

# How do you perform an outer join in SAS?

Authored by  
**stats writer**

June 25, 2024

## RECOMMENDED CITATION

stats writer (2024). *How do you perform an outer join in SAS?*. PSYCHOLOGICAL SCALES.  
Retrieved from <https://scales.arabpsychology.com/?p=151597>

An outer join in SAS is a data manipulation technique used to combine two or more tables in a way that preserves all data from both tables, even if there is no matching data in one of the tables. This is achieved by using the MERGE statement in SAS, which allows for the joining of tables based on a common variable or set of variables. The result of an outer join will include all rows from both tables, with missing values inserted for any unmatched variables. This is a useful tool for analyzing and comparing data from multiple sources, as it ensures that no data is lost in the merging process.

## Perform an Outer Join in SAS (With Example)

You can use the following basic syntax to perform an outer join with two datasets in SAS:

```
proc sql;  
create table final_table asselect coalesce(x.team,  
y.team) as team, x.team, x.points, y.team, y.assists  
from data1 as x full join data2 as y  
on x.team = y.team;  
quit;
```

This particular example performs an outer join using the full join statement and returns all rows from the datasets called data1 and data2.

The following example shows how to use this syntax in practice.

## Example: Perform an Outer Join in SAS

Suppose we have the following two datasets in SAS that contain information about various basketball teams:

```
/*create datasets*/  
data data1;  
input team $ points;  
datalines;  
A 18  
B 22  
C 19  
D 14  
E 14  
F 11  
G 20  
H 28  
;  
run;
```

```
data data2;  
input team $ assists;  
datalines;  
A 4
```

**B 9**

**C 14**

**D 13**

**L 10**

**M 8**

**;**

**run;**

***/\*view datasets\*/***

**proc printdata=data1;**

**proc printdata=data2;**

ARABPSYCHOLOGY.COM

Obs	team	points
1	A	18
2	B	22
3	C	19
4	D	14
5	E	14
6	F	11
7	G	20
8	H	28

Obs	team	assists
1	A	4
2	B	9
3	C	14
4	D	13
5	L	10
6	M	8

**We can use the following syntax to perform an outer join and create a new dataset that contains each row from both datasets:**

```
/*perform outer join*/  
proc sql;  
create table final_table asselect coalesce(x.team,  
y.team) as team, x.team, x.points, y.team, y.assists  
from data1 as x full join data2 as y  
on x.team = y.team;
```

```
quit;
```

```
/*view results of outer join*/
```

```
proc printdata=final_table;
```

Obs	team	points	assists
1	A	18	4
2	B	22	9
3	C	19	14
4	D	14	13
5	E	14	.
6	F	11	.
7	G	20	.
8	H	28	.
9	L	.	10
10	M	.	8

The resulting dataset contains every row from each individual dataset.

Note that we had to use the function to ensure that the team names from both datasets were returned in the resulting dataset.

If we didn't use this function, then only the values from the team column in the first dataset would be displayed in the resulting dataset.

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

ARABPSYCHOLOGY.COM