

How can I set the first row as the header in Pandas?

Authored by
stats writer

June 27, 2024

RECOMMENDED CITATION

stats writer (2024). *How can I set the first row as the header in Pandas?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=155043>

To set the first row as the header in Pandas, the "header" parameter can be used in the `read_csv()` function. This parameter takes in an integer or a list of integers representing the rows to be used as the column names. By setting this parameter to 0, the first row of the dataset will be used as the header. This allows for easier data manipulation and analysis as the column names will accurately reflect the data in each column. This feature is particularly useful when working with large datasets where manually specifying column names may be time-consuming and prone to errors.

Set First Row as Header in Pandas

You can use the following basic syntax to set the first row of a pandas DataFrame as the header:

```
df.columns = df.iloc  
df = df
```

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

Example: Set First Row as Header in Pandas

Suppose we have the following pandas DataFrame that contains information about various basketball players:

```
import pandas as pd  
  
#create DataFrame  
df = pd.DataFrame({'Bad Name 1': ,  
'Bad Name 2': ,
```

```
'Bad Name 3': ,  
'Bad Name 4': })
```

```
#view DataFrame  
print(df)
```

```
Bad Name 1 Bad Name 2 Bad Name 3 Bad Name 4  
0 team points assists rebounds  
1 A 18 5 11  
2 B 22 7 8  
3 C 19 7 10  
4 D 14 9 6  
5 E 14 12 6  
6 F 11 9 5  
7 G 20 9 9  
8 H 28 4 12
```

Suppose the first row contains the values that we actually want to use in the header.

To set the first row as the header, we can use the following syntax:

```
#set column names equal to values in row index  
position 0
```

```
df.columns = df.iloc
```

```
#remove first row from DataFrame
```

```
df = df
```

```
#view updated DataFrameprint(df)
```

```
0 team points assists rebounds
```

```
1 A 18 5 11
```

```
2 B 22 7 8
```

```
3 C 19 7 10
```

```
4 D 14 9 6
```

```
5 E 14 12 6
```

```
6 F 11 9 5
```

```
7 G 20 9 9
```

```
8 H 28 4 12
```

Notice that the values in the first row are now used as the header.

If you'd like to reset the index of the DataFrame, use the following code:

```
#reset index values
```

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

#view updated DataFrameprint(df)

0 team points assists rebounds

0 A 18 5 11

1 B 22 7 8

2 C 19 7 10

3 D 14 9 6

4 E 14 12 6

5 F 11 9 5

6 G 20 9 9

7 H 28 4 12

The index is now reset so that the first row has an index value of 0.

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