LAPORAN PRAKTIKUM

PEMOGRAMAN BERORIENTASI OBJECT

“Inheritance, Abstract Class and Interface”

Disusun oleh :

Alifah Fisalsabilawati

201511035

2B – D3 Teknik Informatika

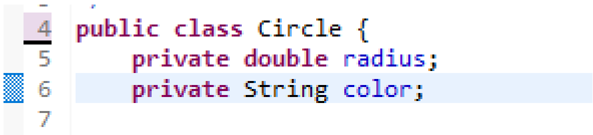
Jurusan Teknik Komputer dan Informatika

Program Studi D3 Teknik Informatika

Politeknik Negeri Bandung

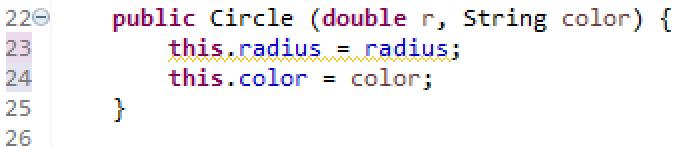
Exercise 1

- **Task 1.1**
  Menambahkan variable color bertipe string



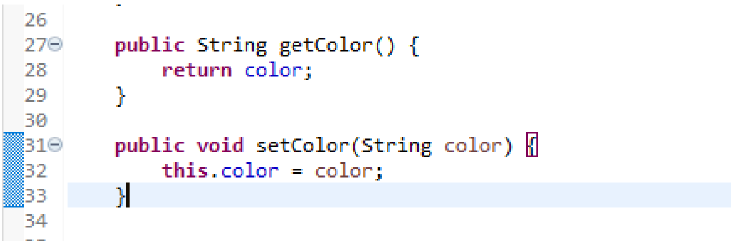
```java
public class Circle {
    private double radius;
    private String color;
```

Menambahkan constructor dengan dua paraemter double r dan string color



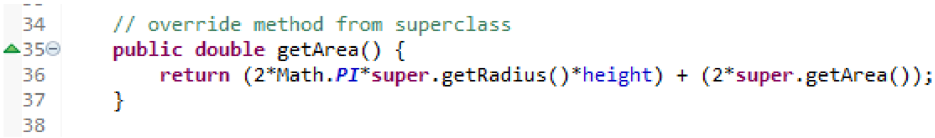
```java
    public Circle (double r, String color) {
        this.radius = radius;
        this.color = color;
    }
```

Menambahkan method getter and setter untuk String color



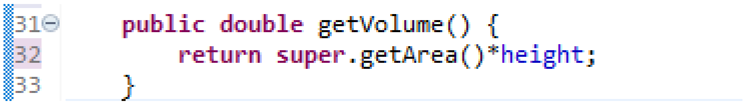
```java
    public String getColor() {
        return color;
    }

    public void setColor(String color) {
        this.color = color;
    }
```

- **Task 1.2**
  Membuat overide dari method getArea dan memasukkan formula untuk menghitung area dari cylinder pada method tersebut



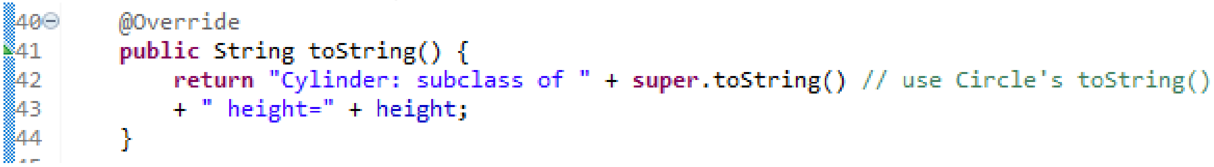
```java
    // override method from superclass
    public double getArea() {
        return (2*Math.PI*super.getRadius()*height) + (2*super.getArea());
    }
```

Membenarkan syntaks method getVolume()



```java
    public double getVolume() {
        return super.getArea()*height;
    }
```

