

“What is the recommended method for removing duplicate values in SAS using PROC SORT with the NODUPKEY option?”

Authored by
stats writer

June 26, 2024

RECOMMENDED CITATION

stats writer (2024). “What is the recommended method for removing duplicate values in SAS using PROC SORT with the NODUPKEY option?”. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=153166>

The recommended method for removing duplicate values in SAS is by using the PROC SORT procedure with the NODUPKEY option. This option ensures that only unique values are retained in the sorted dataset, eliminating any duplicate values. This method is efficient and convenient as it does not require additional coding or complex data manipulation techniques. By using this approach, data analysts can easily identify and remove duplicate values, ensuring the accuracy and reliability of their analysis results. Additionally, the NODUPKEY option can be applied to multiple variables, making it a versatile solution for handling duplicate values in SAS.

SAS: Use PROC SORT with NODUPKEY

You can use PROC SORT in SAS with NODUPKEY to order the observations in a dataset by one or more variables and remove any duplicates.

The following example shows how to use this procedure with the following dataset in SAS:

```
/*create dataset*/  
data original_data;  
input team $ points rebounds;  
datalines;  
A 12 8  
A 12 8  
A 12 8  
A 23 9  
A 20 12  
A 14 7
```

A 14 7

B 20 2

B 20 5

B 29 4

B 14 7

B 20 2

B 20 2

B 20 5

;

run;

/*view dataset*/

proc printdata=original_data;

Obs	team	points	rebounds
1	A	12	8
2	A	12	8
3	A	12	8
4	A	23	9
5	A	20	12
6	A	14	7
7	A	14	7
8	B	20	2
9	B	20	5
10	B	29	4
11	B	14	7
12	B	20	2
13	B	20	2
14	B	20	5

Example: Using PROC SORT with NODUPKEY in SAS

Suppose we simply use proc sort to sort the observations in the dataset in ascending order (smallest to largest) based on the value in the points column:

```
/*sort by points ascending*/  
proc sortdata=original_data out=data2;  
by points;  
run;
```

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

proc printdata=data2;

Obs	team	points	rebounds
1	A	12	8
2	A	12	8
3	A	12	8
4	A	14	7
5	A	14	7
6	B	14	7
7	A	20	12
8	B	20	2
9	B	20	5
10	B	20	2
11	B	20	2
12	B	20	5
13	A	23	9
14	B	29	4

Notice that the observations are sorted in ascending order based on the value in the points column.

However, there are several observations that are duplicates.

To sort the observations based on the values in the points column and remove all duplicates, we can add nodupkey after the proc sort statement:

/*sort by points ascending and remove duplicates*/

```
proc sort data=original_data out=data3 nodupkey;  
by points;  
run;
```

```
/*view sorted dataset*/  
proc print data=data3;
```

Obs	team	points	rebounds
1	A	12	8
2	A	14	7
3	A	20	12
4	A	23	9
5	B	29	4

The observations are now sorted in ascending order based on the value in the points column and all duplicate observations have been removed.

Note that we can also add the argument descending to instead sort the observations based on the value in the points column in descending order and remove all duplicates:

```
/*sort by points descending and remove duplicates*/  
proc sort data=original_data out=data4 nodupkey;
```

```
bydescending points;  
run;
```

```
/*view sorted dataset*/  
proc printdata=data4;
```

Obs	team	points	rebounds
1	B	29	4
2	A	23	9
3	A	20	12
4	A	14	7
5	A	12	8

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