9: Object Oriented Programming

Object-Oriented Programming

📖 Focused on creating objects: entity that contains data and procedures.

Encapsulation

📖 Combining data and procedure into a single object.

Data Hiding

📖 Object's data attributes are hidden from code outside the object gand access is restrictred to the object's methods.

- This protects from accidental corruption
- Outside code does not need to know internal structure of object

Class

⚡ Code that specifies the data attributes and methods of a particular type of object.

Instance

⚡ An object created from a class. There can be many instances of a class.

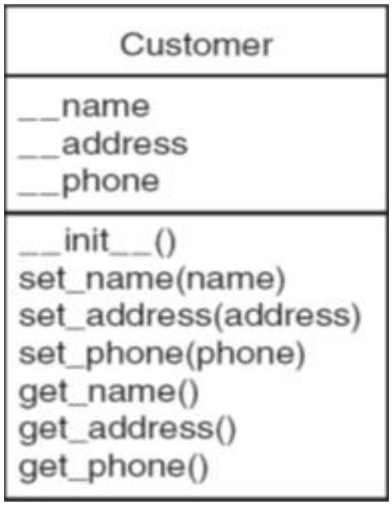
> Note that if many instances of a class are created, each would have its own set of attributes.

Techniques for Designing Classes

📕 Unified Modelling Language (UML) Diagram is a standard diagram for graphically depicting object oriented systems.

Generally,

- Draw a box divided into 3 sections:
  1. Top Section: name of Class
  2. Middle Section: List of Data Attributes
  3. Bottom Section: List of Class Methods



Inheritance

📖 Inheritance refers to the ability of an object to take on one or more characteristics from other classes of objects.

Superclass (Base class)

Subclass (Derived class)

⚡ A general class

⚡ A specialised class, an extended version of the super class.

> A subclass inherrits attributes and methods from the superclass

Polymorphism

📖 Polymorphism refers to an object's abillity to take different forms.

Essential Characteristics:

1. Define method in superclass and override it in a subclass (Define method with same name)
2. Call the correct version of overridden method depending on the type of object that is used to call it

Advantages:

1. Reusing Code
2. Reducing the coupling between different functionalities

Disadvantages:

1. Difficult to implement
2. Poor runtime and readability