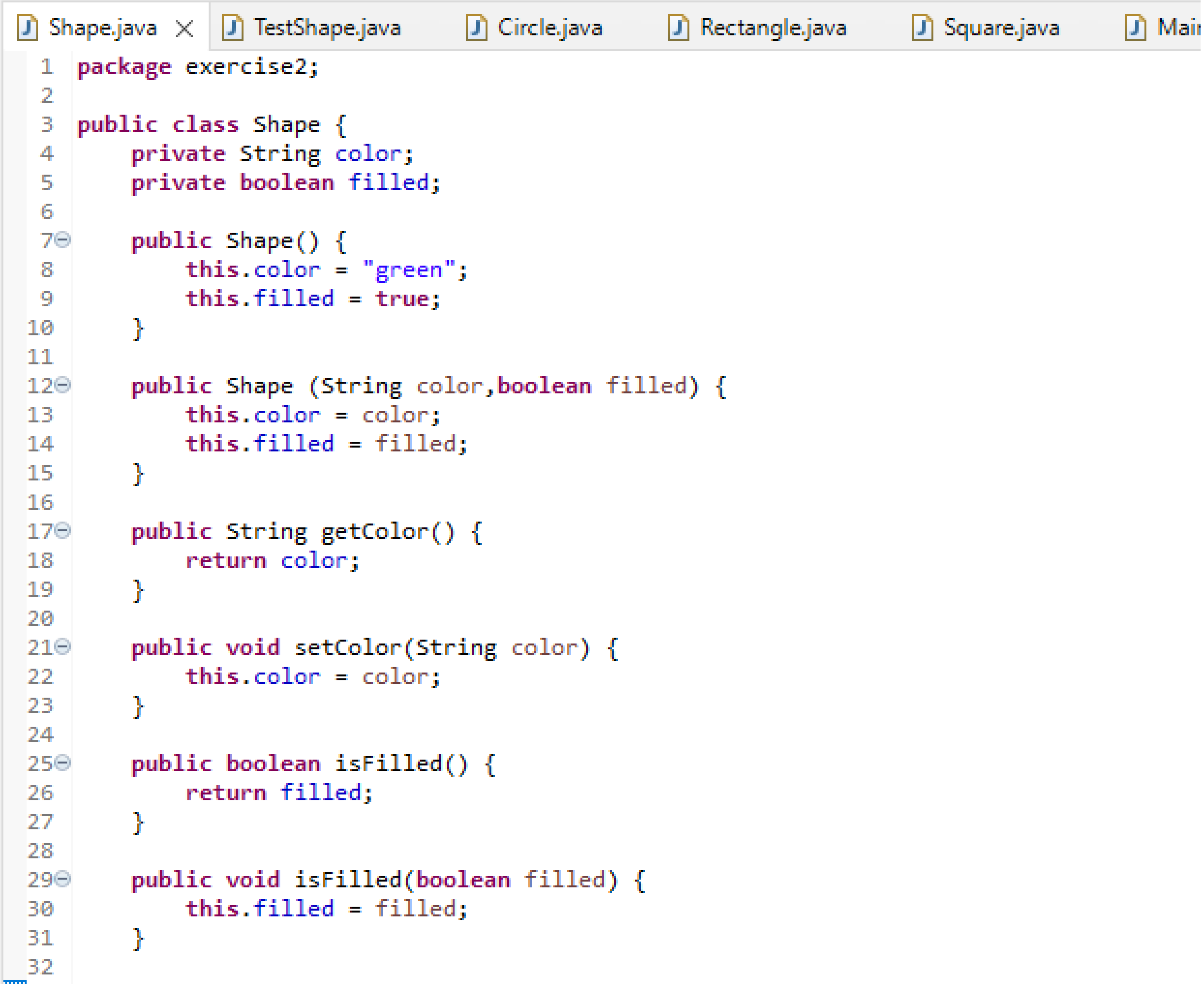
- **Task 1.3**
  Menambahkan method toString di Cylinder class sebagai override karena di subclass nya sudah terdapat method toString



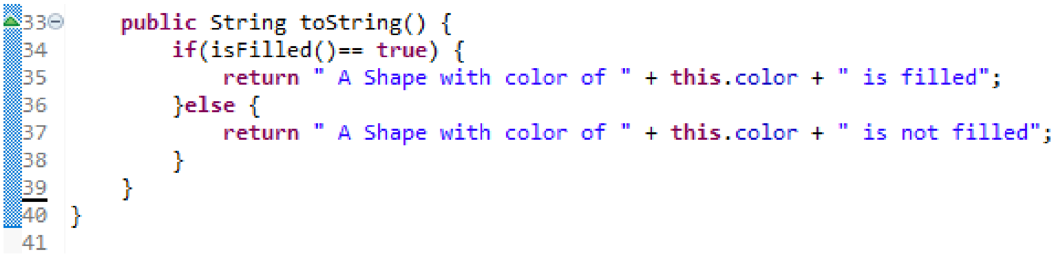
```java
    @Override
    public String toString() {
        return "Cylinder: subclass of " + super.toString() // use Circle's toString()
        + " height=" + height;
    }
```

