# Model paper of Java

# Q2-Define inheritance. Explain any two types of inheritance supported by java with coding examples.

**Ans:**

It is the mechanism in java by which one class is allowed to inherit the features (fields and methods) of another class. In Java, inheritance means creating new classes based on existing ones. A class that inherits from another class can reuse the methods and fields of that class.

**Types of Inheritance:**

There are four types of inheritance.

1. Single Inheritance
2. Multilevel Inheritance
3. Hierarchical Inheritance

**2. Multilevel Inheritance:**

In Multilevel Inheritance, a derived class will be inheriting a base class, and as well as the derived class also acts as the base class for other classes. In the below image, class A serves as a base class for the derived class B, which in turn serves as a base class for the derived class C. In Java, a class cannot directly access the grandparent’s members.

**Program**

**import java.io.\*;**

**import java.lang.\*;**

**import java.util.\*;**

**class one {**

**public void print\_geek() {**

**System.out.println("Geeks"); }}**

**class two extends one {**

**public void print\_for() { System.out.println("for"); }}**

**class three extends two {**

**public void print\_geek()**

**{**

**System.out.println ("Geeks");**

**}**

**}**

**// Derived class**

**public class Main {**

**public static void main(String[] args)**

**{**

**three g = new three();**

**g.print\_geek ();**

**g.print\_for ();**

**g.print\_geek();**

**}**

**}**

**2. Hierarchical Inheritance**

In Hierarchical Inheritance, one class serves as a superclass (base class) for more than one subclass. In the below image, class A serves as a base class for the derived classes B, C, and D.

**Program**

**class A {**

**public void print\_A() { System.out.println("Class A"); }**

**}**

**class B extends A {**

**public void print\_B() { System.out.println("Class B"); }**

**}**

**class C extends A {**

**public void print\_C() { System.out.println("Class C"); }**

**}**

**class D extends A {**

**public void print\_D() { System.out.println("Class D"); }**

**}**

**// Driver Class**

**public class Test {**

**public static void main(String[] args)**

**{**

**B obj\_B = new B();**

**obj\_B.print\_A();**

**obj\_B.print\_B();**

**C obj\_C = new C();**

**obj\_C.print\_A();**

**obj\_C.print\_C();**

**D obj\_D = new D();**

**obj\_D.print\_A();**

**obj\_D.print\_D();**

**}**

**}**