

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 allows a developer to create objects within classes. This tends to lend itself to clear, concise and organized code that has a high readability. Another significant benefit is in its reusability.

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

Objects are variables and methods and classes create and provide a layout for those objects. If, for example, a house is a class, varying objects fall under that class – a kitchen, living room, den, garage.

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 is a large part of what makes OOP work – objects inherit properties from classes. Similarly, classes themselves have a hierarchy with subclasses inheriting properties from a parent class. This avoids unnecessarily repetitive code.
Polymorphism	Polymorphism allows attributes and methods to have the same name across classes and data types but perform different tasks and operations depending on where and how they are defined.
Operator Overloading	Operator Overloading gives the developer an opportunity to adjust and customize the behavior of Python-set operators for objects of a class. Operator Overloading gives a syntax for working with those objects and operators that can give a more readable code.