Python Datetime – Dates and Time

Python Datetime

Python datetime module can be used to work with dates and times.

In this tutorial, we will learn how to use datetime module for different operations involving date, day of the week, seconds, minutes, current time, etc.

Import Datetime Module

To import datetime module, use the following import statement.

```
import datetime
```

Get Current Time

To get current time, use now() function as shown below.

```
import datetime

#get current time
x = datetime.datetime.now()

print(x)
```

Output

```
2019-09-12 12:18:06.255164
```

The output contains the current time in the format YYYY-MM-DD HH:MM:SS.MICROS

Get Current Time

To get current time, use now() function as shown below.

```
import datetime

#get current time
x = datetime.datetime.now()

print(x)
```

Output

```
2019-09-12 12:18:06.255164
```

The output contains the current time in the format YYYY-MM-DD HH:MM:SS.MICROS which has year, month, day, hour, minute, second and micro-second respectively.

Create Date Object

In datetime module, date is an object. You can create a datetime object by passing year, month, day, hour, minute, second and micro-second as arguments.

The syntax to create a datetime object is:

```
datetime.datetime(year, month, day[, hour[, minute[, second[, microsecond[, tzinfo]]]]])
```

The parameters `year`, `month` and `day` are mandatory while `hour`, `minute`, `second`, `microsecond` and timezone information `tzinfo` are optional. The optional parameters are by default `0`.

In the following example, you can observe how the default values for the optional parameters take effect.

```
import datetime

#create datetime object with year, month, day
x = datetime.datetime(2018, 5, 24)
print(x)

#create datetime object with year, month, day, hour
x = datetime.datetime(2018, 5, 24, 5)
print(x)

#create datetime object with year, month, day, hour, minute
x = datetime.datetime(2018, 5, 24, 5, 45)
print(x)

#create datetime object with year, month, day, hour, minute, second
x = datetime.datetime(2018, 5, 24, 5, 45, 34)
print(x)

#create datetime object with year, month, day, hour, minute, second, microsecond
x = datetime.datetime(2018, 5, 24, 5, 45, 34, 542136)
print(x)
```

Output

```
2018-05-24 00:00:00
2018-05-24 05:00:00
2018-05-24 05:45:00
2018-05-24 05:45:34
2018-05-24 05:45:34.542136
```

Conclusion

In this Python Tutorial, we learned about Python datetime library.

Python Programming

- Python Tutorial
- Install Python
- Install Anaconda Python
- Python HelloWorld Program
- Python Variables
- Python Variable Data Type Conversion
- Python Comments

Control Statements

- Python If
- Python If Else
- Python While Loop
- Python For Loop

Python String

- Python String Methods
- Python String Length
- Python String Replace
- Python Split String
- Python Count Occurrences of Sub-String
- Python Sort List of Strings

Functions

- Python Functions

Python Collections

- Python List
- Python Dictionary

Advanced

- Python Multithreading

Useful Resources

- Python Interview Questions