How do I reset an index in a Pandas DataFrame?

Authored by

stats writer

May 12, 2024

Resetting an index in a Pandas DataFrame is a process of setting a new index for the rows in the dataset. This can be done by using the "reset_index()" method in Pandas, which will reassign the row labels starting from 0. This can be useful when the current index is not meaningful or if the data has been sorted or filtered. The reset index function can also be used to remove the current index and replace it with a new one. This allows for better organization and manipulation of the data in the DataFrame.

Reset an Index in Pandas DataFrame (With Examples)

You can use the following syntax to reset an index in a pandas DataFrame:

df.reset_index(drop=True, inplace=True)

Note the following arguments:

drop: Specifying True prevents pandas from saving the original index as a column in the DataFrame.inplace: Specifying True allows pandas to replace the index in the original DataFrame instead of creating a copy of the DataFrame.

The following examples show how to use this sytnax in practice.

Example 1: Reset Index & Drop Old Index

Suppose we have the following pandas DataFrame:

```
import pandas as pd

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

#view DataFrame
print(df)

points assists rebounds
0 25 5 11
4 12 7 8
3 15 7 10
5 14 9 6
2 19 12 6
1 23 9 5
7 25 9 9
6 29 4 12
```

The following code shows how to reset the index of the DataFrame and drop the old index completely:

```
#reset index
```

df.reset_index(drop=True, inplace=True)

#view updated DataFrame
print(df)

points assists rebounds
0 25 5 11
1 12 7 8
2 15 7 10
3 14 9 6
4 19 12 6
5 23 9 5
6 25 9 9
7 29 4 12

Notice that the index has been reset and the values in the index now range from 0 to 7.

Example 2: Reset Index & Retain Old Index as Column

Suppose we have the following pandas DataFrame:

import pandas as pd

#define DataFrame
df = pd.DataFrame({'points': ,

'assists': ,
'rebounds': },
index=)

#view DataFrame
print(df)

points assists rebounds
A 25 5 11
C 12 7 8
D 15 7 10
B 14 9 6
E 19 12 6
G 23 9 5
F 25 9 9
H 29 4 12

The following code shows how to reset the index of the DataFrame and retain the old index as a column in the DataFrame:

#reset index and retain old index as a column
df.reset_index(inplace=True)

#view updated DataFrame

```
print(df)

  index points assists rebounds
0     A     25       5       11
1     C     12       7        8
2     D     15       7       10
3     B     14       9        6
4     E     19      12        6
5     G     23       9        5
6     F     25       9        9
7     H     29       4       12
```

Notice that the index has been reset and the values in the index now range from 0 to 7.

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