

Exercise 1.5: Object-Oriented Programming in Python

Learning Goals

Apply object-oriented programming concepts to your Recipe app

Reflection Questions

1. In your own words, what is object-oriented programming? What are the benefits of OOP?

Object-oriented programming is a way of organising and designing data based on objects. In this context, objects act as containers that hold data, as well as functions that can be used to manipulate this data.

OOP has many benefits which include:

- the re-use of objects which saves time
- new classes can inherit methods from pre existing classes (inheritance)
- objects of different classes can be treated as objects from a common initial class (polymorphism)

2. What are objects and classes in Python? Come up with a real-world example to illustrate how objects and classes work.

In Python, classes are like a template that define the behaviour/structure of objects. Specifically, it can specify the attributes (data) of an object and the methods (actions) it can perform. Where as, objects are an example created from a class. Objects have their own unique attribute and methods.

A real life example and one that we have been looking at through this Python course is recipes. A recipe would be the class and the objects are the different recipes you can have. The objects have their own name, cooking time and ingredients related to the recipes.

3. In your own words, write brief explanations of the following OOP concepts; 100 to 200 words per method is fine.

Method	Description
Inheritance	Inheritance allows for a new class to inherit attributes and methods from a class that is already existing. The child class inherits from the parent class.
Polymorphism	Polymorphism is when objects from different classes can be treated as objects from a common class
Operator Overloading	Operator Overloading is when you define your own methods for operators (e.g +, -, *) within you class.