How to Perform an Outer Join in PySpark with a Practical Example

Authored by

stats writer

February 10, 2026

Performing an outer join in PySpark is a method of combining two datasets by matching their common values and including all values from both datasets, even if there is no match. This can be achieved by using the "join" function in PySpark and specifying the "how" parameter as "outer". An example of an outer join in PySpark would be combining a dataset of customers with a dataset of purchases. This would include all customers, even those who have not made a purchase, and all purchases, even those made by customers not in the initial dataset.

Do an Outer Join in PySpark (With Example)

You can use the following basic syntax to perform an outer join in PySpark:

df_joined = df1.join(df2, on=, how='full').show()

This particular example will perform an outer join using the DataFrames named df1 and df2 by joining on the column named team.

All rows from both DataFrames will be returned in the final DataFrame.

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

Example: How to Do an Outer Join in PySpark

Suppose we have the following DataFrame named df1:

from pyspark.sql import SparkSession

```
spark = SparkSession.builder.getOrCreate()

#define data
data1 = ,
,
,
,
,
]

#define column names
columns1 =

#create dataframe using data and column names
df1 = spark.createDataFrame(data1, columns1)

#view dataframe
df1.show()

+--------+------+
| team|points|
+--------+------+
| Mavs| 11|
| Hawks| 25|
| Nets| 32|
```

```
| Kings| 15|
|Warriors| 22|
| Suns| 17|
+--------+------+
```

And suppose we have another DataFrame named df2:

```
#define data
data2 = ,
,
,
,
]

#define column names
columns2 =

#create dataframe using data and column names
df2 = spark.createDataFrame(data2, columns2)

#view dataframe
df2.show()

+---------+-------+
| team|assists|
```

```
+---------+-------+
| Mavs| 4|
| Nets| 7|
| Suns| 8|
|Grizzlies| 12|
| Kings| 7|
+---------+-------+
```

We can use the following syntax to perform an outer join between these two DataFrames by joining on values from the team column:

```
#perform outer join using 'team' column
df_joined = df1.join(df2, on=, how='full').show()
```

```
+---------+------+-------+
| team|points|assists|
+---------+------+-------+
|Grizzlies| null| 12|
| Hawks| 25| null|
| Kings| 15| 7|
| Mavs| 11| 4|
| Nets| 32| 7|
| Suns| 17| 8|
```

| Warriors| 22| null|
+---------+------+-------+

Notice that the resulting DataFrame contains all rows from both DataFrames.

Note that if a team name didn't appear in both DataFrames, then a value of null appeared in either the points or assists column.

For example, the team name "Hawks" did not exist in df2, so this row received a value of null in the assists column of the final joined DataFrame.

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