

Object-Oriented Python: Takeaways

by Dataquest Labs, Inc. - All rights reserved © 2021

Syntax

- Define an empty class:

```
class MyClass:
    pass
```

- Instantiate an object of a class:

```
class MyClass:
    pass
mc_1 = MyClass()
```

- Define an init function in a class to assign an attribute at instantiation:

```
class MyClass:
    def __init__(self, param_1):
        self.attribute_1 = param_1
mc_2 = MyClass("arg_1")
```

- Define a method inside a class and call it on an instantiated object:

```
class MyClass:
    def __init__(self, param_1):
        self.attribute_1 = param_1
    def add_20(self):
        self.attribute_1 += 20
mc_3 = MyClass(10) # mc_3.attribute is 10
mc_3.add_20()      # mc_3.attribute is 30
```

Concepts

- In **Object-Oriented Programming**, the fundamental building blocks are objects.
 - It differs from **Procedural** programming, which executes sequential steps.
- An **object** is an entity that stores data.
- A **class** describes an object's type. It defines the following:
 - What data is stored in the object, known as attributes
 - What actions the object can do, known as methods
- An **attribute** is a variable that belongs to an instance of a class.
- A **method** is a function that belongs to an instance of a class.
- We access attributes and methods using **dot notation**. Attributes do not use parentheses, whereas methods do.
- An **instance** describes a specific example of a class. For instance, in the code `x = 3`, `x` is an instance of the type `int`.
 - We call creating an object **instantiation**.

- A **class definition** is code that defines how a class behaves, including all methods and attributes.
- The init method is a special method that runs at the moment of an object's instantiation.
 - The init method (`__init__()`) is one of a number of special methods that Python defines.
- All methods must include `self` , representing the object instance, as their first parameter.
- It is convention to start the name of any attributes or methods that we don't intend for external use with an underscore.

Resources

- [Python Documentation: Classes](#)