Candidate Id: 2380155

Name : Bharath Magesh

Assignment No : 03

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.

Answer:

Square.java

```
package shapes;

public class Square {
	void box() {
		System.out.println("Square has 4 sides..");
} }
```

Triangle.java

```
package shapes;

public class Triangle {
	void three() {
		System.out.println("Triangle has 3 sides..");
	}  }
```

Circle.java

```
package shapes;

public class Circle {
	public void round() {
		System.out.println("Circle is round..");
	}
}
```

TestShapes.java

```
package shapes;

import shapes.*;

public class TestShapes {

	public static void main(String[] args) {
		// TODO Auto-generated method stub
		Circle obj = new Circle();
		obj.round();

		Square obj1 = new Square();
		obj1.box();

		Triangle obj2 = new Triangle();
		obj2.three();


	}

}
```

Output:

Circle is round..

Square has 4 sides..

Triangle has 3 sides..

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.

Answer:

Main.class

```
package assignment.day3;

class Shape {

  // Overloaded method for area of square

  public double area(double side) { return side * side; }


  // Overloaded method for area of rectangle

  public double area(double length, double breadth) { return length * breadth; }


  // Overloaded method for perimeter of square

  public double perimeter(double side) { return 4 * side; }


  // Overloaded method for perimeter of rectangle

  public double perimeter(double length, double breadth) { return 2 * (length +
breadth); }

}


public class Main {

  public static void main(String[] args) {

    Shape shape = new Shape();
```

```
        // Square
        double squareSide = 5.0;
        System.out.println("Area of square: " + shape.area(squareSide));
        System.out.println("Perimeter of square: " + shape.perimeter(squareSide));
        // Rectangle
        double rectangleLength = 6.0;
        double rectangleBreadth = 4.0;
        System.out.println("Area of rectangle: " + shape.area(rectangleLength,
rectangleBreadth));
        System.out.println("Perimeter of rectangle: " + shape.perimeter(rectangleLength,
rectangleBreadth));
    }
}
```

Output:

Area of square: 25.0

Perimeter of square: 20.0

Area of rectangle: 24.0

Perimeter of rectangle: 20.0

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 an object to access these methods and invoke these methods with different type of numbers and display the result in the corresponding methods.

Answer:
Main1.class

```
package assignment.day3;


class Calculator {
  // Addition
  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;
}

// Subtraction
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;
}

// Multiplication
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;
    }

    // Division
    public int div(int a, int b) {
        return a / b;
    }

    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 Main1 {
    public static void main(String[] args) {
        Calculator calculator = new Calculator();

        // Testing add method
        System.out.println("Addition:");
        System.out.println("int + int: " + calculator.add(5, 3));
        System.out.println("double + double: " + calculator.add(5.5, 3.2));
        System.out.println("int + double: " + calculator.add(5, 3.2));
        System.out.println("double + int: " + calculator.add(5.5, 3));

        // Testing diff method
        System.out.println("\nSubtraction:");
        System.out.println("int - int: " + calculator.diff(5, 3));
        System.out.println("double - double: " + calculator.diff(5.5, 3.2));
        System.out.println("int - double: " + calculator.diff(5, 3.2));
        System.out.println("double - int: " + calculator.diff(5.5, 3));

        // Testing mul method
        System.out.println("\nMultiplication:");
        System.out.println("int * int: " + calculator.mul(5, 3));
        System.out.println("double * double: " + calculator.mul(5.5, 3.2));
        System.out.println("int * double: " + calculator.mul(5, 3.2));
        System.out.println("double * int: " + calculator.mul(5.5, 3));

        // Testing div method
        System.out.println("\nDivision:");
```

```
        System.out.println("int / int: " + calculator.div(6, 3));
        System.out.println("double / double: " + calculator.div(5.5, 3.2));
        System.out.println("int / double: " + calculator.div(6, 3.2));
        System.out.println("double / int: " + calculator.div(5.5, 3));
    }
}
```

Output:

Addition:

int + int: 8

double + double: 8.7

int + double: 8.2

double + int: 8.5

Subtraction:

int - int: 2

double - double: 2.3

int - double: 1.7999999999999998

double - int: 2.5

Multiplication:

int * int: 15

double * double: 17.6

int * double: 16.0

double * int: 16.5

Division:

int / int: 2

double / double: 1.71875

int / double: 1.875

double / int: 1.8333333333333333

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.

Road.java

```
package assignment.day3;


//Base class
class Vehicle {
 public void startEngine() {
    System.out.println("Engine started.");
 }

 public void stopEngine() {
    System.out.println("Engine stopped.");
 }

 public void drive() {
    System.out.println("Vehicle is moving.");
 }
}
```

```
//Subclass Truck
class Truck extends Vehicle {
 public void loadCargo() {
    System.out.println("Cargo loaded into truck.");
 }
}

//Subclass Bus
class Bus extends Vehicle {
 public void boardPassengers() {
    System.out.println("Passengers boarded on bus.");
 }
}

//Subclass Car
class Car extends Vehicle {
 public void playMusic() {
    System.out.println("Music is playing in the car.");
 }
}

//Road class
public class Road {
 public static void main(String[] args) {
    // Creating objects for Truck, Bus, and Car
    Truck truck = new Truck();
    Bus bus = new Bus();
    Car car = new Car();
```

```
    // Displaying messages for Truck
    System.out.println("Truck:");
    truck.startEngine();
    truck.loadCargo();
    truck.drive();
    truck.stopEngine();

    System.out.println();

    // Displaying messages for Bus
    System.out.println("Bus:");
    bus.startEngine();
    bus.boardPassengers();
    bus.drive();
    bus.stopEngine();

    System.out.println();

    // Displaying messages for Car
    System.out.println("Car:");
    car.startEngine();
    car.playMusic();
    car.drive();
    car.stopEngine();
  }
}
```

Output:

Truck:

Engine started.

Cargo loaded into truck.

Vehicle is moving.

Engine stopped.

Bus:

Engine started.

Passengers boarded on bus.

Vehicle is moving.

Engine stopped.

Car:

Engine started.

Music is playing in the car.

Vehicle is moving.

Engine stopped.