

How can column names be modified in a pivot table using Pandas?

Authored by
stats writer

June 28, 2024

RECOMMENDED CITATION

stats writer (2024). *How can column names be modified in a pivot table using Pandas?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=156431>

Pandas is a popular data analysis library in Python that allows for efficient manipulation and analysis of tabular data. One of its powerful features is the ability to create pivot tables, which are a useful way to summarize and analyze data. In a pivot table, the column names often represent the variables being analyzed, and it may be necessary to modify these names for clarity or consistency. This can be easily achieved using Pandas by accessing the column names and using built-in methods to rename them. By using the appropriate syntax and specifying the desired changes, the column names in a pivot table can be modified to suit the user's needs. This flexibility makes Pandas a valuable tool for data analysis and manipulation.

Pandas: Modify Column Names in Pivot Table

Often you may want to modify or format the column names in a pandas pivot table in a specific way.

Fortunately this is easy to do using built-in functions in pandas.

The following example shows how to do so.

Example: Modify Column Names in 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': ,  
'position': ,
```

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

```
#view DataFrame
```

```
print(df)
```

```
team position points
```

```
0 A G 4
```

```
1 A G 4
```

```
2 A F 6
```

```
3 A C 8
```

```
4 B G 9
```

```
5 B C 5
```

```
6 B F 5
```

```
7 B F 12
```

We can use the following code to create a pivot table in pandas that shows the mean value of points for each team and position in the DataFrame:

```
#create pivot table
```

```
piv = pd.pivot_table(df, values='points', index='team',  
columns='position')
```

```
#view pivot table
```

```
print(piv)
```

position C F G

team

A 8.0 6.0 4.0

B 5.0 8.5 9.0

Now suppose we would like to get rid of the word position in the pivot table and remove the extra team row from the pivot table.

We can use the following syntax to do so:

```
#format column names
```

```
piv.columns =
```

```
#reset index
```

```
piv.reset_index(inplace=True)
```

```
#view updated pivot table
```

```
print(piv)
```

team C F G

0 A 8.0 6.0 4.0

1 B 5.0 8.5 9.0

Notice that we were able to get rid of the word position

in the pivot table and remove the extra team row from the pivot table.

Also note that this general solution will work for a pivot table with a MultiIndex as well.

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

Additional Resources

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