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 (OOP) is a way of writing code that makes it easier to understand and organize. We can use these objects in different parts of our program, which saves time and makes our code more efficient. OOP helps us keep our code neat and flexible, just like how we organize and reuse things in our everyday lives.

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

For example, imagine a class called "Car" that defines what a car is. It specifies the car's properties (like colour and brand) and actions it can perform.

Now, we can create car objects from the Car class. Each car object will have its own colour brand. For instance, we can create a red Toyota and a blue Honda. Each car has its own colour and brand. We can also make them perform actions like starting the engine.

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 when one class can inherit or receive characteristics from another class. It's like how children inherit traits from their parents. In programming, this helps us avoid repeating code and organize our classes better.

Polymorphism	Polymorphism is when different objects can be treated as the same type. It's like using a single tool for different purposes. In programming, it allows us to write code that can work with different types of objects, making our code more flexible and reusable.
Operator Overloading	Operator overloading is when we give special meanings to operators like "+", "-", "*", etc., so they can work with our custom objects. This allows us to use operators in ways that are different from their default behaviors. It helps us use familiar operators to perform specific actions with our objects, making our code more expressive and easier to use.