1. Interface program

interface Shape{

double pi =3.14;

public double perimeter();

public double area();

}

class Circle implements Shape{

float radius;

Circle(float r){

radius = r;

}

public double area(){

return pi\*radius\*radius;

}

public double perimeter(){

return 2\*pi\*radius;

}

}

class Ecllipse implements Shape{

float a,b;

Ecllipse(float a,float b){

this.a = a;

this.b=b;

}

public double perimeter(){

return (2\*pi\*Math.sqrt(a\*a+b\*b));

}

public double area(){

return pi\*a\*b;

}

}

public class Main

{

public static void main(String[] args) {

Circle c = new Circle(3.0f);

Ecllipse e = new Ecllipse(7.0f,3.0f);

System.out.println("area of circle is "+ c.area());

System.out.println("area of ellipse is "+ e.area());

System.out.println("perimeter of circle is "+ c.perimeter());

System.out.println("perimeter of ecclipse is "+ e.perimeter());

}

}

\_\_\_\_\_\_\_\_\_\_\_\_\*\_\_\_\_\_\_\_\_\_\_\_\_

1. Vehicle program

class vehicle {

int capacity;

float consumption;

float distance ;

vehicle(int c, float consume , float d)

{

capacity = c;

consumption = consume;

distance = d;

}

}

class car extends vehicle {

car(int c, float consume , float d)

{

super(c,consume,d);

}

float computeDistance()

{

float d = (capacity \* 100.0f) / consumption;

return d;

}

float computeFuel(int d)

{

return d/ this.distance;

}

}

class Truck extends vehicle

{

Truck(int c, float consume , float d)

{

super(c,consume,d);

}

float computeDistance()

{

float d = (capacity \* 100.0f) / consumption;

return d;

}

float computeFuel(int d)

{

return d/ this.distance;

}

}

public class Main

{

public static void main(String args[])

{

car c = new car(70,10,750);

Truck T = new Truck (50,17,750);

System.out.println("distance covered on ful tank " + c.computeDistance());

System.out.println("no of filling is required"+ c.computeFuel(750));

System.out.println("distance covered on ful tank " + T.computeDistance());

System.out.println("no of filling is required"+ T.computeFuel(750));

}

}

\_\_\_\_\_\_\_\_\_\_\_\*\_\_\_\_\_\_\_\_\_\_\_\_

1. Inheritance program:
2. class Student
3. {
4. int regno;
5. String name, dept;
6. Student(int rno, String name, String dept)
7. {
8. regno = rno; this.name=name; this.dept =dept;
9. }
10. void display()
11. {
12. System.out.print(regno + " "+ name + " "+ dept);
13. }
14. }
15. class Test extends Student
16. {
17. int marks[];
18. Test(int r, String n, String d, int m[])
19. {
20. super(r,n,d);
21. marks = m;
22. }
23. }
24. class Result extends Test
25. {
26. Result(int r, String n, String d, int m[])
27. {
28. super(r,n,d,m);
29. }
30. void printResult() {
31. display();
32. int sum = 0;
33. for (int i =0; i< marks.length; i++)
34. sum += marks[i];
35. System.out.println(" Total Marks: "+ sum + " Average = " + (sum/5.0) );
36. }
37. }
38. public class TestMain
39. {
40. public static void main(String arg[])
41. {
42. int mark[]= {97,89,98,96,87};
43. Result r = new Result(4045, "Malini", "CSE", mark1);
44. r.printResult();
45. int mark2[] = {86,98,86,89,98};
46. Result r2 = new Result(4055,"Pramika", "CSE", mark2);
47. r2.printResult();
48. }
49. }

\_\_\_\_\_\_\_\_\_\_\_\_\*\_\_\_\_\_\_\_\_\_\_\_