

# Object Oriented Programming Concepts

Thursday, August 20, 2020

10:00 PM

Object Oriented Programming is a style of programming. As a definition, OOP is a method of implementation in which programs are organized as a cooperative collection of objects, each of which represents an instance of the class'. In other words OOP encapsulates data (attributes) and methods (behavior) into objects.

OOP is an advanced version of all the other programming techniques like assembly language programming, procedural and structured programming. In all the specified programming techniques, the following drawbacks can be identified that are solved in OOPS to a greater extent:

- Emphasis is more on the process rather than data.
- Functions are more interdependent and difficult to separate.
- Modifications of one function could lead to recompilation of entire application. No concept of true code reusability.
- Dynamic behavior is not easy to implement.
- Software development / maintenance is difficult to achieve and moreover expensive. Complex code, hard to write, debug and maintain.

OOPS Features:

1. Classes and objects
2. Data abstraction
3. Data encapsulation
4. Message Passing
5. Polymorphism
6. Inheritance
7. Extensibility

Class

A class is a blue print for generating various objects (or) a class is a blue print that defines the variables and methods common to all objects of a certain kind (or) a class is a prototype based on which objects can be derived (or) a class is a collection of data and it's corresponding methods. Or a class is a description of several objects.

NAME

- DATA # DATA

+ DATA

- METHOD #METHOD

+METHOD

Object

An object is an instance of a class. In other words an object is a physical construct that occupies certain amount of space in memory.

As a class is a logical construct, it doesn't occupy any memory where as an object is a physical construct and it occupies certain space.

The fundamental concept for OOP is an 'object ', which is an entity that has existence.

An object fundamentally consists of three characteristics:

- A state
- A behavior
- An identity

**Data abstraction:** Abstraction is the process of exhibiting only the essential characteristics of an object depending on programmers view.

Abstraction is of two types:

- Data abstraction
- Functional Abstraction

Ex :Data Abstraction - Empno,

Ename, Desg,

salary Functional Abstraction -

getDetails( ), dispDetails( )

**Data Encapsulation:** It is a process of wrapping up of data and methods in to a single entity.

**Message Passing:** It is nothing but invoking a method on an object.

**Polymorphism:** It is the ability to provide multiple definitions to the same method signature. It allows providing one interface multiple methods.

**Inheritance:** It is the ability to extract the features of one class to another. It is one of the striking features of inheritance. Java supports inheritance extensively as the main objective of inheritance is code reusability.

**Extendibility:** As we use object oriented approach, it is one of advantages. The code independence can be achieved.

Advantages of OOPs:

**Modularity:** All classes and objects can be treated as separate modules. This makes the designing simpler.

**Ease of Maintenance:** Code is easier to maintain. The concept of encapsulation localizes the errors. I.e., debugging is made easy.

**Reusability and Extensibility:** Through the concept of inheritance, we can easily extend existing classes. We can alter the behavior and also add new features. Code reusability is the great advantage.

**Powerful modeling paradigm:** The system based on OOP is close to real world and hence simple to understand.