

How can I concatenate two Pandas DataFrames?

Authored by
stats writer

July 2, 2024

RECOMMENDED CITATION

stats writer (2024). *How can I concatenate two Pandas DataFrames?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=165637>

Concatenating two Pandas DataFrames refers to the process of merging or combining two separate DataFrames into one. This can be achieved by using the "concat" function in Pandas, which allows for the joining of DataFrames either vertically or horizontally based on their indexes. This can be useful for organizing and analyzing large datasets, as it allows for the combination of multiple sources of data into a single, comprehensive DataFrame. Additionally, concatenating DataFrames can aid in data manipulation and visualization, making it an essential tool for data analysts and scientists.

Concatenate Two Pandas DataFrames (With Examples)

You can use the following basic syntax to concatenate two pandas DataFrames:

```
df3 = pd.concat(, ignore_index=True)
```

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

Example: How to Concatenate Two Pandas DataFrames

Suppose we have the following two pandas DataFrames:

```
import pandas as pd
```

```
#define DataFrames
```

```
df1 = pd.DataFrame({'team': ,  
'assists': ,
```

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

```
df2 = pd.DataFrame({'team': ,
```

```
'assists': ,
```

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

```
#view DataFrames
```

```
print(df1)
```

```
team assists points
```

```
0 A 5 11
```

```
1 A 7 8
```

```
2 A 7 10
```

```
3 A 9 6
```

```
print(df2)
```

```
team assists points
```

```
0 B 4 14
```

```
1 B 4 11
```

```
2 B 3 7
```

```
3 B 7 6
```

We can use the following syntax to concatenate the two DataFrames:

```
#concatenate the DataFrames
```

```
df3 = pd.concat()
```

```
#view resulting DataFrame
```

```
print(df3)
```

```
team assists points
```

```
0 A 5 11
```

```
1 A 7 8
```

```
2 A 7 10
```

```
3 A 9 6
```

```
0 B 4 14
```

```
1 B 4 11
```

```
2 B 3 7
```

```
3 B 7 6
```

The result is one DataFrame that contains the data from both DataFrames.

If you'd like to create a new index when concatenating the DataFrames, you must use the `ignore_index` argument:

```
#concatenate the DataFrames and ignore index
```

```
df3 = pd.concat(, ignore_index=True)
```

```
#view resulting DataFrame  
print(df3)
```

```
team assists points
```

```
0 A 5 11
```

```
1 A 7 8
```

```
2 A 7 10
```

```
3 A 9 6
```

```
4 B 4 14
```

```
5 B 4 11
```

```
6 B 3 7
```

```
7 B 7 6
```

Notice that the index of the resulting DataFrame ranges from 0 to 7.

Note #1: In this example we concatenated two pandas DataFrames, but you can use this exact syntax to concatenate any number of DataFrames that you'd like.

Note #2: You can find the complete documentation for the pandas concat() function .

Additional Resources

The following tutorials explain how to perform other

common operations in pandas:

ARABPSYCHOLOGY.COM