Sara O’Neal

June 2, 2021

Foundations of Python Programming (IT FDN 110 A)

Assignment 08

GitHub:

# Classes in Python

## Introduction

The purpose of this paper is to provide an overview of how to complete the eighth assignment in the Foundations of Python Programming course, which covers how to work with classes in Python.

## Creating Python Scripts Using Classes

The following steps outline an example of how to use classes within Python. The main purpose of using classes is to organize functions and data that are used to create new instances of an object.

1. First, understand the basic components of the starter script outlined by the pseudo-code, as shown in Figure 1. The Python script for Assignment 8 is organized by data, processing, presentation and the main body of the script.

Table

Description automatically generated

**Figure 1**: Pseudo-code from Assignment 8 starter code.

1. The standard class pattern in the Python language includes fields, constructors, attributes, properties, and methods. Figure 2 shows the basic components of a class as well as how to format the code within a class.

* **Fields** are created using variables and constants and considered the data members of a class.
* **Constructors** are special methods (functions) that set the initial values of field data and automatically run when an object from the class is created.
* **Attributes** are an invisible field that hold internal data and are similar to fields in that they are considered variables.
* **Properties** are also special methods (functions) used to manage field or attribute data. It is best practice to create two properties for each field or attribute, one for “getting” data and one for “setting” data.
* **Methods** are other functions inside of a class that are slightly different than properties. Methods are used to organize processing statements into named groups.

Text

Description automatically generated

**Figure 2:** An outline of the different components of a Python class.

1. In the Data section, declare variables.

Shape

Description automatically generated

**Figure 3:** Variables are declared in the Data section for the file to be used in the program and the list of product objects

1. Next within the Data section, define the Product class using the keyword “class” followed by docstring to provide additional context related to the code. Lines 28 illustrates how “\_\_intit\_\_” (double underscore “duder”) is used to set the initial values of product name and product price fields when. The keyword “self” is used in the constructor method to refer to data or functions found in an object instance, but not directly in the class. The try except block is used to handling errors that may occur when creating the initial values for the object.

Graphical user interface, text, application, email

Description automatically generated

**Figure 4:** An example of constructor and attribute code within a class called Product.

1. Code for properties is also included within the Data section and is used to manage field or attribute data. The “@property” directive is used to indicate a getter function that allows a programmer to add code to format a field or attribute’s data. The setter properties are used for both validation and error handling and use the “@name\_of\_method.setter” directive. Note, that the directive and the function/method name must match. In Figure 5, “product\_name” is used in the setter directive and the function that follows it.

Graphical user interface, text, application, email

Description automatically generated

**Figure 5:** Example of Properties within the Product class.

1. The last piece of defining a class in the Data section are the methods, which are other functions used to organize processing statements into a named group. The “\_\_str\_\_” method is used to represent the class objects as a string. This method is called when print() or str() function is used on an object. The “\_\_doc\_\_” method is used provide documentation on the object, which in this script will return a message that states, “This class holds product information”, as shown in Figure 6.

Table

Description automatically generated

**Figure 6:** Example of Methods within the Product class.

1. The Processing and Presentation section utilize code that was taught in previously modules and organize it within the FileProcessor and IO classes, respectively.

Table

Description automatically generated

**Figure 7:** Processing and Presentation sections of the Python script also utilize classes.

## Summary

This week, we developed a basic understanding of classes and the different components that fall within a class in Python. While this week’s topic was more advanced, beginner programmers will find using classes helpful for creating objects and future instances of objects within Python a lot easier.