

**Attempt Any two questions:-**

1. Create one superclass HillStations and three subclasses Manali, Mussoorie, Gulmarg. Subclasses extend the superclass and override its location() and famousFor() method. i.call the location() and famousFor() method by the Parent class’, i.e. Hillstations class. As it refers to the base class object and the base class method overrides the superclass method; the base class method is invoked at runtime. ii.call the location() and famousFor() method by the all subclass’,and print accordingly.

Code:

Public class HillStations {

public void location() {

System.out.println("Location: Various locations in India");

}

public void famousFor() {

System.out.println("Famous for: Scenic beauty and adventure sports");

}

}

class Manali extends HillStations {

public void location() {

System.out.println("Location: Manali");

}

public void famousFor() {

System.out.println("Famous for: mountains ");

}

}

class Mussoorie extends HillStations {

public void location() {

System.out.println("Location: Mussoorie");

}

public void famousFor() {

System.out.println("Famous for: Hills ");

}

}

class Gulmarg extends HillStations {

public void location() {

System.out.println("Location: Gulmarg");

}

public void famousFor() {

System.out.println("Famous for: sky dive");

}

}

public static void main(String[] args) {

HillStations hillStation = new HillStations();

hillStation.location();

hillStation.famousFor();

System.out.println();

HillStations manali = new Manali();

manali.location();

manali.famousFor();

System.out.println();

HillStations mussoorie = new Mussoorie();

mussoorie.location();

mussoorie.famousFor();

System.out.println();

HillStations gulmarg = new Gulmarg();

gulmarg.location();

gulmarg.famousFor();

}

}

---------------------------------------------------------------------

1. Write a Java program that demonstrates method overriding by creating a superclass called Animal and two subclasses called Dog and Cat. ● The Animal class should have a method called makeSound(), which simply prints "The animal makes a sound." ● The Dog and Cat classes should override this method to print "TheCat/The dog meows/barks" respectively. ● The program should allow the user to create and display objects of each class.

class Animal {

public void makeSound() {

System.out.println("The animal makes a sound");

}

}

class Dog extends Animal {

public void makeSound() {

System.out.println("The dog barks");

}

}

class Cat extends Animal {

public void makeSound() {

System.out.println("The cat meows");

}

}

public static void main(String[] args) {

Animal obj1 = new Animal();

Dog obj2 = new Dog();

Cat obj3 = new Cat();

animal.makeSound();

dog.makeSound();

cat.makeSound();

}

}