How can I read a CSV file in Pandas without headers? Can you provide an example?

Authored by

stats writer

June 25, 2024

Pandas is a widely used Python library for data analysis. It offers a convenient way to read and manipulate data stored in CSV (Comma Separated Values) files. By default, Pandas assumes that the first row of a CSV file contains the column names or headers. However, there may be cases where the CSV file does not have any headers. In such cases, it is necessary to specify this while reading the file into Pandas. This can be done by setting the "header" parameter to "None" in the "read_csv()" function. This will instruct Pandas to use default column names (e.g. 0, 1, 2, etc.) for the data. An example of reading a CSV file without headers in Pandas would be:

df = pd.read_csv('file.csv', header=None)

This will create a dataframe (df) with the data from the CSV file, without any headers. From there, you can use Pandas' functions to manipulate and analyze the data as needed.

Read CSV Without Headers in Pandas (With Example)

You can use the following basic syntax to read a CSV file without headers into a pandas DataFrame:

df = pd.read_csv('my_data.csv', header=None)

The argument header=None tells pandas that the first row should not be used as the header row.

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

Example: Read CSV Without Headers in Pandas

Suppose we have the following CSV file called players_data.csv:

```
A,22,10
B,14,9
C,29,6
D,30,2
E,22,9
F,31,10
```


From the file we can see that the first row does not contain any column names.

If we import the CSV file using the read_csv() function, pandas will attempt to use the first row as a header row:

```
import pandas as pd

#import CSV file
df = pd.read_csv('players_data.csv')

#view resulting DataFrame
```

```
print(df)

A 22 10
0 B 14 9
1 C 29 6
2 D 30 2
3 E 22 9
4 F 31 10
```

However, we can specify header=None so that pandas knows not to use the first row as a header row:

```
import pandas as pd

#import CSV file without header
df = pd.read_csv('players_data.csv', header=None)

#view resulting DataFrame
print(df)

0 1 2
0 A 22 10
1 B 14 9
2 C 29 6
3 D 30 2
```

4 E 22 9
5 F 31 10

Notice that the first row in the CSV file is no longer used as the header row.

Also notice that pandas uses a range of numerical values (0, 1, 2) by default as the column names.

To specify your own column names when importing the CSV file, you can use the names argument as follows:

```
import pandas as pd

#specify column names
cols =

#import CSV file without header and specify column names
df = pd.read_csv('players_data.csv', header=None, names=cols)

#view resulting DataFrame
print(df)

team points rebounds
```

0 A 22 10

1 B 14 9

2 C 29 6

3 D 30 2

4 E 22 9

5 F 31 10

The DataFrame now has the column names that we specified using the names argument.

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