

# OOPs Concept in Java

## Class-

Class is a collection of object. It does not consume any space. It is a logical entity.

## Object-

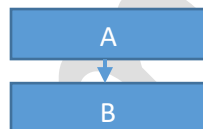
Object is an instance of class. It contains space in memory and address. It is a physical entity.

## Inheritance in java

When one class acquires all the properties and behavior of any other parent class, it is called an inheritance.

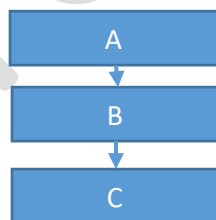
## Single Inheritance-

When one class inherits the properties of another single class, it is called a single inheritance.



## Multilevel Inheritance-

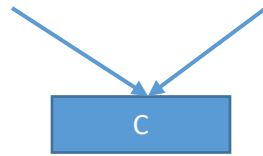
When there is a chain of inheritance, it is called a multilevel inheritance.



## Hierarchical Inheritance

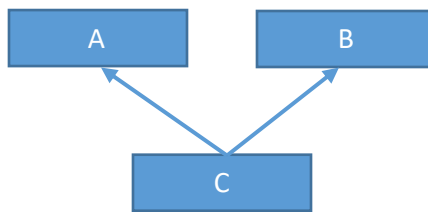
When two or more classes inherit the property of another single class, it is called a hierarchical inheritance.



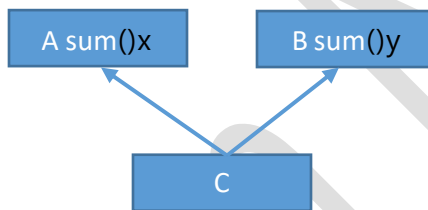


## Multiple Inheritance

When a One class inherits the properties of two or more class is called a multiple inheritance.



***Multiple Inheritance is not support in java.Why??***



We create object of C class

```
C c1 = new C ();
```

```
C1. sum()A/B
```

**HERE JVM will be confused as to what to display  
that's why multiple inheritance is not support in java.**

## Encapsulation

Wrapping or binding the data and method into a single unit is called an encapsulation. Encapsulation is like a capsule which has different ingredients in a single unit.

Class is a best example of encapsulation which contain data and methods in a single unit.

**Abstraction** Hiding internal part and showing the functionalities is called abstraction. Abstraction like a car which has hide the engine and show the car functionalities.

Company or collage portal is the best example of abstraction which only shows its functionality, it does not show the internal code.

## **Polymorphism**

When one task perform in a different way is called a polymorphism. Same object having different way.

**Real-life Illustration of Polymorphism in Java:** A person at the same time can have different characteristics. Like a man at the same time is a father, a husband, and an employee. So the same person possesses different behaviors in different situations. This is called polymorphism.

### **There are two types of polymorphism:-**

1. Compile time polymorphism
2. Runtime polymorphism

### **Compile time polymorphism:**

Those code which runs at the compile time that is called compile time polymorphism. This is also known as method overloading. If one class contain more than one method having same name but different in parameter that is called method **overloading**

```
Class A{  
  
    Public static sum(int a, int b){  
  
        //some code  
  
    }  
}
```

```

    Public static sum(int x, int y){

        //some code

    }

}

```

## **Runtime Polymorphism**

Those Program which runs at the runtime that is called runtime polymorphism. This is also known as method overriding. If child class has same method as declared in the parent class, it is called Method overriding.

```

// Creating parent class
Class A{
//Defining method
void sum()
//creating child class
}
Class B extends A{
//defining same method as in a parent class
Void sum()
}
Public static void main(string args[]){
Bike b = new bike();//object created
b.run();//calling the method
}
}

```

## **Constructor**

Constructor is a block of code which has automatically called when object created.

### **Type of constructor:**

#### **1. Default Constructor**

Those constructor which have not any parameter that is called Default constructor.

## 2. Parameterized constructor

Those constructor which have parameter that is called parameterized constructor.

## 3. Copy constructor

One constructor copy another constructor that is called copy constructor.

## **Static Keyword**

Static keyword is mainly used for the memory management. We are apply static keyword with variables, blocks, methods.

*If we declare variable as a static that is called **static variable***

## **This keyword**

In Java, this keyword is a reference variable that is refer to a current object.