OOPS

1) Abstraction:

-> It shows only essential attributes and hides unnecessary information.

-> Abstraction is slecting data from a larger Pool to show only relevant details of the object to the user.

ex: A car is viewed as a car rather than its individual components.

-> The man only knows that pressing the accelerators will increase the speed of a car (or) applying brakes will stop the car, but he don't know how on pressing the accelerator the speed is actually increasing.

-> He doesn't know the inner mechanism of car.

2) Polymorphism:

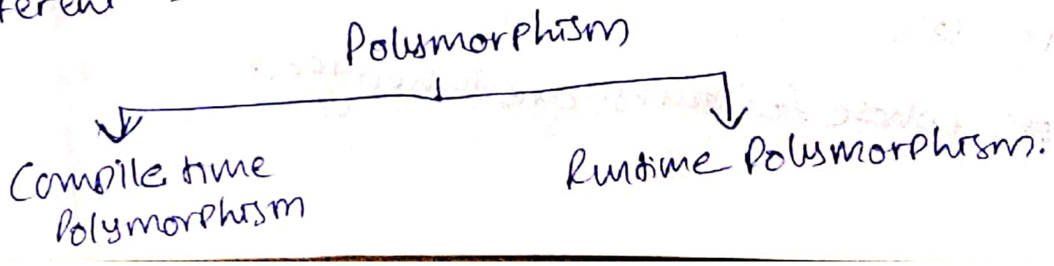
-> Ability of amessage to be displayed in more than one form as per the object class.

-> Allows to define one interface and have multiple implementations.

ex: A person at same time can have different characteristics.

-> Like a man at the same time is a father, a husband, a son and an employee.

-> So, Here same person posses different behaviour in different situations.



Compile time Polymorphism:

-> Also known as Static polymorphism.

-> This is achieved by function oveloading or operator overloading.

-> But java doesn't support the operator overloading.

Method overloading:

-> When there are multiple functions with same name but different parameters then these functions are said to be overloaded.

-> Functions can be overloaded by change in number of arguments or change in type of arguments.

Runtime Polymorphism:

-> Also known as Dynamic Method Dispatch.

-> This is a process in which a function call to the overridden method is resolved at Runtime.

-> This is achieved by Method overriding.

Method overriding:

-> When a derived class has a definition for one of the member functions of the base class.

-> That basic function is said to be overridden.

-> That methods must have the same no. of Parameters & same type of Parameter.

3) Inheritence:

-> It is the mechanism in java by which one class is allowed to inherit the features of another class.

Super class:

-> Class whose features are inherited.

Sub class:

-> The class that inherits the other class.

Reusability:

-> when we want to create a new class & there is already a class that includes some of the code that we want, we can derive our new class from the existing class.

-> By doing this, we are reusing the fields & methods of the existing class.

Syntax of inheritance:

Types of Inheritance:

i) Single Inheritance:

-> subclasses inherit the features of one superclass.

ii) Multilevel Inheritance:

~~-> Multilevel Inheritan~~

-> A derived class will be inheriting a base class & as well as the derived class also act as the base class to other class.

iii) Hierarchical Inheritance:

-> one class serves as a superclass (base class) ~~store~~ for more than one subclass.

iv) Multiple Inheritance:

-> one class can have more than one superclass & inherit features from all parent classes.

iv) Encapsulation:

-> Defined as the wrapping up data under a single unit. Another way in hide data that Prevent from being accused by the code outside the shield.

-> In encapsulation the variables (or data of a class is hidden from any other class & can be accessed only through any member function of its own class in which it is declared.

Advantages:

i) Data hiding

ii) Increased flexibility

iii) Reusability

iv) Testing code is easy

Basic definitions:

Object: An object is a collection of data & methods that operate on its data (Instance of class).

ex: i) car is an object, it has attributes like weight, color methods like drive, brake.

class: A class is an object constructor or blue print for creating objects.