Exercise 2

1. Membuat class Shape.java yang berisi atribut dan method sesuai dengan perintah

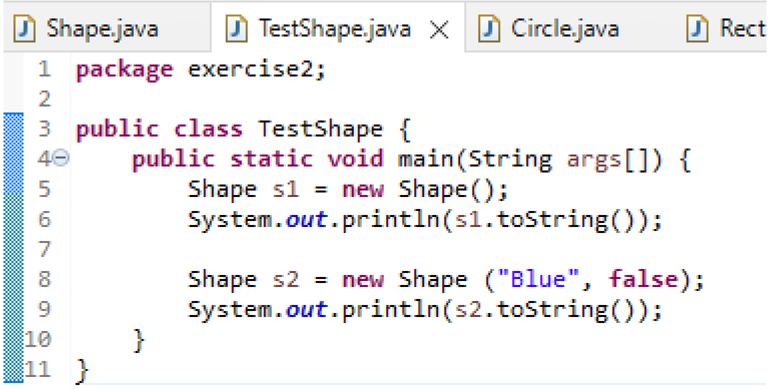


```java
package exercise2;

public class Shape {
    private String color;
    private boolean filled;

    public Shape() {
        this.color = "green";
        this.filled = true;
    }

    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 boolean isFilled() {
        return filled;
    }

    public void isFilled(boolean filled) {
        this.filled = filled;
    }

```



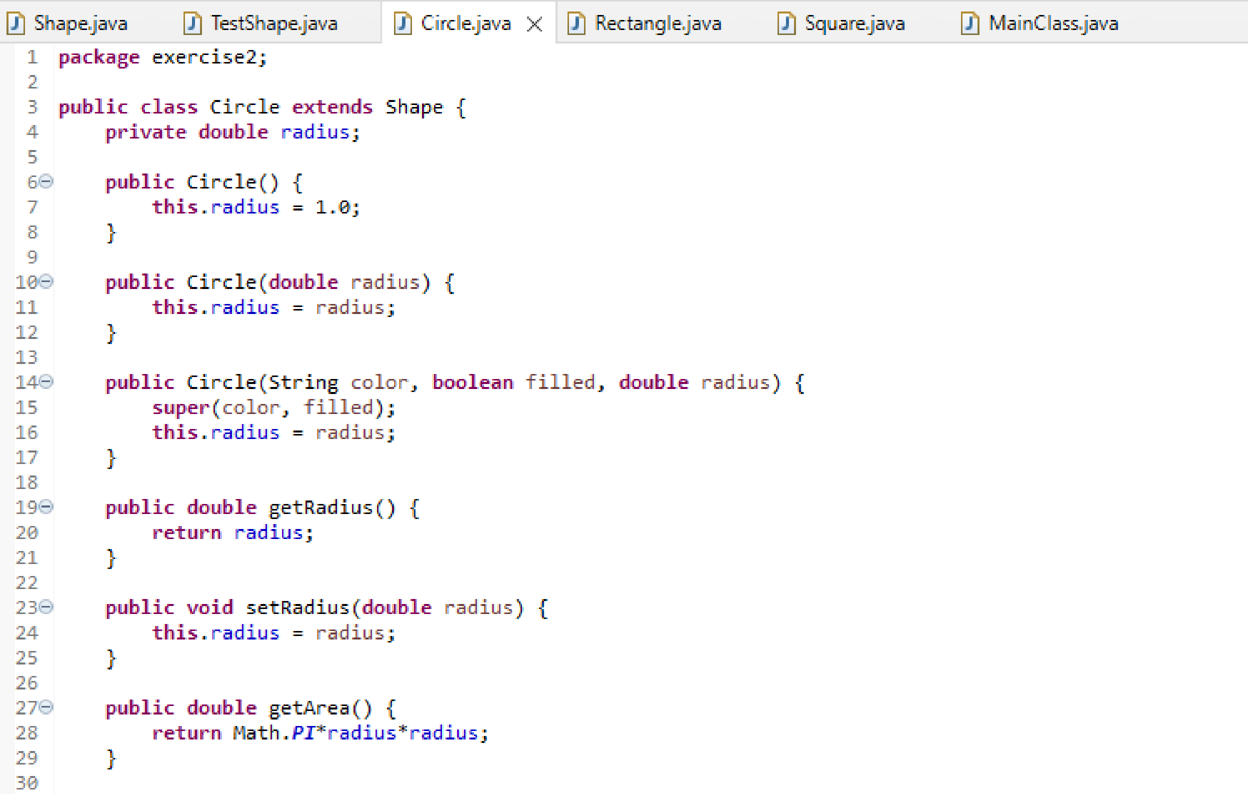
```java
    public String toString() {
        if(isFilled()== true) {
            return " A Shape with color of " + this.color + " is filled";
        }else {
            return " A Shape with color of " + this.color + " is not filled";
        }
    }
}

