Week 5 Homework Q21

Telmen Enkhbold

San Fransico Bay University

CE480 - Java and Internet Application

Dr. Chang, Henry

10/12/2023

# Author Note

# The Question

* 1. class Square
     + Referenes
       - [Class with numberical attribute](https://hc.labnet.sfbu.edu/~henry/sfbu/course/introjava/java_class/slide/classStruct.htm#Class%20with%20numberical%20attribute)
     + Package
     + package shape;

* + - Attributes
      * int s ; // side
    - Member functions
      * Helping function
        + int square(int i);

Return the square of i

* + - * Manager functions
        + Constructor
      * Implementor
        + void enLarge(int ds);

s is increased by ds

* + - * + int area();
        + Call the method square() to compute area of a square

* + - * + int circumference();

Return the cicumference of a square

* + - * Access functions
        + 1 get functions
        + 1 set functions
        + Predicate

isLarge();

A square is large if side is greater than 10

* 1. class Circle
     + Referenes
       - [Class with numberical attribute](https://hc.labnet.sfbu.edu/~henry/sfbu/course/introjava/java_class/slide/classStruct.htm#Class%20with%20numberical%20attribute)
     + Package
     + package shape;

* + - Attributes
      * double r; // radius
    - Member functions
      * Helping function
        + double pi();

Return the value 3.1416

* + - * Manager functions
        + Constructor
      * Implementor
        + void enLarge(double dr);

r is increased by dr

* + - * + double area();

Call the method pi() to compute area of a circle

* + - * + double circumference();

Call the method pi() to compute the cicumference of a circle

* + - * Access functions
        + 1 get function
        + 1 set function
        + Predicate

isLarge();

A circle is large if its radius is greater than 10

isAPoint();

A circle is a point if r == 1;

* 1. class Coin
     + References
       - [Class containg one layer of simple object attribute](https://hc.labnet.sfbu.edu/~henry/sfbu/course/introjava/java_class/slide/classStruct.htm#Class%20containing%20one%20layer%20of%20simple%20object%20attribute)
     + Package
     + package money;
     + // Since this class uses the Cirle and the Square classes,
     + // their package needs to be imported.
     + import shape.\*;

* + - Attributes
      * Cicle circleObj;
      * Square squareObj;
    - Member functions
      * Helping function
        + getCircleArea();

Return the area of the coin's circle portion

* + - * + getSquareArea();

Return the area of the coin's square portion

* + - * Manager functions
        + Two Constructors
        + public Coin(int s1, double r1);
        + public Coin(Square squareObj1, Circle circleObj1);

* + - * Implementor
        + area(): Use the helping functions to compute the area of a coin which is equal to the substraction of the square's area from its circle's area.
      * Access functions
        + 2 get functions
        + 2 set functions
        + Predicate

isNormal();

* + - * A coin is normal if its diameter is longer
      * than the diagnal of the square.

* 1. Test your class by creating the file TestCoin.java.
     + Import the packages
     + import money.\*;
     + import shape.\*;

* + - Create 1 object
      * coinObj whose square's side is 2 and its circle's radius is 2
    - Print the area of coinObj
    - Check if coinObj is normal

=============================ScreenShot================================

A screenshot of a computer program

Description automatically generated

A black screen with white text

Description automatically generated

A screenshot of a computer program

Description automatically generated

==============================Source Code==============================

**TestCoin.java**

import money.Coin;

import shape.Circle;

import shape.Square;

public class TestCoin {

    public static void main(String[] args) {

        // Create a coin object with a square's side of 2 and a circle's radius of 2

        Square square = new Square(2);

        Circle circle = new Circle(2);

        Coin coinObj = new Coin(square, circle);

        // Print the area of coinObj

        double coinArea = coinObj.area();

        System.out.println("The area of the coin is: " + coinArea);

        // Check if coinObj is normal

        if (coinObj.isNormal()) {

            System.out.println("The coin is normal.");

        } else {

            System.out.println("The coin is not normal.");

        }

    }

}

**Coin.java**

// Coin.java in the money package

package money;

import shape.Circle;

import shape.Square;

public class Coin {

    // Attributes

    private Circle circleObj;

    private Square squareObj;

    // Constructors

    public Coin(int s1, double r1) {

        this.circleObj = new Circle(r1);

        this.squareObj = new Square(s1);

    }

    public Coin(Square squareObj1, Circle circleObj1) {

        this.circleObj = circleObj1;

        this.squareObj = squareObj1;

    }

    // Helping function to get the area of the coin's circle portion

    public double getCircleArea() {

        return circleObj.area();

    }

    // Helping function to get the area of the coin's square portion

    public int getSquareArea() {

        return squareObj.area();

    }

    // Implementor to calculate the area of the coin

    public double area() {

        return getCircleArea() - getSquareArea();

    }

    // Access function to get the square object

    public Square getSquareObj() {

        return squareObj;

    }

    // Access function to set the square object

    public void setSquareObj(Square squareObj) {

        this.squareObj = squareObj;

    }

    // Access function to get the circle object

    public Circle getCircleObj() {

        return circleObj;

    }

    // Access function to set the circle object

    public void setCircleObj(Circle circleObj) {

        this.circleObj = circleObj;

    }

    // Predicate function to check if the coin is normal

    public boolean isNormal() {

        double diameter = 2 \* circleObj.getRadius();

        double diagonal = Math.sqrt(2) \* squareObj.getSide();

        return diameter > diagonal;

    }

}

**Cirlce.java**

// Circle.java in the shape package

package shape;

public class Circle {

    private double r; // radius

    // Constructor

    public Circle(double r) {

        this.r = r;

    }

    // Helping function to get the value of pi

    public double pi() {

        return 3.1416;

    }

    // Implementor to increase the radius by dr

    public void enLarge(double dr) {

        r += dr;

    }

    // Calculate the area of the circle using the pi() method

    public double area() {

        return pi() \* r \* r;

    }

    // Calculate the circumference of the circle using the pi() method

    public double circumference() {

        return 2 \* pi() \* r;

    }

    // Get function for radius

    public double getRadius() {

        return r;

    }

    // Set function for radius

    public void setRadius(double r) {

        this.r = r;

    }

    // Predicate to check if the circle is large

    public boolean isLarge() {

        return r > 10;

    }

    // Predicate to check if the circle is a point

    public boolean isAPoint() {

        return r == 1;

    }

}

**Square.java**

// Square.java in the shape package

package shape;

public class Square {

    private int s; // side

    // Constructor

    public Square(int s) {

        this.s = s;

    }

    // Helping function to calculate the square of a number

    public int square(int i) {

        return i \* i;

    }

    // Implementor to increase the side by ds

    public void enLarge(int ds) {

        s += ds;

    }

    // Calculate the area of the square using the square() method

    public int area() {

        return square(s);

    }

    // Calculate the circumference of the square

    public int circumference() {

        return 4 \* s;

    }

    // Get function for side

    public int getSide() {

        return s;

    }

    // Set function for side

    public void setSide(int s) {

        this.s = s;

    }

    // Predicate to check if the square is large

    public boolean isLarge() {

        return s > 10;

    }

}