

How can I use a combination of GroupBy and Sort functions in Pandas to arrange data within groups?

Authored by
stats writer

June 29, 2024

RECOMMENDED CITATION

stats writer (2024). *How can I use a combination of GroupBy and Sort functions in Pandas to arrange data within groups?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=159539>

Pandas is a powerful library in Python commonly used for data manipulation and analysis. One useful feature in Pandas is the ability to group data based on certain criteria using the GroupBy function. This allows for easy organization and aggregation of data within the specified groups. Additionally, the Sort function in Pandas allows for arranging the data in a specified order. By combining these two functions, one can effectively arrange data within groups, providing a better understanding and analysis of the data. This can be particularly useful in scenarios where there are multiple groups with large amounts of data, allowing for a more structured and organized approach to data analysis.

Pandas: Use GroupBy & Sort Within Groups

You can use the following syntax to group rows in a pandas DataFrame and then sort the values within groups:

```
df.sort_values(,ascending=False).groupby('var1').head()
```

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

Example: Use GroupBy & Sort Within Groups in Pandas

Suppose we have the following pandas DataFrame that shows the sales made at two different store locations:

```
import pandas as pd
```

```
#create DataFrame
```

```
df = pd.DataFrame({'store': ,
```

```
'sales': })
```

```
#view DataFrame
```

```
print(df)
```

```
store sales
```

```
0 B 12
```

```
1 B 25
```

```
2 A 8
```

```
3 A 14
```

```
4 B 10
```

```
5 B 20
```

```
6 A 30
```

```
7 A 30
```

We can use the following syntax to group the rows by the store column and sort in descending order based on the sales column:

```
#group by store and sort by sales values in descending order
```

```
df.sort_values(ascending=False).groupby('store').head( )
```

```
store sales
```

1 B 25

5 B 20

0 B 12

4 B 10

6 A 30

7 A 30

3 A 14

2 A 8

Note that we could also drop the `ascending=False` argument to sort the sales values in ascending order:

#group by store and sort by sales values in ascending order

```
df.sort_values().groupby('store').head()
```

store sales

2 A 8

3 A 14

6 A 30

7 A 30

4 B 10

0 B 12

5 B 20

1 B 25

Note that the `head()` function only displays the first 5 values by group.

To display the top n values by group, simply use `head(n)` instead.

Note: You can find the complete documentation for the GroupBy operation in pandas .

Additional Resources

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