```

2. Membuat class TestProgram.java untuk melakukan test pada method yang sudah di buat pada class Shape.java

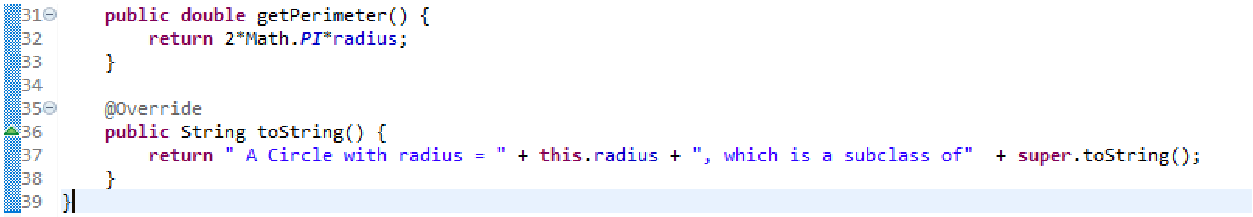


```java
package exercise2;

public class TestShape {
    public static void main(String args[]) {
        Shape s1 = new Shape();
        System.out.println(s1.toString());

        Shape s2 = new Shape ("Blue", false);
        System.out.println(s2.toString());
    }
}
```

3. Membuat Subclass Circle.java yang berisi atribut dan method sesuai dengan perintah

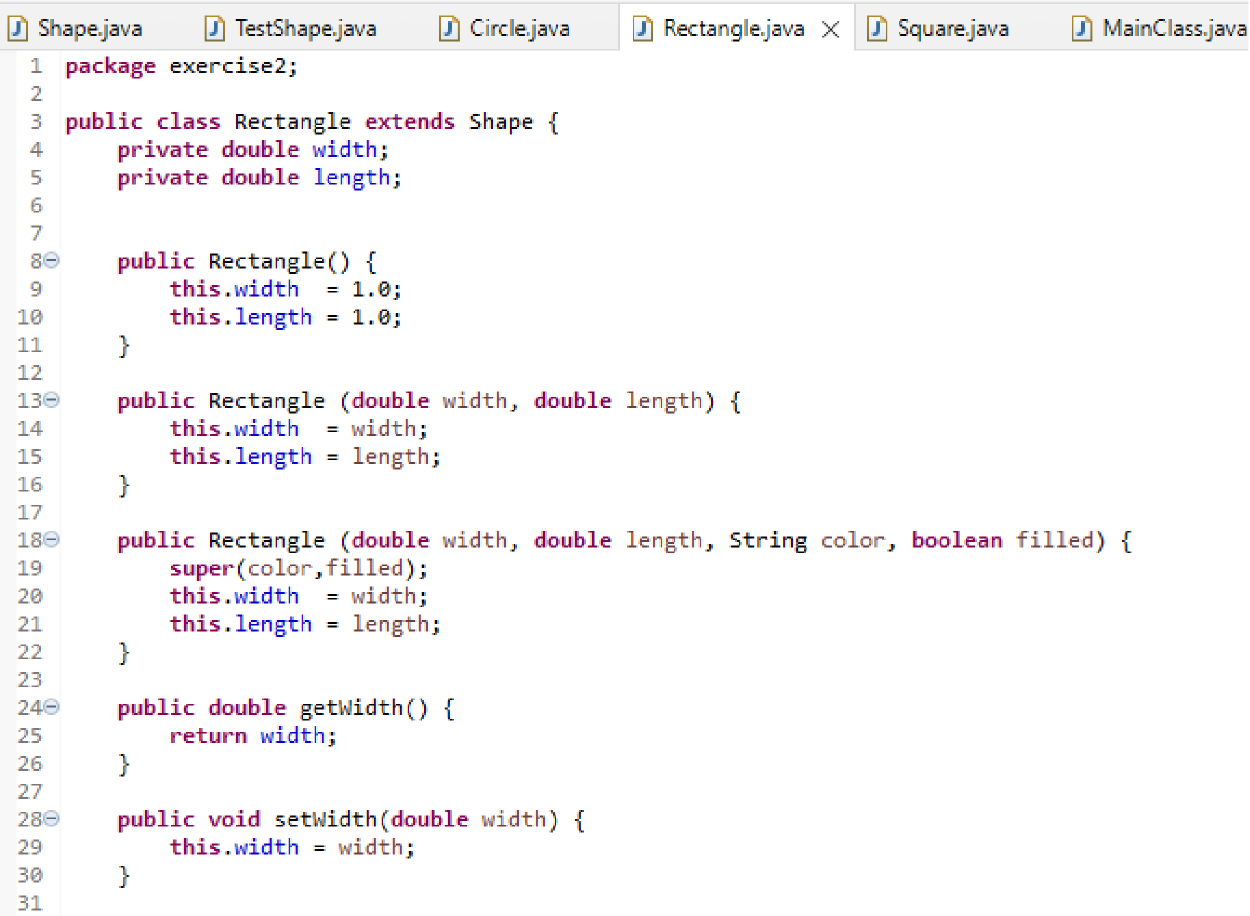


```java
package exercise2;

public class Circle extends Shape {
    private double radius;

    public Circle() {
        this.radius = 1.0;
    }

    public Circle(double radius) {
        this.radius = radius;
    }

    public Circle(String color, boolean filled, double radius) {
        super(color, filled);
        this.radius = radius;
    }

    public double getRadius() {
        return radius;
    }

    public void setRadius(double radius) {
        this.radius = radius;
    }

    public double getArea() {
        return Math.PI*radius*radius;
    }

