

# How can I group a Pandas dataframe by two columns and aggregate the data?

Authored by  
**stats writer**

June 24, 2024

## RECOMMENDED CITATION

stats writer (2024). *How can I group a Pandas dataframe by two columns and aggregate the data?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=151335>

To group a Pandas dataframe by two columns and aggregate the data, you can use the `groupby` function in conjunction with the `agg` function. This allows you to group the data by specific columns and then apply a function to the grouped data to summarize or aggregate it. This can be useful for organizing and analyzing data in a more meaningful way, especially when dealing with large datasets. By grouping and aggregating by multiple columns, you can gain insights into the relationships between different variables in your dataframe.

## Pandas: Groupby Two Columns and Aggregate

You can use the following basic syntax with the `groupby()` function in pandas to group by two columns and aggregate another column:

```
df.groupby().mean()
```

This particular example groups the DataFrame by the `var1` and `var2` columns, then calculates the mean of the `var3` column.

The following examples show how to group by two columns and aggregate using the following pandas DataFrame:

```
import pandas as pd
```

```
#create DataFrame
```

```
df = pd.DataFrame({'team': ,  
'position': ,
```

```
'points': })
```

```
#view DataFrame
```

```
print(df)
```

```
team position points
```

```
0 A G 15
```

```
1 A G 22
```

```
2 A F 24
```

```
3 A F 25
```

```
4 A F 20
```

```
5 B G 35
```

```
6 B G 34
```

```
7 B G 19
```

```
8 B G 14
```

```
9 B F 12
```

**Example 1: Groupby Two Columns and Calculate Mean of Another Column**

**We can use the following syntax to calculate the mean value of the points column, grouped by the team and position columns:**

```
#calculate mean of points grouped by team and
```

**position columns**

**df.groupby().mean()**

**team position**

**A F 23.0**

**G 18.5**

**B F 12.0**

**G 25.5**

**Name: points, dtype: float64**

**From the output we can see:**

**The mean points value for players on team A in position F is 23. The mean points value for players on team A in position G is 18.5.**

**And so on.**

**Example 2: Groupby Two Columns and Calculate Max of Another Column**

**We can use the following syntax to calculate the max value of the points column, grouped by the team and position columns:**

**#calculate max of points grouped by team and position**

**columns**

**df.groupby().max()**

**team position**

**A F 25**

**G 22**

**B F 12**

**G 35**

**Name: points, dtype: int64**

**From the output we can see:**

**The max points value for players on team A in position F is 25. The max points value for players on team A in position G is 22.**

**And so on.**

**Example 3: Groupby Two Columns and Count Occurrences**

**We can use the following syntax to count the occurrences of each combination of the team and position columns:**

**#count occurrences of each combination of team and position columns**

```
df.groupby().size()
```

```
team position
```

```
A F 3
```

```
G 2
```

```
B F 1
```

```
G 4
```

```
dtype: int64
```

From the output we can see:

There are 3 players on team A in position F. There are 2 players on team A in position G.

And so on.

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