* Key concepts

1. Class:

* A **class** is a blueprint or template for creating objects.
* It defines the properties (attributes) and behaviors (methods) that objects created from the class will have

1. Object:

* An **object** is an instance of a class.
* it represents a specific implementation of the class, with actual values.

1. Encapsulation:

* Hides the internal state of an object and only allows access through public methods.
* Promotes **data security** and **modularity**.

1. Inheritance:

* Allows one class (derived class) to inherit properties and behaviors from another class (base class).
* Promote **code reusability**.

1. Polymorphism:

* Means “many forms.” Allows methods to do different things based on the object calling them.

Two types:

* **Compile-time (function overloading)**
* **Run-time (function overriding)**

1. Abstraction: -

* Hides complex implementation details and shows only essential features.
* Achieved using abstract classes or interfaces.