|  |
| --- |
|  |
| МИНОБРНАУКИ РОССИИ |
| *Федеральное государственное бюджетное образовательное учреждение высшего образования* ***«МИРЭА – Российский технологический университет»***  **РТУ МИРЭА** |

**Институт информационных технологий (ИТ)**

**Кафедра инструментального и прикладного программного обеспечения (ИиППО)**

**Дисциплина «Программирование на языке Джава»**

**ОТЧЕТ**

**ПО ПРАКТИЧЕСКОМУ ЗАНЯТИЮ №2**

Выполнил студент группы ИНБО-02-20 Баринов.И.В.

Принял Степанов П.В.

Практические работы выполнены «\_\_\_»\_\_\_\_\_\_\_2021г.

«\_\_\_\_\_\_\_\_\_\_\_\_\_\_» «\_\_\_»\_\_\_\_\_\_\_2021г.

Отметка о выполнении

**Москва – 2021 г.**

## **Задание**

## Задание перепишите суперкласс Shape и его подклассы так как это представлено на диаграмме Circle, Rectangle and Square

## **Ход Работы**

В ходе выполнения работы были получены следующие исходные коды:

package Practise3;  
  
public abstract class Shape {  
 protected String color;  
 protected boolean filled;  
  
 public Shape() {  
 }  
  
 public Shape(String color, boolean filled) {  
 this.color = color;  
 this.filled = filled;  
 }  
  
 public String getColor() {  
 return color;  
 }  
  
 public void setColor(String color) {  
 this.color = color;  
 }  
  
 public void setFilled(boolean filled) {  
 this.filled = filled;  
 }  
  
 public boolean isFilled() {  
 return filled;  
 }  
  
 @Override  
 public String toString() {  
 return "Shape{" +  
 "color='" + color + '\'' +  
 ", filled=" + filled +  
 '}';  
 }  
  
 abstract double getArea();  
 abstract double getPerimeter();  
}

package Practise3;  
  
public class Circle extends Shape {  
 protected double radius;  
  
 public Circle(double radius, String color, boolean filled) {  
 super(color, filled);  
 this.radius = radius;  
 }  
  
 public Circle(double radius) {  
 this.radius = radius;  
 }  
  
 public double getRadius() {  
 return radius;  
 }  
  
 public void setRadius(double radius) {  
 this.radius = radius;  
 }  
  
 @Override  
 public double getArea() {  
 return Math.*PI* \* radius \* radius;  
 }  
  
 @Override  
 public double getPerimeter() {  
 return 2 \* Math.*PI* \* radius;  
  
 }  
}

package Practise3;  
  
 public class Rectangle extends Shape {  
 protected  
 double width;  
 protected  
 double length;  
  
 public Rectangle() {  
 }  
  
 public Rectangle(double width, double length) {  
 this.width = width;  
 this.length = length;  
 }  
  
 public Rectangle(String color, boolean filled, double width, double length) {  
 super(color, filled);  
 this.width = width;  
 this.length = length;  
 }  
  
 public double getLength() {  
 return length;  
 }  
  
 public void setLength(double length) {  
 this.length = length;  
 }  
  
 @Override  
 public String toString() {  
 return "Rectangle{" +  
 "width=" + width +  
 ", length=" + length +  
 '}';  
 }  
  
 @Override  
 public double getArea() {  
 return width\*length;  
 }  
 @Override  
 public double getPerimeter() {  
 return (width+length)\*2;  
  
 }  
  
 }

package Practise3;  
  
public class Square extends Rectangle {  
 protected  
 double side;  
  
 public Square() {  
 }  
  
 public Square(double side) {  
 this.side = side;  
 }  
  
 public Square(String color, boolean filled, double side) {  
 this.color = color;  
 this.filled = filled;  
 this.side = side;  
 }  
  
 public double getSide() {  
 return side;  
 }  
  
 public void setSide(double side) {  
 this.side = side;  
 }  
  
 public void setWidth(double width) {  
 this.width = width;  
 }  
  
 public void setLength(double length) {  
 this.length = length;  
 }  
  
 @Override  
 public String toString() {  
 return "Square{" +  
 "color='" + color + '\'' +  
 ", filled=" + filled +  
 ", side=" + side +  
 '}';  
 }  
  
 @Override  
 public double getArea() {  
 return side \* side;  
 }  
  
 @Override  
 public double getPerimeter() {  
 return (side) \* 4;  
  
 }  
}

## **Вывод**

Получены знания по работе с абстрактным классом

GitHub: https://github.com/BarinovIvan/RuPractises/tree/master/src/Practise3/exercise2