

How can I merge multiple CSV files in pandas, with an example?

Authored by
stats writer

June 25, 2024

RECOMMENDED CITATION

stats writer (2024). *How can I merge multiple CSV files in pandas, with an example?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=152093>

Pandas is a Python library that provides powerful tools for data manipulation and analysis. One useful feature of pandas is the ability to merge multiple CSV (Comma Separated Values) files into one dataframe. This allows for the consolidation of data from multiple sources, making it easier to analyze and manipulate.

To merge multiple CSV files in pandas, first import the necessary libraries and read the CSV files using the `read_csv` function. Then, use the `merge` function to join the dataframes based on a common column or index.

For example, let's say we have three CSV files with the following data:

- File 1: Employee ID, Name, Department
- File 2: Employee ID, Salary
- File 3: Employee ID, Address, Phone Number

We can merge these files using the following code:

```
import pandas as pd

#read csv files
df1 = pd.read_csv("file1.csv")
df2 = pd.read_csv("file2.csv")
df3 = pd.read_csv("file3.csv")

#merge dataframes
merged_df = pd.merge(df1, df2, on="Employee ID")
merged_df = pd.merge(merged_df, df3, on="Employee ID")
```

This will result in a new dataframe with all the data from the three CSV files merged together. The resulting dataframe can then be further manipulated or exported to a new CSV file.

In conclusion, merging multiple CSV files in pandas can be done easily and efficiently, providing a convenient way to handle large amounts of data.

Merge Multiple CSV Files in Pandas (With Example)

You can use the following basic syntax to merge multiple CSV files located in the same folder into a pandas DataFrame:

```
import pandas as pd
import glob
import os

#define path to CSV files
path = r'C:\Users\bob\Documents\my_data_files'

#identify all CSV files
all_files = glob.glob(os.path.join("*".csv"))

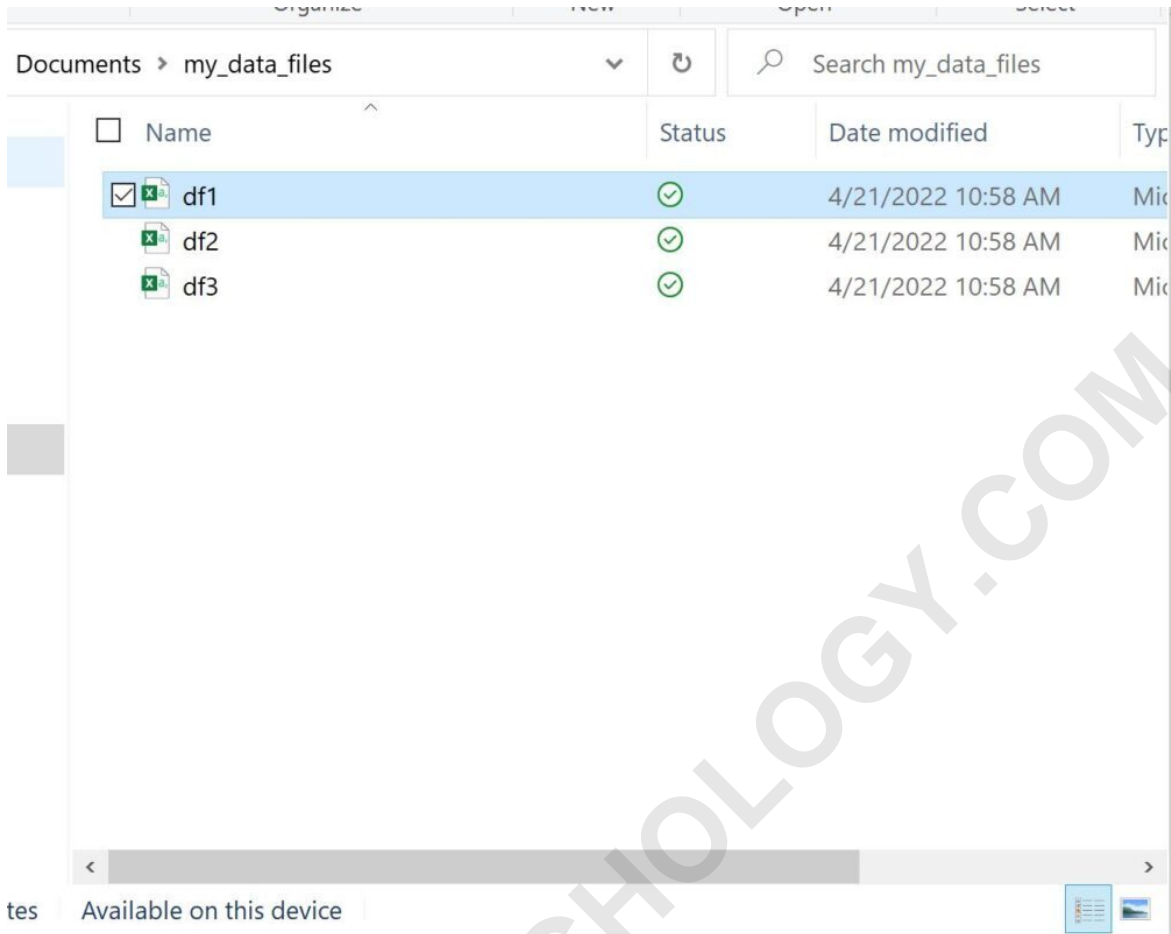
#merge all CSV files into one DataFrame
df = pd.concat((pd.read_csv(f) for f in all_files),
ignore_index=True)
```

This particular example will merge all of the CSV files located in the folder called my_data_files into one pandas DataFrame.

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

Example: Merge Multiple CSV Files in Pandas

Suppose I have a folder on my computer called my_data_files that contains three CSV files:



Each CSV file contains two columns called points and assists, which represents the points and assists of various basketball players.

Here is what the first CSV called df1 looks like:

```
1 "points", "assists"  
2 4, 3  
3 5, 2  
4 5, 4  
5 6, 4  
6 8, 6  
7 9, 3  
8
```

We can use the following syntax to merge all three CSV files from the folder into one pandas DataFrame:

```
import pandas as pd  
import glob  
import os
```

```
#define path to CSV files
```

```
path = r'C:UsersbobDocumentsmy_data_files'
```

```
#identify all CSV files
```

```
all_files = glob.glob(os.path.join("*.csv"))
```

```
#merge all CSV files into one DataFrame
```

```
df = pd.concat((pd.read_csv(f) for f in all_files),  
ignore_index=True)
```

```
#view resulting DataFrameprint(df)
```

```
points assists
```

```
0 4 3
```

```
1 5 2
```

```
2 5 4
```

```
3 6 4
```

```
4 8 6
```

```
5 9 3
```

```
6 2 3
```

```
7 10 2
```

```
8 14 9
```

```
9 15 3
```

```
10 6 10
```

```
11 8 6
```

```
12 9 4
```

Notice that all three CSV files have been successfully

imported and merged into one DataFrame.

The final DataFrame contains 13 rows and 2 columns.

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

The following tutorials explain how to perform other common tasks in Python:

ARABPSYCHOLOGY.COM