

OOPs Concepts

Object Oriented Programming is a paradigm that provides many concepts such as **inheritance, data binding, polymorphism** etc.

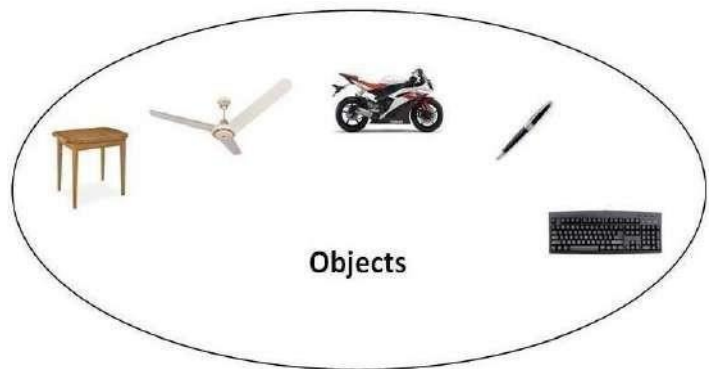
Simula is considered as the first object-oriented programming language. The programming paradigm where everything is represented as an object is known as truly object-oriented programming language.

Smalltalk is considered as the first truly object-oriented programming language.

OOPs (Object Oriented Programming System)

Object means a real word entity such as pen, chair, table etc. **Object-Oriented Programming** is a methodology or paradigm to design a program using classes and objects. It simplifies the software development and maintenance by providing some concepts:

- Object
- Class
- Inheritance
- Polymorphism
- Abstraction
- Encapsulation



Object

Any entity that has state and behavior is known as an object. For example: chair, pen, table, keyboard, bike etc. It can be physical and logical.

Class

Collection of objects is called class. It is a logical entity.

Inheritance

When one object acquires all the properties and behaviours of parent object i.e. known as inheritance. It provides code reusability. It is used to achieve runtime polymorphism.

Polymorphism

When **one task is performed by different ways** i.e. known as polymorphism. For example: to convince the customer differently, to draw something e.g. shape or rectangle etc.

In java, we use method overloading and method overriding to achieve polymorphism. Another example can be to speak something e.g. cat speaks meow, dog barks woof etc.

Abstraction

Hiding internal details and showing functionality is known as abstraction. For example: phone call, we don't know the internal processing.

In java, we use abstract class and interface to achieve abstraction.

Encapsulation

Binding (or wrapping) code and data together into a single unit is known as encapsulation. For example: capsule, it is wrapped with different medicines.

A java class is the example of encapsulation. Java bean is the fully encapsulated class because all the data members are private here.

Benefits of Inheritance

- One of the key benefits of inheritance is to minimize the amount of duplicate code in an application by sharing common code amongst several subclasses. Where equivalent code exists in two related classes, the hierarchy can usually be refactored to move the common code up to a mutual superclass. This also tends to result in a better organization of code and smaller, simpler compilation units.
 - Inheritance can also make application code more flexible to change because classes that inherit from a common superclass can be used interchangeably. If the return type of a method is superclass
 - **Reusability** - facility to use public methods of base class without rewriting the same.
 - **Extensibility** - extending the base class logic as per business logic of the derived class.

- **Data hiding** - base class can decide to keep some data private so that it cannot be altered by the derived class

Procedural and object oriented programming paradigms

Features	Procedural Oriented Programming (POP)	Object Oriented Programming (OOPS)
Divided into	In POP, program is divided into smaller parts called as functions.	in OOPs , the program is divided into parts known as objects.
Importance	In POP, importance is not given to data but to functions as well as sequence of actions to be done.	In OOPs, Importance is given to the data rather than procedures or functions because it works as a real world .
Approach	POP follows Top Down approach .	OOPs follows Bottom Up approach .
Access Specifiers	POP does not have any access specifier.	OOPs has access specifiers named Public, Private, Protected, etc.
Data Moving	In POP, Data can move freely from function to function in the system.	In OOPs, objects can move and communicate with each other through member functions.
Data Access	In POP, Most function uses Global data for sharing that can be accessed freely from function to function in the system.	In OOPs, data can not move easily from function to function.it can be kept public or private so we can control the access of data.
Data Hiding	POP does not have any proper way for hiding data so it is less secure .	OOPs provides Data Hiding so provides more security .
Overloading	In POP, Overloading is not possible.	In OOPs, overloading is possible in the form of Function Overloading and Operator Overloading.
Examples	C, VB, FORTRAN, Pascal.	C++, JAVA, VB.NET, C#.NET.

Java Programming- History of Java

The history of java starts from Green Team. Java team members (also known as **Green Team**), initiated a revolutionary task to develop a language for digital devices such as set-top boxes, televisionsetc.

For the green team members, it was an advance concept at that time. But, it was suited for internet programming. Later, Java technology as incorporated by Netscape.

Currently, Java is used in internet programming, mobile devices, games, e-business solutions etc. There are given the major points that describes the history of java.

- 1) **James Gosling, Mike Sheridan, and Patrick Naughton** initiated the Java language project in June 1991. The small team of sun engineers called **Green Team**.
- 2) Originally designed for small, embedded systems in electronic appliances like set-topboxes.
- 3) Firstly, it was called "**Greentalk**" by James Gosling and file extension was.gt.
- 4) **After that, it was called Oak and was developed as a part of the Green project.**

Java Version History

There are many java versions that has been released. Current stable release of Javais Java SE 8.

- JDK Alpha and Beta (1995)
- 2. JDK 1.0 (23rd Jan, 1996)
- 3. JDK 1.1 (19th Feb, 1997)
- 4. J2SE 1.2 (8th Dec, 1998)
- 5. J2SE 1.3 (8th May, 2000)
- 6. J2SE 1.4 (6th Feb, 2002)
- 7. J2SE 5.0 (30th Sep,2004)
- Java SE 6 (11th Dec,2006)
- Java SE 7 (28th July, 2011)
- 10.Java SE 8 (18th March,2014)