knóldus

SOLID PRINCIPLES ASSIGNMENT

knólˣ

Question

Identify how the following code violates the Liskov Substitution Principle? Also, correct the code in order it to be correct as per the Liskov Substitution Principle.

Code:

```
class Rectangle{
  int m_width;
  int m_height;

  public void setWidth(int width){
    m_width = width;
  }

  public void setHeight(int h) {
    m_height = h;
  }

  public int getWidth() {
    return m_width;
  }

  public int getHeight() {
    return m_height;
  }

  public int getArea() {
    return m_width * m_height;
  }
}

class Square extends Rectangle {
  public void setWidth (int width) {
    m_width = width;
    m_height = width;
  }

  public void setHeight (int height) {
    m_width = height;
    m_height = height;
  }
}
```

Solution

In the given Code it violates the Liskov Substitution Principle. Liskop Substitution Principle state that the child class behave like parent class.

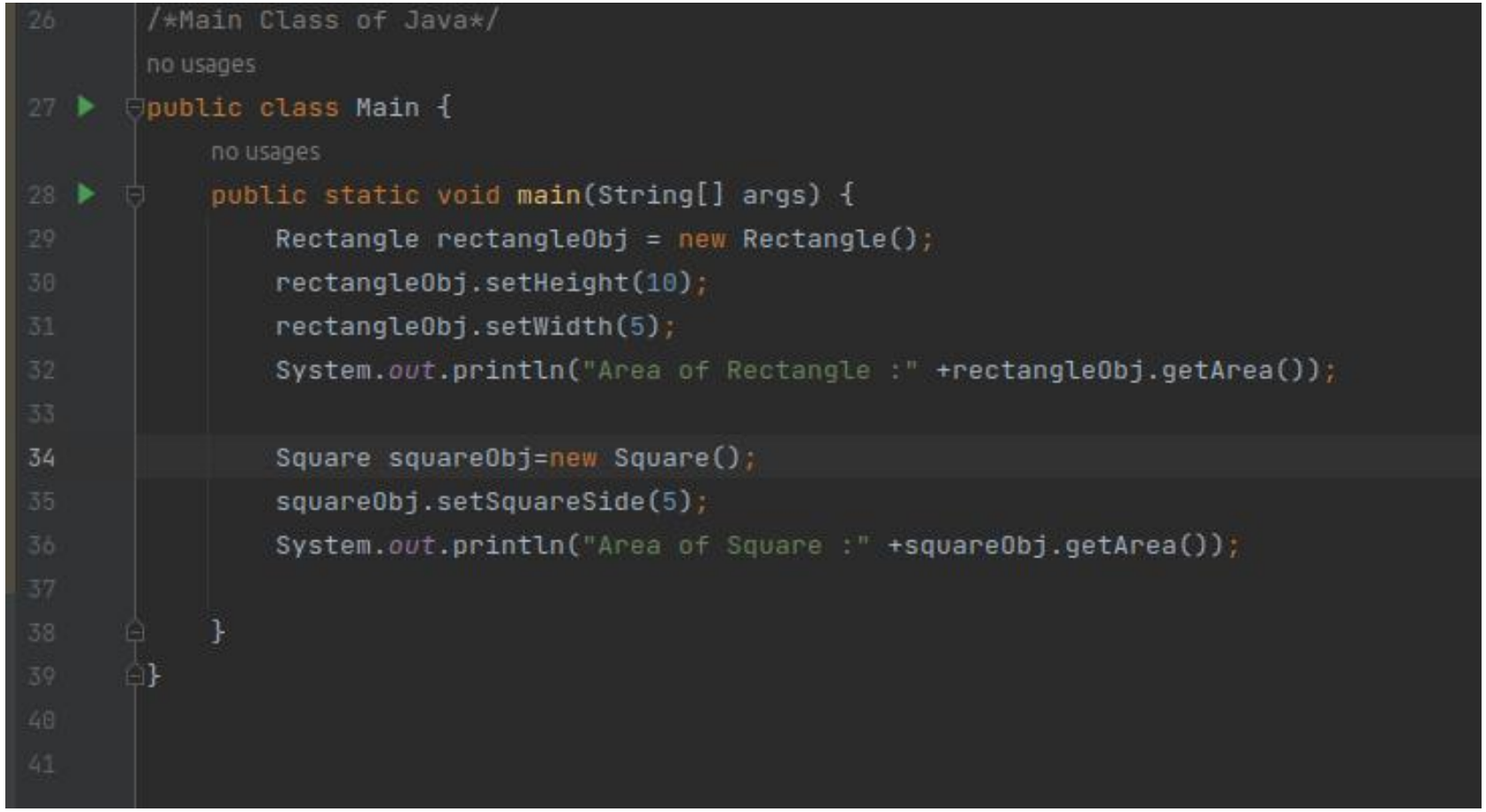
- This code violates the behaviour of Rectangle class. As we know that LSP principle states that any instance of subtype should be able to used in place of parent class.
- However in this code an object of **Square class has both set width and set height**, it **will affect** the code which is **not** follow the **Liskov Substitution Principle**.
- That's why we have to add the **setSquareSide** in the **Square class** and pass only single parameter in it.

