

How can I rename only the last column in a Pandas DataFrame?

Authored by
stats writer

June 25, 2024

RECOMMENDED CITATION

stats writer (2024). *How can I rename only the last column in a Pandas DataFrame?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=152199>

To rename only the last column in a Pandas DataFrame, you can use the "rename" function and specify the column name and new name using the "columns" parameter. This will allow you to change the name of the last column without affecting any other columns in the DataFrame. Additionally, you can use the "inplace" parameter to make the changes directly to the original DataFrame. This method is useful for organizing and labeling data in a more meaningful way.

Pandas: Rename Only the Last Column in DataFrame

You can use the following basic syntax to rename only the last column in a pandas DataFrame:

```
df.columns = , 'new_name']
```

This particular example renames the last column new_name in a pandas DataFrame called df.

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

Example: Rename Only the Last Column in Pandas

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': ,  
'rebounds': })
```

```
#view DataFrame  
print(df)
```

```
team points assists rebounds
```

```
0 A 18 5 11
```

```
1 B 22 7 8
```

```
2 C 19 7 10
```

```
3 D 14 9 6
```

```
4 E 14 12 6
```

```
5 F 11 9 5
```

```
6 G 20 9 9
```

```
7 H 28 4 12
```

Currently the last column in the DataFrame is named rebounds.

We can use the following syntax to rename this column to rebs:

```
#rename last column to 'rebs'  
df.columns = , 'rebs']
```

```
#view updated DataFrame  
print(df)
```

```
team points assists rebs  
0 A 18 5 11  
1 B 22 7 8  
2 C 19 7 10  
3 D 14 9 6  
4 E 14 12 6  
5 F 11 9 5  
6 G 20 9 9  
7 H 28 4 12
```

Notice that the last column has been renamed to rebs and all other columns have remained unchanged.

We can also use the following syntax to view a list of all of the column names in the DataFrame:

```
#view column names  
print(df.columns)  
  
Index(, dtype='object')
```

We can see that the last column has indeed been

renamed to rebs.

The benefit of using this syntax is that we don't need to know ahead of time how many columns are in the DataFrame.

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

ARABPSYCHOLOGY.COM