Assignment 1.2: Intro to Object Oriented Programming

Jaimil Dalwadi | Sept. 17, 2020 | ICS4UR

# **Questions:**

1. **Explain the benefits of object-oriented programming.**

Object-oriented programming has several benefits which allow it to allow for better functionality in many cases. To begin with, the code is organized into separate files instead of one large file. These files can be classes, which contain different objects (data and functions). Such classes work together to make one large program, making it easier to read, allows for less repeated code, and allows for simultaneous teamwork on a project. OOP programming supports reusable code allowing a developer to use past project classes with new projects. Java also provides several pre-made classes which can be used. As seen in the interactive presentation, Bob uses multiple companies (classes) to perform his required tasks (functions) with his product (data). This makes work easier to manage, modify, and improve.

1. **Discuss the difference between a class and an object.**

A class defines a type or template and an object is an instance of that type or class. The object can be changed, modified, or manipulated. The difference is that you cannot manipulate the class or type itself, but you can manipulate an instance of that class. For example, the class is like a template, and an object is a project you make using that template demonstrating how objects can be edited.

1. **Discuss the difference between properties and methods of a class.**

As mentioned previously, a class is like a template for an object. The properties and methods of a class are generally referred to as members of the class. The properties are like the characteristics of the class (data and variables) and the methods of the class are like the behaviours or actions of the class (functions and methods).

1. **What is code reuse? How does it relate to inheritance?**

Code reuse is the ability to use pre-made code instead of rewriting it. Classes can be extended to create subclasses. These subclasses allow for inheritance; the ability to inherit all the properties and methods of the original class. This original class is considered the super class. When working with the subclass, developers can define additional methods and properties, and override and/or overload methods (use the same name for multiple methods in a class) as well as properties defined in the superclass. Another way code repetition is minimized is through modules. Modules are collections of data and methods which can be used across different classes.

1. **What is polymorphism? How does it relate to methods?**

At its basis, polymorphism means “many-forms.” When applied to object-oriented programming, it refers to the ability of an object to take on multiple forms. This was seen earlier, where a superclass’ methods can be modified, overridden, and can be redefined in the subclass. For example, you can make a general template (superclass) for all your presentation projects. When you create a specific presentation for computer science, polymorphism allows change of the original template’s sections, fonts, and much more.

1. **Explain how the concepts of encapsulation and delegation a), relate to each other and b), how they differ from each other.**
   * 1. Encapsulation and delegation both work to fulfill responsibilities of the program. For example, the responsibility of a catering service is to provide large amounts of food to clients. This service can use both delegation and encapsulation to complete tasks and responsibilities.
     2. Encapsulation is the restriction of work to a singular responsibility or task, whereas delegation is the “handing-off” of tasks or responsibilities to other classes or programs. As with the example of the catering service, the main chef may delegate general tasks such as getting groceries, containers, or more, and encapsulate himself/herself to preparing the food.