How can I add or subtract time to a datetime object in Pandas?

Authored by

stats writer

June 26, 2024

Pandas is a popular library in Python used for data manipulation and analysis. One of its useful functions is the ability to add or subtract time to a datetime object. This can be achieved by using the built-in functions provided by Pandas, such as the "timedelta" function, which allows for the addition or subtraction of specific time units (e.g. days, hours, minutes) to a given datetime object. This feature is particularly helpful for time series data analysis, where the need to manipulate and adjust datetime values is common. The process of adding or subtracting time to a datetime object in Pandas is efficient and straightforward, making it a valuable tool for data handling and manipulation tasks.

Pandas: Add/Subtract Time to Datetime

You can use the following basic syntax to add or subtract time to a datetime in pandas:

```
#add time to datetime
df = df + pd.Timedelta(hours=5, minutes=10, seconds=3)

#subtract time from datetime
df = df - pd.Timedelta(hours=5, minutes=10, seconds=3)
```

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

Example: Add/Subtract Time to Datetime in Pandas

Suppose we have the following pandas DataFrame that shows the sales made by some store during 10 different datetimes:

```
import pandas as pd

#create DataFrame
df = pd.DataFrame({'time': pd.date_range('2022-01-01',
periods=10),
'sales': })

#view DataFrame
print(df)

time sales
0 2022-01-01 14
1 2022-01-02 22
2 2022-01-03 25
3 2022-01-04 29
4 2022-01-05 31
5 2022-01-06 10
6 2022-01-07 12
7 2022-01-08 8
8 2022-01-09 22
9 2022-01-10 25
```

We can use the pandas Timedelta function to add 5 hours, 10 minutes, and 3 seconds to each datetime value in the "time" column:

#create new column that contains time + 5 hours, 10 minutes, 3 seconds
df = df + pd.Timedelta(hours=5, minutes=10, seconds=3)

#view updated DataFrame
print(df)

time sales time_plus_some
0 2022-01-01 14 2022-01-01 05:10:03
1 2022-01-02 22 2022-01-02 05:10:03
2 2022-01-03 25 2022-01-03 05:10:03
3 2022-01-04 29 2022-01-04 05:10:03
4 2022-01-05 31 2022-01-05 05:10:03
5 2022-01-06 10 2022-01-06 05:10:03
6 2022-01-07 12 2022-01-07 05:10:03
7 2022-01-08 8 2022-01-08 05:10:03
8 2022-01-09 22 2022-01-09 05:10:03
9 2022-01-10 25 2022-01-10 05:10:03

And we can just as easily create a new column that subtracts 5 hours, 10 minutes, and 3 seconds from each datetime value in the "time" column:

#create new column that contains time - 5 hours, 10 minutes, 3 seconds

```
df = df - pd.Timedelta(hours=5, minutes=10, seconds=3)

#view updated DataFrame
print(df)

time sales time_minus_some
0 2022-01-01 14 2021-12-31 18:49:57
1 2022-01-02 22 2022-01-01 18:49:57
2 2022-01-03 25 2022-01-02 18:49:57
3 2022-01-04 29 2022-01-03 18:49:57
4 2022-01-05 31 2022-01-04 18:49:57
5 2022-01-06 10 2022-01-05 18:49:57
6 2022-01-07 12 2022-01-06 18:49:57
7 2022-01-08 8 2022-01-07 18:49:57
8 2022-01-09 22 2022-01-08 18:49:57
9 2022-01-10 25 2022-01-09 18:49:57
```

Note #1: In these examples we used a specific number of hours, minutes, and seconds, but you can also use only one of these units if you'd like. For example, you can specify pd.Timedelta(hours=5) to simply add five hours to a datetime value.

Note #2: You can find the complete documentation for the pandas Timedelta function .

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