**Assignment: (Core Java)**

**10: Interfaces and Abstract Classes**

**Que.1 Basics of OOP: Encapsulation, Inheritance, Polymorphism, Abstraction**

**Ans.1 An abstract class** in Java is a class that is declared with the keyword abstract.

* It cannot be instantiated.
* It may contain abstract methods as well as concrete methods.
* Abstract classes are mainly used to provide a base class for other classes to extend.

An **abstract method**: Is declared using the abstract keyword. Does not have a body . Must be implemented by the subclass.

**Que.2 Interfaces: Multiple Inheritance in Java**

**Ans.2** An interface in Java is like a blueprint of a class.

* It contains abstract methods by default.
* From Java 8 onwards, interfaces can also have default and static methods.
* From Java 9 onwards, they can have private methods as helpers.
* A class can implement multiple interfaces, thus achieving multiple inheritance.

**Que.3 Implementing Multiple Interfaces**

**Ans.3** A class can only extend one parent class (single inheritance). But it can implement multiple interfaces → this is how multiple inheritance is achieved.

// First interface

interface Printable {

void print();

}

// Second interface

interface Showable {

void show();

}

// Class implementing both interfaces

class Demo implements Printable, Showable {

@Override

public void print() {

System.out.println("Hello ...");

}

@Override

public void show() {

System.out.println("How are you...");

}

}

public class Main {

public static void main(String[] args) {

Demo obj = new Demo();

obj.print();

obj.show();

}

}