

How can I utilize the IN= option in SAS?

Authored by
stats writer

June 23, 2024

RECOMMENDED CITATION

stats writer (2024). *How can I utilize the IN= option in SAS?*. PSYCHOLOGICAL SCALES.
Retrieved from <https://scales.arabpsychology.com/?p=148122>

The IN= option in SAS allows users to specify a variable that will indicate whether a particular observation meets certain conditions. This can be useful in situations where data needs to be filtered or grouped based on specific criteria. By utilizing the IN= option, users can easily identify and analyze subsets of data within a larger dataset. This option can also be combined with other SAS statements and functions to further manipulate and analyze the data. Overall, the IN= option provides a powerful tool for data manipulation and analysis in SAS.

Use the IN= Option in SAS

You can use the IN= option in SAS to create a Boolean variable that indicates whether or not the current observation comes from the input dataset.

The IN= option is commonly used when you append two datasets together and you'd like to know if a particular row in the resulting dataset came from one of the specific input datasets.

The following example shows how to use the IN= option in practice.

Example: How to Use the IN= Option in SAS

Suppose we have two datasets that contain information about basketball players in the Eastern and Western conferences of the NBA:

```
/*create East dataset*/
```

```
data east_data;  
input team $ points;  
datalines;  
Celtics 22  
Pistons 14  
Nets 35  
Hornets 19  
Magic 22  
;  
run;
```

```
/*create West dataset*/
```

```
data west_data;  
input team $ points;  
datalines;  
Mavs 40  
Rockets 39  
Warriors 23  
Lakers 19  
Clippers 25  
;  
run;
```

```
/*view datasets*/
```

```
proc printdata=east_data;  
proc printdata=west_data;
```

Obs	team	points
1	Celtics	22
2	Pistons	14
3	Nets	35
4	Hornets	19
5	Magic	22

Obs	team	points
1	Mavs	40
2	Rockets	39
3	Warriors	23
4	Lakers	19
5	Clippers	25

We can use the following syntax to create a new dataset that appends these two datasets together:

```
/*create new dataset*/  
data all_data;  
set east_data west_data;  
run;
```

```
/*view new dataset*/
```

```
proc printdata=all_data;
```

Obs	team	points
1	Celtics	22
2	Pistons	14
3	Nets	35
4	Hornets	19
5	Magic	22
6	Mavs	40
7	Rockets	39
8	Warriors	23
9	Lakers	19
10	Clippers	25

Notice that each row from each dataset belongs to the new dataset.

To know which dataset each row came from, we can use the IN= option as follows:

```
/*create new dataset*/  
data all_data;  
set east_data west_data(in=i);  
if i then conf='West';  
else conf='East';  
run;
```

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

Obs	team	points	conf
1	Celtics	22	East
2	Pistons	14	East
3	Nets	35	East
4	Hornets	19	East
5	Magic	22	East
6	Mavs	40	West
7	Rockets	39	West
8	Warriors	23	West
9	Lakers	19	West
10	Clippers	25	West

By using the IN= option, we are able to create a new column called conf that takes on a value of "East" if the row came from the dataset called east_data and a value of West" if the row came from the dataset called west_data.

Note that we could also use only an IF statement without an ELSE statement to create a new column that simply takes on a value of * if the row came from the dataset called east_data:

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

```
data all_data;  
set east_data(in=i) west_data;  
if i then east_conf='*';  
run;  
  
/*view new dataset*/  
proc printdata=all_data;
```

Obs	team	points	east_conf
1	Celtics	22	*
2	Pistons	14	*
3	Nets	35	*
4	Hornets	19	*
5	Magic	22	*
6	Mavs	40	
7	Rockets	39	
8	Warriors	23	
9	Lakers	19	
10	Clippers	25	

By using the IN= option, we are able to create a new column called east_conf that takes on a value of * if the row came from the dataset called east_data and no value if the row came from the dataset called west_data.

The following tutorials explain how to perform other

common tasks in SAS:

ARABPSYCHOLOGY.COM