```

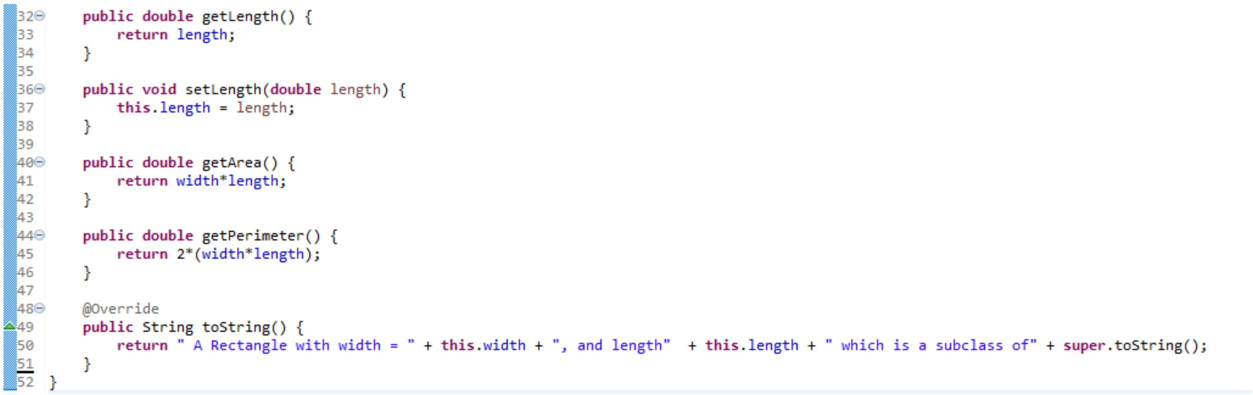


```java
    public double getPerimeter() {
        return 2*Math.PI*radius;
    }

    @Override
    public String toString() {
        return " A Circle with radius = " + this.radius + ", which is a subclass of"  + super.toString();
    }
}
```

