1) Exercise Objective(s):Package

Exercise:Create a package called shapes. Create some classes in the package representing some

common geometric shapes like Square, Triangle, Circle and so on. Create a class called

TestShapes and create objects for all the shapes and print corresponding messages.

Execute the TestShapes class.

package shapes;

public class Circle {

public void volume() {

System.***out***.println("the volume of circle is 3.14\*r\*r");

}

}

package shapes;

public class Square {

public void volume() {

System.***out***.println("the volume of square is a\*a");

}

}

package shapes;

public class Triangle {

public void vol() {

System.***out***.println("the volume of triangle is 1/2\*b\*h");

}

}

package classprg;

import shapes.Circle;

import shapes.Square;

import shapes.Triangle;

public class TestShapes {

public static void main(String[] args) {

Circle circle = new Circle();

Square square = new Square();

Triangle triangle = new Triangle();

circle.volume();

square.volume();

triangle.vol();

}

}

2) Exercise Objective(s):Overloading

Exercise:Create a class called shape with the following methods

1. area

2. perimeter

Overload the area and perimeter method to calculate for both square and rectangle.

Create a main class and invoke the area method to calculate the area of the square and

rectangle. Also invoke the perimeter method to calculate the perimeter of the square

and rectangle.

package assignment;

public class Shapes {

public void area(int a) {

int sqArea=0;

sqArea=a\*a;

System.***out***.println("the area of square is: "+sqArea);

}

public void area(int l,int b,int h) {

int rectArea=0;

rectArea=l\*b\*h;

System.***out***.println("the area of square is: "+rectArea);

}

public void Perimeter(int a) {

int perimeter=0;

perimeter=4\*a;

System.***out***.println("The perimeter of square is "+perimeter);

}

public void Perimeter(int l,int b,int h) {

int perimeter=0;

perimeter=2\*(l+b);

System.***out***.println("The perimeter of square is "+perimeter);

}

public static void main(String[] args) {

Shapes s=new Shapes();

s.area(4);

s.area(2,3,4);

s.Perimeter(4);

s.Perimeter(2,3,4);

}

}

3) Exercise Objective(s):Overloading

Exercise:Create a class called Calculator which has 4 different methods add, diff, mul and div which

accepts two numbers as parameters. Overload the methods such that the parameters can be

of the following pattern.

1. Both are of int data type.

2. Both are of double data type.

3. First parameter is of int data type and second parameter is of double data type.

4. First parameter is of double data type and second parameter is of int data type.

Create anobject to access these methods and invoke these methods with different type of

numbers and display the result in the corresponding methods.

package assignment;

class Calculater {

public int add(int a, int b) {

return a + b;

}

public double add(double a, double b) {

return a + b;

}

public double add(int a, double b) {

return a + b;

}

public double add(double a, int b) {

return a + b;

}

public int diff(int a, int b) {

return a - b;

}

public double diff(double a, double b) {

return a - b;

}

public double diff(int a, double b) {

return a - b;

}

public double diff(double a, int b) {

return a - b;

}

public int mul(int a, int b) {

return a \* b;

}

public double mul(double a, double b) {

return a \* b;

}

public double mul(int a, double b) {

return a \* b;

}

public double mul(double a, int b) {

return a \* b;

}

public int div(int a, int b) {

return a / b;

}

// Method for dividing two doubles

public double div(double a, double b) {

return a / b;

}

public double div(int a, double b) {

return a / b;

}

public double div(double a, int b) {

return a / b;

}

}

public class Calculator2 {

public static void main(String[] args) {

Calculater calculator = new Calculater();

System.***out***.println("Addition:");

System.***out***.println("Int + Int: " + calculator.add(5, 3));

System.***out***.println("Double + Double: " + calculator.add(5.5, 3.3));

System.***out***.println("Int + Double: " + calculator.add(5, 3.3));

System.***out***.println("Double + Int: " + calculator.add(5.5, 3));

System.***out***.println("\nDifference:");

System.***out***.println("Int - Int: " + calculator.diff(5, 3));

System.***out***.println("Double - Double: " + calculator.diff(5.5, 3.3));

System.***out***.println("Int - Double: " + calculator.diff(5, 3.3));

System.***out***.println("Double - Int: " + calculator.diff(5.5, 3));

System.***out***.println("\nMultiplication:");

System.***out***.println("Int \* Int: " + calculator.mul(5, 3));

System.***out***.println("Double \* Double: " + calculator.mul(5.5, 3.3));

System.***out***.println("Int \* Double: " + calculator.mul(5, 3.3));

System.***out***.println("Double \* Int: " + calculator.mul(5.5, 3));

System.***out***.println("\nDivision:");

System.***out***.println("Int / Int: " + calculator.div(5, 3));

System.***out***.println("Double / Double: " + calculator.div(5.5, 3.3));

System.***out***.println("Int / Double: " + calculator.div(5, 3.3));

System.***out***.println("Double / Int: " + calculator.div(5.5, 3));

}

}

4) Exercise Objective(s):The concept of inheritance

Exercise:Create a class called Vehicle. Create subclasses like Truck, Bus, Car etc. Add common methods

in the base class and specific methods in the corresponding class. Create a class called Road

and create objects for the Truck, Car, Bus etc and display the appropriate message

package assignment;

class Vehicle {

public void start() {

System.***out***.println("Vehicle is starting.");

}

public void stop() {

System.***out***.println("Vehicle is stopping.");

}

}

class Truck extends Vehicle {

public void Truckvehicle() {

System.***out***.println("This is truck.");

}

}

class Bus extends Vehicle {

public void BusVehicle() {

System.***out***.println("This is bus.");

}

}

class Car extends Vehicle {

public void CarVehicle() {

System.***out***.println("This is car.");

}

}

class Road {

public static void main(String[] args) {

Truck truck = new Truck();

Bus bus = new Bus();

Car car = new Car();

truck.start();

truck.Truckvehicle();

truck.stop();

bus.start();

bus.BusVehicle();

bus.stop();

car.start();

car.CarVehicle();

car.stop();

}

}