Solution

knóldus

```
3 usages  1 inheritor
class Rectangle{
    4 usages
    int width;
    4 usages
    int height;
    1 usage
    public void setWidth(int width){
        this.width = width;
    }
    1 usage
    public void setHeight(int height) {
        this.height = height;
    }
    no usages
    public int getWidth() {
        return this.width;
    }
    no usages
    public int getHeight() {
        return this.height;
    }
    2 usages
    public int getArea() {
        return this.width * this.height;
    }
}
2 usages
class Square extends Rectangle{
    1 usage
    public void setSquareSide(int side){
        width = side;
        height = side;
    }
}
```

Solution

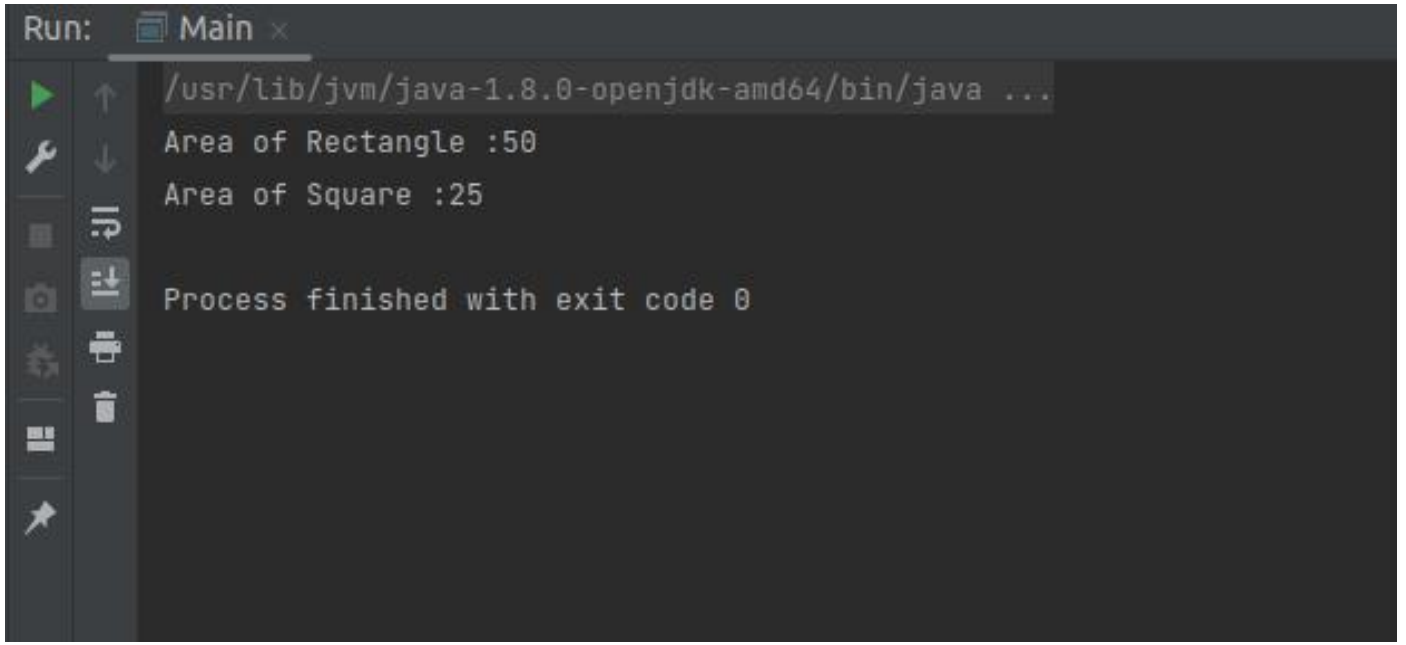


```java
/*Main Class of Java*/
no usages
public class Main {
    no usages
    public static void main(String[] args) {
        Rectangle rectangleObj = new Rectangle();
        rectangleObj.setHeight(10);
        rectangleObj.setWidth(5);
        System.out.println("Area of Rectangle :" +rectangleObj.getArea());

        Square squareObj=new Square();
        squareObj.setSquareSide(5);
        System.out.println("Area of Square :" +squareObj.getArea());

    }
}
```

Output


Run:
Main
/usr/lib/jvm/java-1.8.0-openjdk-amd64/bin/java ...
Area of Rectangle :50
Area of Square :25
Process finished with exit code 0

knóldus

THANK YOU

knólˣ