1. Write a program Employee parent class in that Data Member salary and member function void dispSalary(). PermanentEmp class and TemporaryEmp both child class both having incrementSalary() for salary calculation. PermanentEmp hike is 50% and TemporaryEmp hike 35% calculate both employee salary after hike .
2. Write Java program to show that private member of a super class cannot be accessed from the derived classes
3. Consider trunk calls of a telephone exchange. A trunk call can be ordinary, urgent or lightning call. The charges depend on the duration and the type of the call. Write a program using concept of polymorphism in Java to calculate the chases
4. A superclass named “Shapes” has a method “area()”. Subclasses of “Shapes” can be “Triangle”, “circle”, “Rectangle”. Each subclass has its way of calculating area. Using Inheritance and Polymorphism, the subclasses can use the “area()” method to find the area’s formula for that shape.
5. Creating one superclass Hillstations and three subclasses Manali, Mussoorie, Gulmarg. Subclasses extend the superclass and override its location() and famousfor() method. We will 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

1.

class Employee{  
float salary = 40000;  
}  
class PermanentEmp extends Employee{  
double hike = 0.5;  
}  
class TemporaryEmp extends Employee{  
double hike = 0.35;  
}  
public class HerInheritanceDemo  
{  
public static void main(String args[]){  
PermanentEmp p = new PermanentEmp();  
TemporaryEmp t = new TemporaryEmp();  
// All objects of inherited classes can access the variable of class Employee  
System.out.println("Permanent Employee salary is :" +p.salary);  
System.out.println("Hike for Permanent Employee is:" +p.hike);  
System.out.println("Temporary Employee salary is :" +t.salary);  
System.out.println("Hike for Temporary Employee is :" +t.hike);  
}  
}

2

class room  
{  
private int l,b;  
room(int x,int y)  
{ l=x; b=y;}  
int area()  
{return(l\*b);}  
}  
class class\_room extends room  
{  
int h;  
class\_room(int x,int y,int z)  
{  
super(x,y);  
h=z;  
}  
int volume()  
{  
return(area()\*h);  
}  
}  
class s04\_01  
{  
public static void main(String args[])  
{  
class\_room cr=new class\_room(10,20,15);  
int a1=cr.area();  
int v1=cr.volume();  
System.out.println("Area of Room : "+a1);  
System.out.println("Volume of Room : "+v1);  
}  
}

3

class call  
{  
float dur;  
String type;  
float rate()  
{  
if(type.equals("urgent"))  
return 4.5f;  
else if(type=="lightning")  
return 3.5f;  
else  
return 3f;  
}  
}  
class bill extends call  
{  
float amount;  
DataInputStream in=null;  
bill()  
{  
try  
{  
in=new DataInputStream(System.in);  
}  
catch(Exception e)  
{ System.out.println(e); }  
}  
void read()throws Exception  
{  
String s;  
System.out.println("enter call type(urgent,lightning,ordinary):");  
type=in.readLine();  
System.out.println("enter call duration:");  
s=in.readLine();  
dur=Float.valueOf(s).floatValue();  
}  
void calculate()  
{  
if(dur<=1.5)  
amount=rate()\*dur+1.5f;  
else if(dur<=3)  
amount=rate()\*dur+2.5f;  
else if(dur<=5)  
amount=rate()\*dur+4.5f;  
else  
amount=rate()\*dur+5f;  
}  
void print()  
{  
System.out.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");  
System.out.println(" PHONE BILL ");  
System.out.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");  
System.out.println(" Call type : "+type);  
System.out.println(" Duration : "+dur);  
System.out.println(" CHARGE : "+amount);  
System.out.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");  
}  
}  
class s04\_04  
{  
public static void main(String arg[])throws Exception  
{  
bill b=new bill();  
b.read();  
b.calculate();  
b.print();  
}  
}

4

public class Shape

{

private double height; // To hold height.

private double width; //To hold width or base

// Set height and width

public void setValues(double height, double width)

{

this.height = height;

this.width = width;

}

//Get height

public double getHeight()

{

return height;

}

//Get width

public double getWidth()

{

return width;

}

}

public class Rectangle extends Shape

{

//Calculate and return area of rectangle

public double getArea()

{

return getHeight() \* getWidth();

}

}

public class Triangle extends Shape

{

//Calculate and return area of triangle

public double getArea()

{

return (getHeight() \* getWidth()) / 2;

}

}

public class PolymorphismDemo

{

public static void main(String[] args)

{

Shape shape;

// assign subclass reference to subclass variable

Rectangle rect = new Rectangle();

// shape points to the object rect.

shape = rect;

// Set data in Rectangle's object

shape.setValues(78, 5);

//Display the area of rectangle

System.out.println("Area of rectangle : " + rect.getArea());

// assign subclass reference to subclass variable

Triangle tri = new Triangle();

// shape points to the object rect.

shape = tri;

// Set data in Triangle's object

shape.setValues(34,3);

//Display the area of triangle

System.out.println("Area of triangle : " + tri.getArea());

}

}

5

class Hillstations{

  void location(){

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

}

void famousfor(){

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

}

}

class Manali extends Hillstations {

  void location(){

System.out.println("Manali is in Himachal Pradesh");

}

void famousfor(){

System.out.println("It is Famous for Hadimba Temple and adventure sports");

}

  }

class Mussoorie extends Hillstations {

  void location(){

System.out.println("Mussoorie is in Uttarakhand");

}

void famousfor(){

System.out.println("It is Famous for education institutions");

}

  }

class Gulmarg extends Hillstations {

  void location(){

System.out.println("Gulmarg is in J&amp;K");

}

void famousfor(){

System.out.println("It is Famous for skiing");

}

  }

class main{

  public static void main(String args[]){

    Hillstations A = new Hillstations();

    Hillstations M = new Manali();

    Hillstations Mu = new Mussoorie();

    Hillstations G = new Gulmarg();

    A.location();

A.famousfor();

M.location();

M.famousfor();

Mu.location();

Mu.famousfor();

G.location();

G.famousfor();

  }

}