How can I rename columns in Pandas using a dictionary?

Authored by

stats writer

June 27, 2024

Pandas is a popular data manipulation and analysis library in Python that provides various functions for working with tabular data. One of the common tasks in data analysis is to rename columns in a DataFrame. This can be easily achieved using a dictionary in Pandas. By creating a dictionary with the old column names as keys and the desired new names as values, the "rename" function in Pandas can be used to map the old names to the new names. This approach allows for a quick and efficient way to rename multiple columns at once, making data analysis and manipulation more streamlined and organized.

Pandas: Rename Columns with Dictionary

You can use the following basic syntax to rename columns with a dictionary in pandas:

```
#define dictionary
some_dict = {'old_col1': 'new_col1',
'old_col2': 'new_col2',
'old_col3': 'new_col3'}

#rename columns in DataFrame using dictionary
df.rename(columns=some_dict, inplace=True)
```

Note: We must specify inplace=True to modify the column names of the original DataFrame.

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

Example: Rename Columns in Pandas with Dictionary

Suppose we have the following pandas DataFrame:

import pandas as pd

#create DataFrame
df = pd.DataFrame({'rebounds': ,
'points': ,
'assists': })

#view DataFrame
print(df)

rebounds points assists
0 10 30 5
1 14 22 6
2 14 19 6
3 13 14 5
4 13 14 8
5 12 11 7
6 10 20 7
7 7 28 9

We can use the following syntax to rename each of the

columns in the DataFrame using a dictionary:

```
#define dictionary with new column names
some_dict = {'rebounds': 'rebs',
'points': 'pts',
'assists': 'ast'}

#rename columns in DataFrame using dictionary
df.rename(columns=some_dict, inplace=True)

#view updated DataFrame
print(df)
```

```
rebs pts ast
0 10 30 5
1 14 22 6
2 14 19 6
3 13 14 5
4 13 14 8
5 12 11 7
6 10 20 7
7 7 28 9
```

Notice that each of the columns have been renamed based on the values we specified in the dictionary.

It's worth noting that you don't have to rename every single column using a dictionary.

For example, we could create a dictionary to only rename the points and assists columns in the DataFrame:

```
#define dictionary with new column names for points and assists only
some_dict = {'points': 'pts',
'assists': 'ast'}

#rename columns in DataFrame using dictionary
df.rename(columns=some_dict, inplace=True)

#view updated DataFrame
print(df)
```

```
rebounds pts ast
0 10 30 5
1 14 22 6
2 14 19 6
3 13 14 5
4 13 14 8
5 12 11 7
```

6 10 20 7

7 7 28 9

Only the points and assists columns were renamed.

Since the rebounds column was not included in the dictionary, it was not renamed in the DataFrame.

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