4. Membuat Subclass Rectangle.java yang berisi atribut dan method sesuai dengan perintah

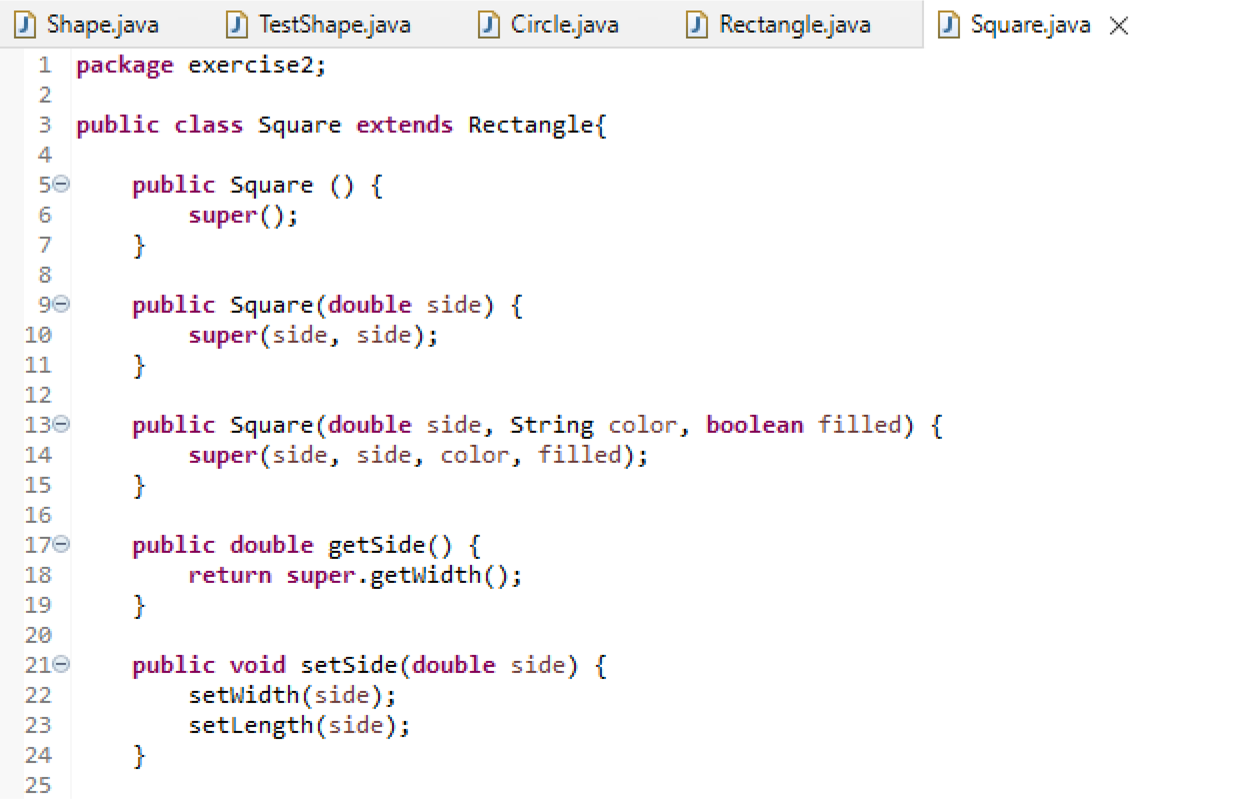


```java
package exercise2;

public class Rectangle extends Shape {
    private double width;
    private double length;


    public Rectangle() {
        this.width  = 1.0;
        this.length = 1.0;
    }

    public Rectangle (double width, double length) {
        this.width  = width;
        this.length = length;
    }

    public Rectangle (double width, double length, String color, boolean filled) {
        super(color,filled);
        this.width  = width;
        this.length = length;
    }

    public double getWidth() {
        return width;
    }

    public void setWidth(double width) {
        this.width = width;
    }

```

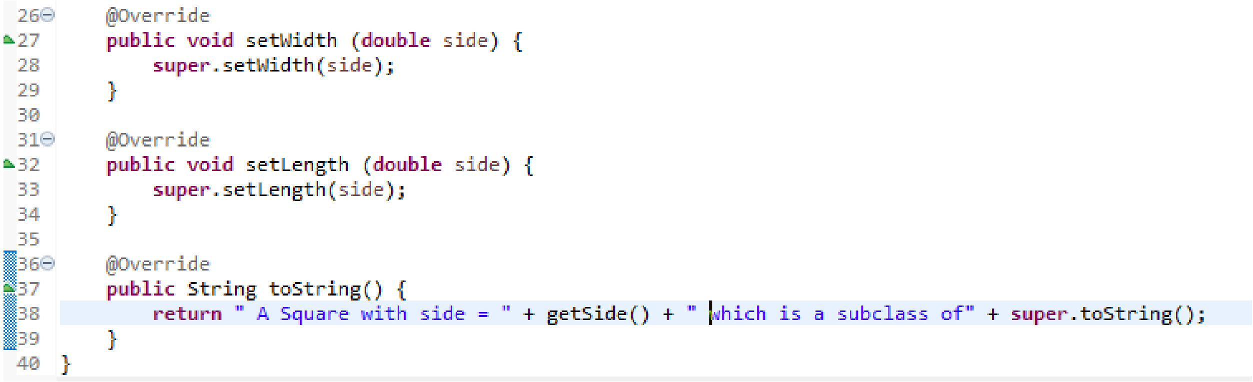


```java
    public double getLength() {
        return length;
    }

    public void setLength(double length) {
        this.length = length;
    }

    public double getArea() {
        return width*length;
    }

    public double getPerimeter() {
        return 2*(width*length);
    }

    @Override
    public String toString() {
        return " A Rectangle with width = " + this.width + ", and length"  + this.length + " which is a subclass of" + super.toString();
    }
}
```

