

How to Perform an Outer Join in SAS: A Step-by-Step Guide

Authored by
stats writer

November 20, 2025

RECOMMENDED CITATION

stats writer (2025). *How to Perform an Outer Join in SAS: A Step-by-Step Guide*.
PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=98585>

In SAS, an outer join can be performed by specifying the **OUTER** keyword in the **JOIN** statement. This will return all of the values from both tables, including any unmatched rows from either table. The **OUTER** keyword can be used with any of the join types - left, right, or full outer join - and can be used to join two or more tables.

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

```
proc sql;  
create table final_table as  
select 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 print data=data1;  
proc print data=data2;
```

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 as  
select 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 print data=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: