

# How can specific rows be summed in Pandas, and what are some examples of doing so?

Authored by  
**stats writer**

June 26, 2024

## RECOMMENDED CITATION

stats writer (2024). *How can specific rows be summed in Pandas, and what are some examples of doing so?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=154084>

Pandas is a popular library in Python used for data manipulation and analysis. One of its useful functions is the ability to sum specific rows in a dataset. This can be done by using the "sum" method and specifying the rows to be summed as parameters. For example, if we have a dataset of students' grades in different subjects, we can use the sum method to calculate the total grade for each student by specifying the rows containing the grades for each subject. Another example is when working with financial data, we can use the sum method to calculate the total revenue or expenses for a specific time period by selecting the rows containing the relevant data. In summary, Pandas allows for efficient and straightforward summation of specific rows in a dataset, making it a useful tool for various data analysis tasks.

## Sum Specific Rows in Pandas (With Examples)

You can use the following methods to find the sum of specific rows in a pandas DataFrame:

### Method 1: Sum Specific Rows by Index

```
#sum rows in index positions 0, 1, and 4  
df.iloc].sum()
```

### Method 2: Sum Specific Rows by Label

```
#sum rows with index labels 'A', 'B', and 'E'  
df.loc].sum()
```

The following examples show how to use each method in practice with the following pandas DataFrame:

```
import pandas as pd
```

```
#create DataFrame
```

```
df = pd.DataFrame({'points': ,  
'rebounds': ,  
'assists': })
```

```
#set index
```

```
df = df.set_index())
```

```
#view DataFrame
```

```
print(df)
```

```
points rebounds assists
```

```
A 28 5 10
```

```
B 17 6 13
```

```
C 19 4 7
```

```
D 14 7 8
```

```
E 23 14 4
```

```
F 26 12 5
```

```
G 5 9 8
```

**Example 1: Sum Specific Rows by Index**

**The following code shows how to sum the values in the rows with index values 0, 1, and 4 for each column in the DataFrame:**

```
#sum rows in index positions 0, 1, and 4  
df.iloc].sum()
```

```
points 68  
rebounds 25  
assists 27  
dtype: int64
```

From the output we can see:

The sum of rows with index values 0, 1, and 4 for the points column is 68. The sum of rows with index values 0, 1, and 4 for the rebounds column is 25. The sum of rows with index values 0, 1, and 4 for the assists column is 27.

Also note that you can sum a specific range of rows by using the following syntax:

```
#sum rows in index positions between 0 and 4  
df.iloc.sum()
```

```
points 78  
rebounds 22  
assists 38
```

**dtype: int64**

From the output we can see the sum of the rows with index values between 0 and 4 (not including 4) for each of the columns in the DataFrame.

**Example 2: Sum Specific Rows by Label**

The following code shows how to sum the values in the rows with index labels 'A', 'B', and 'E' for each column in the DataFrame:

```
#sum rows with index labels 'A', 'B', and 'E'  
df.loc].sum()  
  
points 68  
rebounds 25  
assists 27  
dtype: int64
```

From the output we can see:

The sum of rows with index values 'A', 'B', and 'E' for the points column is 68. The sum of rows with index values 'A', 'B', and 'E' for the rebounds column is 25. The

**sum of rows with index values 'A', 'B', and 'E' for the assists column is 27.**

**Related:**

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

ARABPSYCHOLOGY.COM