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 method in which everything is considered an object that can be broken down into the stored data and the methods used to interact with it. The benefit of this method of programming is being able to use classes to generate new objects with the predefined attributes and methods.

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

   Everything in Python is considered an object, and every object has a corresponding class that act as a template for the objects' structure. A real-world example that can illustrate how this concept works is the animal kingdom. Your class, for example, can be warm-blooded vertebrates and so that lays out the blueprint for what kind of animal (our object) you could find in that class – the animal is warm-blooded and has a backbone.

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 you to use data attributes and methods previously defined in a whole new class. The data and procedures are available to the subclass without having to copy any code. Inheritance only works in one direction, so whatever extra attributes or procedures are defined in the subclass cannot be accessed by the parent class.
Polymorphism	A data attribute or method with the same name exists across different classes, but they perform different operations depending on where it was defined. For example, if two different classes are defined with a method called speak(), but when called by objects of their respective classes, the output is different and dependent on what was defined.
Operator Overloading	When utilizing Pythons ability to create your own custom class, the operators don't work the same way as they would on Python's built in classes. In order to use these operators, you need to define the methods of how the class should use that specific operator.