. By using these statements, the SAS approach can be effectively applied to manipulate datasets and analyze data with precision and efficiency.

SAS: Use the KEEP and DROP Statements

You can use the KEEP and DROP statements in SAS when creating a new dataset to keep or drop specific variables from an existing dataset.

These statements use the following basic syntax:

Method 1: Choose Which Columns to KEEP

```
data new_data;  
set original_data;  
keep var1 var3;  
run;
```

Method 2: Choose Which Columns to DROP

```
data new_data;  
set original_data;  
drop var5;
```

run;

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

/*create dataset*/

data original_data;

input team \$ points rebounds;

datalines;

Warriors 25 8

Wizards 18 12

Rockets 22 6

Celtics 24 11

Thunder 27 14

Spurs 33 19

Nets 31 20

;

run;

/*view dataset*/

proc printdata=original_data;

Obs	team	points	rebounds
1	Warriors	25	8
2	Wizards	18	12
3	Rockets	22	6
4	Celtics	24	11
5	Thunder	27	14
6	Spurs	33	19
7	Nets	31	20

Example 1: Choose Which Columns to KEEP

The following code shows how to create a new dataset in SAS by using the KEEP statement to only keep the team and rebounds variables from the original dataset:

```
/*create new dataset*/  
data new_data;  
set original_data;  
keep team rebounds;  
run;  
  
/*view new dataset*/proc printdata=new_data;
```

Obs	team	rebounds
1	Warriors	8
2	Wizards	12
3	Rockets	6
4	Celtics	11
5	Thunder	14
6	Spurs	19
7	Nets	20

Notice that we kept the team and rebounds variables from the original dataset and any other variables were simply dropped.

Example 2: Choose Which Columns to DROP

The following code shows how to create a new dataset in SAS by using the DROP statement to drop the rebounds variable from the original dataset:

```
/*create new dataset*/
```

```
data new_data;
```

```
set original_data;
```

```
drop rebounds;
```

```
run;
```

```
/*view new dataset*/proc printdata=new_data;
```

Obs	team	points
1	Warriors	25
2	Wizards	18
3	Rockets	22
4	Celtics	24
5	Thunder	27
6	Spurs	33
7	Nets	31

Notice that the rebounds variable was dropped from the dataset while all other variables from the original dataset were kept.

KEEP vs. DROP: Which Statement Should You Use?

The KEEP and DROP statements accomplish the same outcome: They both allow you to create a new dataset by keeping or dropping certain variables from an existing dataset.

Typically we use the KEEP statement when we only want to keep a few variables from the original dataset since this is faster than typing out all of the variables we'd like to drop.

Conversely, we typically use the DROP statement when we only want to drop a few variables from the original

dataset since this is faster than typing out all of the variables we'd like to keep.

Additional Resources

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

ARABPSYCHOLOGY.COM