01 - Major Characteristics of OOP

OBJECT-ORIENTED PROGRAMMING

CHARACTERISTICS OF OOP

- Objects
- Classes

OBJECTS

- Are single, uniquely identifiable entities or items
- Used as the building blocks of object-oriented programming
- Have identity, data, and behavior
- Can be simple or complex
- Can be real or imaginary
- Have attributes and operations
- Are a dynamic instance of a class

OBJECTS: EXAMPLES

- Cloud:
  - attributes:
    - shape, size, water
  - operations:
    - rain, thunder
- Bank Account:
  - attributes:
    - id, balance
  - operations:
    - open, close, withdraw, deposit

CLASSES

- A class is a definition of an object
- All object are instantiated or created from a class

MAJOR FEATURES OF OBJECT-ORIENTED PROGRAMMING

- Abstraction
- Encapsulation
- Association
- Aggregation
- Composition
- Inheritance
- Cohesion and Coupling
- Polymorphism

ABSTRACTION

- Process of ignoring details to concentrate on essential characteristics of an object or entity
- Simplifies functionality and the information
- Helps users interact with the object

ENCAPSULATION

- Refers to hiding the data inside of an object
- Produces two views of each object:
  - Outside view
  - Inside view

ASSOCIATION

- Refers to a way by which objects interact
- Objects are associated when one “uses” the services or operations of another

AGGREGATION

- Refers to the process of defining an object in terms of its components parts
- Is a type of Association
- Qualified by a “Has a” relationship

COMPOSITION

- Takes place when one object is contained within another
- Is a type of Association
- Qualified by a “Contains” relationship

EXERCISE 1: ABSTRACTION AND ENCAPSULATION

- **Objective:** To discuss the OO feature of Abstraction by describing the features of simple, everyday objects
- **Tasks**: discuss how to abstract the following
  - A car
  - An employee
  - A bank account

INHERITANCE

- Is a mechanism for defining a new class in terms of an existing class
- Qualified by the phrase “Is a” or “Kind of”
- Allows you to group related classes so they can be managed collectively and reused

COHESION AND COUPLING

- Cohesion – Measure of how a class, or a group of classes contribute to a single purpose within the system
- Coupling – Measure of how much two or more classes are connected to each other
- Coupling is also a measure of the dependencies within objects

POLYMORPHISM

- Refers to functions that you can apply to objects of different classes to achieve the same semantic result
- Similar operations defined for more than one class are polymorphic
- Based on Inheritance
- The implementation of a polymorphic function depends on the object to which it is applied

EXERCISE 2: INHERITANCE AND POLYMORPHISM

- **Discussion** – To use the OO features of Inheritance and Polymorphism.
- ***Task 1***. Classify the different types of vehicles, using inheritance to factor out common features.
- ***Task 2***. Think of as many polymorphic methods as you can, that is, the methods that apply to a class and all of its descendants
  - All electrical appliances that have a `turnOn` and `turnOff` method.
  - All audio appliances that have an `adjustVolume` method.
  - All telephones that have a `dial` and `hangup` method.