How to group by 5-Minute Intervals in Pandas

Authored by

stats writer

November 22, 2025

Pandas has a resample() function that can be used to group observations into 5-minute intervals. This function takes the dataframe as an argument and the desired interval as an argument, and returns an object representing the data grouped by the given interval. This object can then be used to get the aggregated values for each interval. To get the time intervals, the index of the resulting object is used. The object can also be manipulated with other Pandas functions.

You can use the following basic syntax to group rows by 5-minute intervals in a pandas DataFrame:

df.resample('5min').sum()

This particular formula assumes that the index of your DataFrame contains datetime values and it calculates the sum of every column in the DataFrame, grouped by 5-minute intervals.

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

Example: How to Group by 5-Minute Intervals in Pandas

Suppose we have the following pandas DataFrame that shows the sales made by some company on various dates and times:

import pandas as pd

#create DataFrame
df = pd.DataFrame({'date': pd.date_range(start='1/1/2020', freq='min', periods=12),
'sales': ,
'returns': })

#set 'date' column as index
df = df.set_index('date')

#view DataFrame
print(df)

sales returns
date
2020-01-01 00:00:00 6 0
2020-01-01 00:01:00 8 3
2020-01-01 00:02:00 9 2
2020-01-01 00:03:00 11 2

2020-01-01 00:04:00 13 1
2020-01-01 00:05:00 8 3
2020-01-01 00:06:00 8 2
2020-01-01 00:07:00 15 4
2020-01-01 00:08:00 22 1
2020-01-01 00:09:00 9 5
2020-01-01 00:10:00 8 3
2020-01-01 00:11:00 4 2

We can use the following syntax to calculate the sum of sales grouped by 5-minute intervals:

#calculate sum of sales and returns grouped by 5-minute intervals
df.resample('5min').sum()

sales returns
date
2020-01-01 00:00:00 47 8
2020-01-01 00:05:00 62 15
2020-01-01 00:10:00 12 5

Here's how to interpret the output:

Total sales during minutes 0-4 was **47** and total returns was **8**.
Total sales during minutes 5-9 was **62** and total returns was **15**.
Total sales during minutes 10-14 was **1 2**and total returns was **5**.

We can use similar syntax to calculate the max of the sales values and returns values, grouped by 5-minute intervals :

#calculate max of sales and max of returns grouped by 5-minute intervals
df.resample('5min').max()

sales returns
date
2020-01-01 00:00:00 13 3
2020-01-01 00:05:00 22 5
2020-01-01 00:10:00 8 3

We can use similar syntax to calculate any value we'd like grouped by 5-minute intervals.