

# How can I skip rows when reading a CSV file in Pandas?

Authored by  
**stats writer**

June 27, 2024

## RECOMMENDED CITATION

stats writer (2024). *How can I skip rows when reading a CSV file in Pandas?*.  
PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=155178>

Pandas is a popular Python library used for data analysis and manipulation. It has a built-in function for reading CSV files, which is a common file format used for storing tabular data. One common issue that users may encounter when reading CSV files in Pandas is the need to skip certain rows in the file. This can be easily achieved by using the "skiprows" parameter in the "read\_csv" function. By specifying the row numbers or a list of row numbers to be skipped, users can effectively exclude these rows from the data frame that is created from the CSV file. This allows for a more efficient and accurate analysis of the data, without having to manually remove the unwanted rows. Overall, the ability to skip rows when reading a CSV file in Pandas is a useful feature that enhances the functionality and flexibility of the library for data analysis tasks.

## **Pandas: Skip Rows when Reading CSV File**

**You can use the following methods to skip rows when reading a CSV file into a pandas DataFrame:**

### **Method 1: Skip One Specific Row**

```
#import DataFrame and skip 2nd row  
df = pd.read_csv('my_data.csv', skiprows=)
```

### **Method 2: Skip Several Specific Rows**

```
#import DataFrame and skip 2nd and 4th row  
df = pd.read_csv('my_data.csv', skiprows=)
```

### **Method 3: Skip First N Rows**

```
#import DataFrame and skip first 2 rows
```

```
df = pd.read_csv('my_data.csv', skiprows=2)
```

The following examples show how to use each method in practice with the following CSV file called `basketball_data.csv`:

```
1 team,points,rebounds
2 A, 22, 10
3 B, 14, 9
4 C, 29, 6
5 D, 30, 2
```

### Example 1: Skip One Specific Row

We can use the following code to import the CSV file and skip the second row:

```
import pandas as pd
```

```
#import DataFrame and skip 2nd row
```

```
df = pd.read_csv('basketball_data.csv', skiprows=)
```

```
#view DataFrame
```

```
df
```

```
team points rebounds
```

```
0 A 22 10
```

```
1 C 29 6
```

```
2 D 30 2
```

**Notice that the second row (with team 'B') was skipped when importing the CSV file into the pandas DataFrame.**

**Note: The first row in the CSV file is considered to be row 0.**

**Example 2: Skip Several Specific Rows**

**We can use the following code to import the CSV file and skip the second and fourth rows:**

```
import pandas as pd
```

```
#import DataFrame and skip 2nd and 4th rows
```

```
df = pd.read_csv('basketball_data.csv', skiprows=)
```

```
#view DataFrame
```

```
df
```

**team points rebounds**

**0 A 22 10**

**1 C 29 6**

### Example 3: Skip First N Rows

We can use the following code to import the CSV file and skip the first two rows:

```
import pandas as pd
```

```
#import DataFrame and skip first 2 rows
```

```
df = pd.read_csv('basketball_data.csv', skiprows=2)
```

```
#view DataFrame
```

```
df
```

```
B 14 9
```

```
0 C 29 6
```

```
1 D 30 2
```

Notice that the first two rows in the CSV file were skipped and the next available row (with team 'B') became the header row for the DataFrame.

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

**[How to Export NumPy Array to CSV File](#)**

ARABPSYCHOLOGY.COM