

Exercise 1.5

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

OOP creates objects from classes. When using OOP, code can be very clear, organized and does not require lots of new code every time a new object is introduced. This makes the code more efficient and readable.

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

Classes are a mold that new objects can fit into. For example, class “recipe” can be used to create any object that fits within its parameters. The parameters might be name, cooking time, ingredients, etc.

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

Method	Description
Inheritance	This is when you inherit traits from one class to another. Properties from a parent class can be shared with an inherited class to avoid repeating code over and over in different classes.
Polymorphism	Polymorphism is where a given data attribute or method has the same name across different classes or data types, but performs different operations depending on where it was defined.
Operator Overloading	Special operators will not work in custom classes so they need to be overridden with the process operator overloading. All it requires is defining a function with a name that Python already reserves for your operator, and surrounding it with double underscores like <code>__add__()</code> and <code>__sub__()</code> .