5. Membuat Subclass Square.java yang berisi atribut dan method sesuai dengan perintah

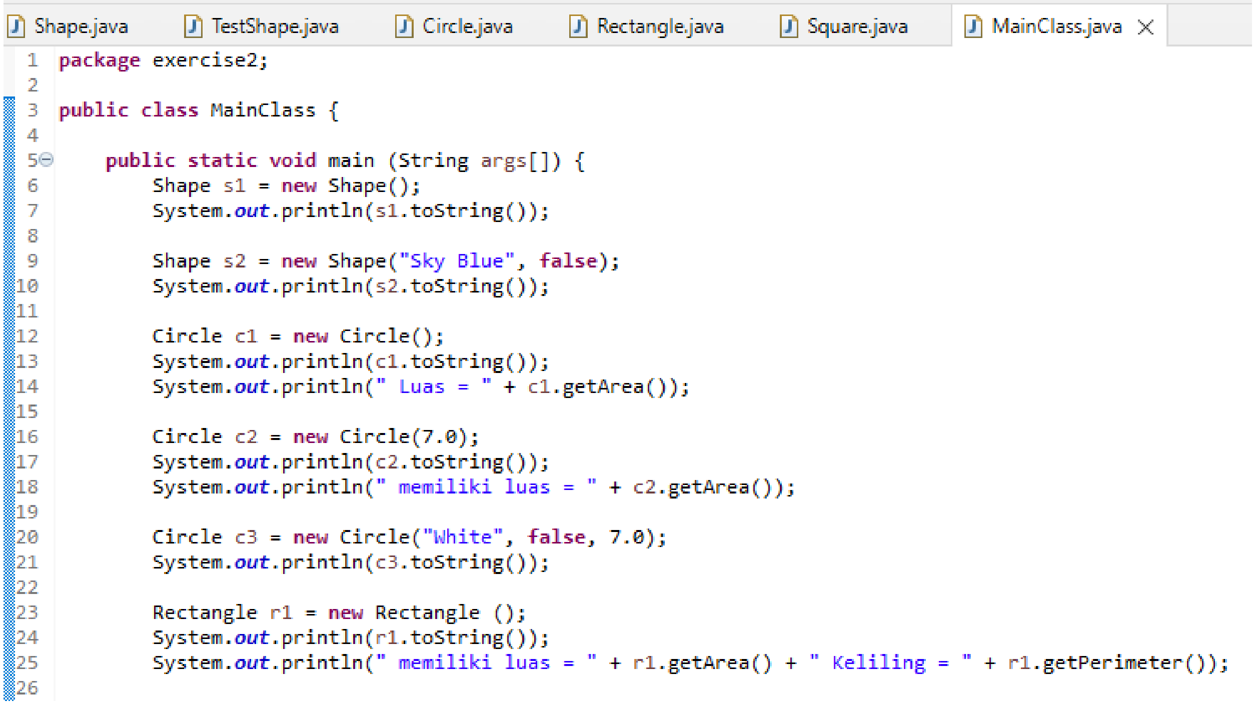


```java
package exercise2;

public class Square extends Rectangle{

    public Square () {
        super();
    }

    public Square(double side) {
        super(side, side);
    }

    public Square(double side, String color, boolean filled) {
        super(side, side, color, filled);
    }

    public double getSide() {
        return super.getWidth();
    }

    public void setSide(double side) {
        setWidth(side);
        setLength(side);
    }

```

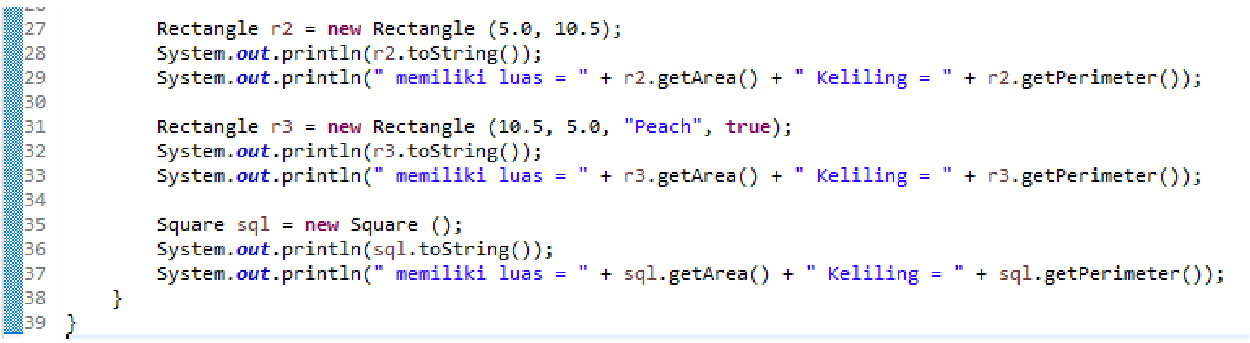


```java
    @Override
    public void setWidth (double side) {
        super.setWidth(side);
    }

    @Override
    public void setLength (double side) {
        super.setLength(side);
    }

    @Override
    public String toString() {
        return " A Square with side = " + getSide() + " which is a subclass of" + super.toString();
    }
}
```

