

How can I add a filter to a pivot table in Pandas?

Authored by
stats writer

June 26, 2024

RECOMMENDED CITATION

stats writer (2024). *How can I add a filter to a pivot table in Pandas?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=153658>

To add a filter to a pivot table in Pandas, you can use the "pivot_table" function and specify the desired columns and aggregation functions. Then, use the "query" function to specify the conditions for the filter. This will allow you to filter the data in the pivot table based on specific criteria, enabling you to analyze and visualize the data more effectively. Additionally, you can also use the "pivot_table" function's "margins" parameter to include summary rows and columns in the pivot table. Overall, adding a filter to a pivot table in Pandas allows for more precise data analysis and reporting.

Pandas: Add Filter to Pivot Table

You can use the following basic syntax to add a filtering condition to a pandas pivot table:

```
df.pivot_table(index='col1', values=, aggfunc='sum')
```

This particular example creates a pivot table that displays the sum of values in col2 and col3, grouped by col1.

The filter before the pivot_table() function specifies that we only want to include rows where the value in col1 of the original DataFrame has a value of 'A'.

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

Example: How to Add Filter to Pandas Pivot Table

Suppose we have the following pandas DataFrame that

contains information about various basketball players:

```
import pandas as pd
```

```
#create DataFrame
```

```
df = pd.DataFrame({'team': ,  
'points': ,  
'assists': })
```

```
#view DataFrame
```

```
print(df)
```

```
team points assists
```

```
0 A 4 2
```

```
1 A 4 2
```

```
2 A 2 5
```

```
3 A 8 5
```

```
4 B 9 4
```

```
5 B 5 7
```

```
6 B 5 5
```

```
7 B 7 3
```

```
8 C 8 9
```

```
9 C 8 8
```

```
10 C 4 4
```

```
11 C 3 4
```

We can use the following code to create a pivot table in pandas that shows the sum of the values in the points and assists columns grouped by team only for the rows where the original DataFrame has a value in the team column equal to 'A':

```
#create pivot table for rows where team is equal to 'A'  
df.pivot_table(index='team', values=,  
aggfunc='sum')
```

```
assists points
```

```
team
```

```
A 14 18
```

Notice that the pivot table only summarizes the values in the points and assists columns for the rows where the team is equal to 'A'.

You can also use the operators & and | to apply a filter that uses "AND" or "OR" logic.

For example, we can use the following syntax to create a pivot table that filters for rows where the value in the team column of the original DataFrame is equal to 'A' or 'B':

#create pivot table for rows where team is equal to 'A' or 'B'

```
df.pivot_table(index='team',  
values=,  
aggfunc='sum')
```

assists points

team

A 14 18

B 19 26

Notice that the pivot table only summarizes the values in the points and assists columns for the rows where the team is equal to 'A' or 'B'.

Note: You can find the complete documentation for the pandas `pivot_table()` function .