WORKING WITH DATES AND TIMES IN PYTHON

Adding time to the mix

Max Shron

Data Scientist and Author

Dates and Times

Date

Time

October 1   2017,   3:   23: 25 PM

Dates and Times

Date   Time

October 1   2017,   3:   23: 25 PM

```python
# Import datetime
from datetime import datetime
```

Dates and Times


```
# Import datetime
from datetime import datetime

dt = datetime(
```

Dates and Times


```
# Import datetime
from datetime import datetime

dt = datetime(2017, 10, 1
```

Dates and Times

Date   Time

October 1   2017,   3: 23: 25 PM

```python
# Import datetime
from datetime import datetime

dt = datetime(2017, 10, 1, 15
```

Dates and Times


```
# Import datetime
from datetime import datetime

dt = datetime(2017, 10, 1, 15, 23,
```

Dates and Times


```
# Import datetime
from datetime import datetime

dt = datetime(2017, 10, 1, 15, 23, 25)
```

Dates and Times


```
# Import datetime
from datetime import datetime

dt = datetime(2017, 10, 1, 15, 23, 25, 500000)
```

Dates and Times


```
# Import datetime
from datetime import datetime

dt = datetime(year=2017, month=10, day=1,
              hour=15, minute=23, second=25,
              microsecond=500000)
```

Replacing parts of a datetime

```
print(dt)
2017-10-01 15:23:25.500000

dt_hr = dt.replace(minute=0, second=0, microsecond=0)
print(dt_hr)
2017-10-01 15:00:00
```

Capital Bikeshare

Capital Bikeshare Station Installed at the Lincoln Memorial by Euan Fisk, licensed CC B 2.0

WORKING WITH DATES AND TIMES IN PYTHON

Adding time to the mix

WORKING WITH DATES AND TIMES IN PYTHON

Printing and parsing datetimes

Max Shron

Data Scientist and Author

Printing datetimes

```python
# Create datetime
dt = datetime(2017, 12, 30, 15, 19, 13)
```

Printing datetimes

```python
# Create datetime
dt = datetime(2017, 12, 30, 15, 19, 13)

print(dt.strftime("%Y-%m-%d"))
2017-12-30
```

Printing datetimes

```
# Create datetime
dt = datetime(2017, 12, 30, 15, 19, 13)

print(dt.strftime("%Y-%m-%d"))
2017-12-30

print(dt.strftime("%Y-%m-%d %H:%M:%S"))
2017-12-30 15:19:13
```

Printing datetimes

```python
# Create datetime
dt = datetime(2017, 12, 30, 15, 19, 13)

print(dt.strftime("%Y-%m-%d"))
2017-12-30

print(dt.strftime("%Y-%m-%d %H:%M:%S"))
2017-12-30 15:19:13

print(dt.strftime("%H:%M:%S on %d/%m/%Y"))
15:19:13 on 2017/12/30
```

ISO 8601 Format

```python
# Create datetime
dt = datetime(2017, 10, 1, 15, 23, 25)

# ISO 8601 format
print(dt.isoformat())
2017-12-30T15:19:13
```

Parsing datetimes with strptime

```python
# Import datetime
from datetime import datetime
```

Parsing datetimes with strptime

```python
# Import datetime
from datetime import datetime

dt = datetime.strptime(
```

Parsing datetimes with strptime

```python
# Import datetime
from datetime import datetime

dt = datetime.strptime("12/30/2017  15:19:13"
```

Parsing datetimes with strptime

```python
# Import datetime
from datetime import datetime

dt = datetime.strptime("12/30/2017 15:19:13",
                       "%m/%d/%Y %H:%M:%S")
```

Parsing datetimes with strptime

```python
# Import datetime
from datetime import datetime

dt = datetime.strptime("12/30/2017 15:19:13",
                       "%m/%d/%Y %H:%M:%S")

# What did we make?
print(type(dt))
<class 'datetime.datetime'>

print(dt)
2017-12-30 15:19:13
```

Parsing datetimes with strptime

```python
# Import datetime
from datetime import datetime

# Incorrect format string
dt = datetime.strptime("2017-12-30 15:19:13", "%Y-%m-%d")
ValueError: unconverted data remains:  15:19:13
```

Parsing datetimes with Pandas

```python
# A timestamp
ts = 1514665153.0

# Convert to datetime and print
print(datetime.fromtimestamp(ts))

2017-12-30 15:19:13
```