6. Membuat MainClass.java untuk memanggil semua class yang sudah di buat tadi (superclass dan subclass)

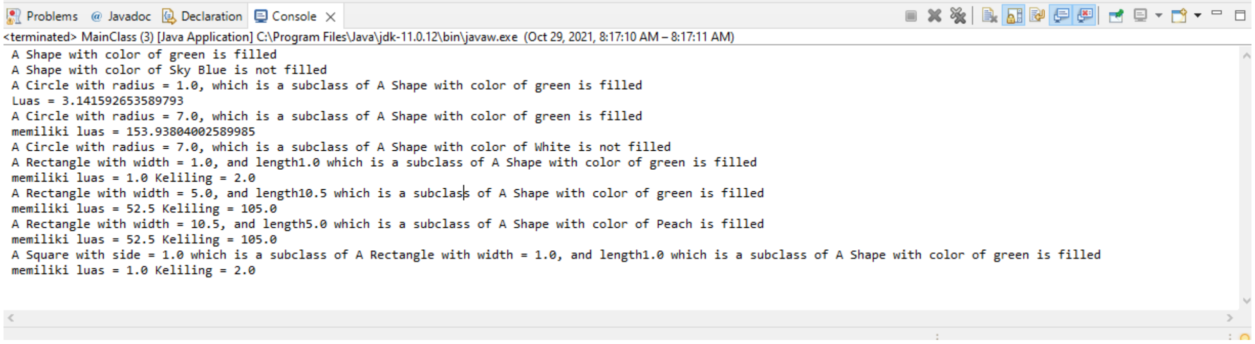


```java
package exercise2;

public class MainClass {

    public static void main (String args[]) {
        Shape s1 = new Shape();
        System.out.println(s1.toString());

        Shape s2 = new Shape("Sky Blue", false);
        System.out.println(s2.toString());

        Circle c1 = new Circle();
        System.out.println(c1.toString());
        System.out.println(" Luas = " + c1.getArea());

        Circle c2 = new Circle(7.0);
        System.out.println(c2.toString());
        System.out.println(" memiliki luas = " + c2.getArea());

        Circle c3 = new Circle("White", false, 7.0);
        System.out.println(c3.toString());

        Rectangle r1 = new Rectangle ();
        System.out.println(r1.toString());
        System.out.println(" memiliki luas = " + r1.getArea() + " Keliling = " + r1.getPerimeter());

```



```java
        Rectangle r2 = new Rectangle (5.0, 10.5);
        System.out.println(r2.toString());
        System.out.println(" memiliki luas = " + r2.getArea() + " Keliling = " + r2.getPerimeter());

        Rectangle r3 = new Rectangle (10.5, 5.0, "Peach", true);
        System.out.println(r3.toString());
        System.out.println(" memiliki luas = " + r3.getArea() + " Keliling = " + r3.getPerimeter());

        Square sql = new Square ();
        System.out.println(sql.toString());
        System.out.println(" memiliki luas = " + sql.getArea() + " Keliling = " + sql.getPerimeter());
    }
}
```

7. Hasil akhir program



```
<terminated> MainClass (3) [Java Application] C:\Program Files\Java\jdk-11.0.12\bin\javaw.exe  (Oct 29, 2021, 8:17:10 AM – 8:17:11 AM)
A Shape with color of green is filled
A Shape with color of Sky Blue is not filled
A Circle with radius = 1.0, which is a subclass of A Shape with color of green is filled
Luas = 3.141592653589793
A Circle with radius = 7.0, which is a subclass of A Shape with color of green is filled
memiliki luas = 153.93804002589985
A Circle with radius = 7.0, which is a subclass of A Shape with color of White is not filled
A Rectangle with width = 1.0, and length1.0 which is a subclass of A Shape with color of green is filled
memiliki luas = 1.0 Keliling = 2.0
A Rectangle with width = 5.0, and length10.5 which is a subclass of A Shape with color of green is filled
memiliki luas = 52.5 Keliling = 105.0
A Rectangle with width = 10.5, and length5.0 which is a subclass of A Shape with color of Peach is filled
memiliki luas = 52.5 Keliling = 105.0
A Square with side = 1.0 which is a subclass of A Rectangle with width = 1.0, and length1.0 which is a subclass of A Shape with color of green is filled
memiliki luas = 1.0 Keliling = 2.0
```