WORKING WITH DATES AND TIMES IN PYTHON

Printing and parsing datetimes

WORKING WITH DATES AND TIMES IN PYTHON

Working with durations

Max Shron

Data Scientist and Author

Working with durations

2017-10-08
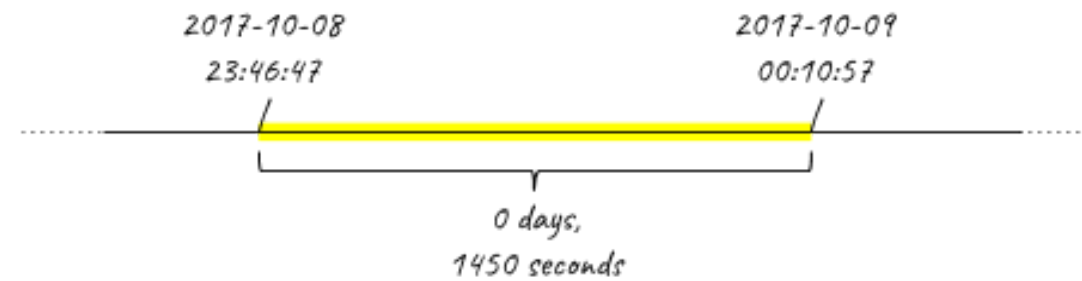23:46:47

2017-10-09
00:10:57

Working with durations


```
# Create example datetimes
start = datetime(2017, 10, 8, 23, 46, 47)
end = datetime(2017, 10, 9, 0, 10, 57)
```
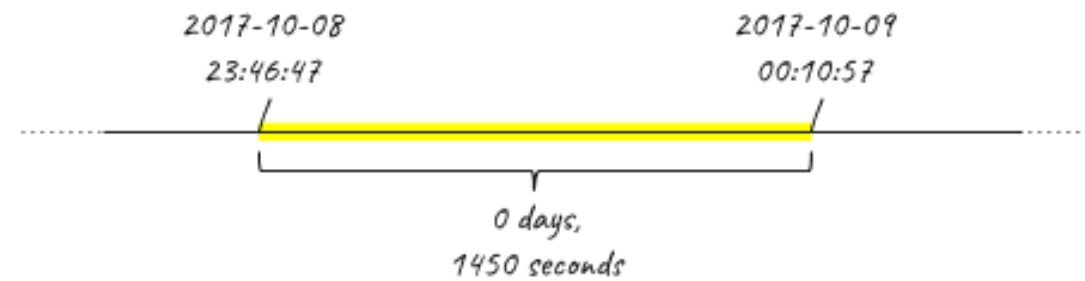
Working with durations


```python
# Create example datetimes
start = datetime(2017, 10, 8, 23, 46, 47)
end = datetime(2017, 10, 9, 0, 10, 57)

# Subtract datetimes to create a timedelta
duration = end - start
```

Working with durations


```python
# Create example datetimes
start = datetime(2017, 10, 8, 23, 46, 47)
end = datetime(2017, 10, 9, 0, 10, 57)

# Subtract datetimes to create a timedelta
duration = end - start

print(duration.total_seconds())
1450.0
```

Creating timedeltas

```python
# Import timedelta
from datetime import timedelta
```

Creating timedeltas

```python
# Import timedelta
from datetime import timedelta

# Create a timedelta
delta1 = timedelta(seconds=1)
```

Creating timedeltas

```python
# Import timedelta
from datetime import timedelta

# Create a timedelta
delta1 = timedelta(seconds=1)

print(start)
2017-10-08 23:46:47
print(start + delta1)
2017-10-08 23:46:48
```

Creating timedeltas

```python
# Import timedelta
from datetime import timedelta

# Create a timedelta
delta1 = timedelta(seconds=1)

print(start)
2017-10-08 23:46:47
print(start + delta1)
2017-10-08 23:46:48

# Create another timedelta
delta2 = timedelta(days=1, seconds=1)

print(start)
2017-10-08 23:46:47
print(start + delta2)
2017-10-09 23:46:48
```

Negative timedeltas

```python
# Import timedelta
from datetime import timedelta

# Create a negative timedelta
delta3 = timedelta(weeks=-1)

print(start)
2017-10-08 23:46:47
print(start + delta3)
2017-10-01 23:46:47
```

Negative timedeltas

```python
# Import timedelta
from datetime import timedelta

# Create a negative timedelta
delta3 = timedelta(weeks=-1)

print(start)
2017-10-08 23:46:47
print(start + delta3)
2017-10-01 23:46:47

# Same, but we'll subtract this time
delta4 = timedelta(weeks=1)

print(start)
2017-10-08 23:46:47
print(start - delta4)
2017-10-01 23:46:47
```

WORKING WITH DATES AND TIMES IN PYTHON

Working with durations