Practice-8. Inheritance in classes

Lecturer: T.Kuchkorov

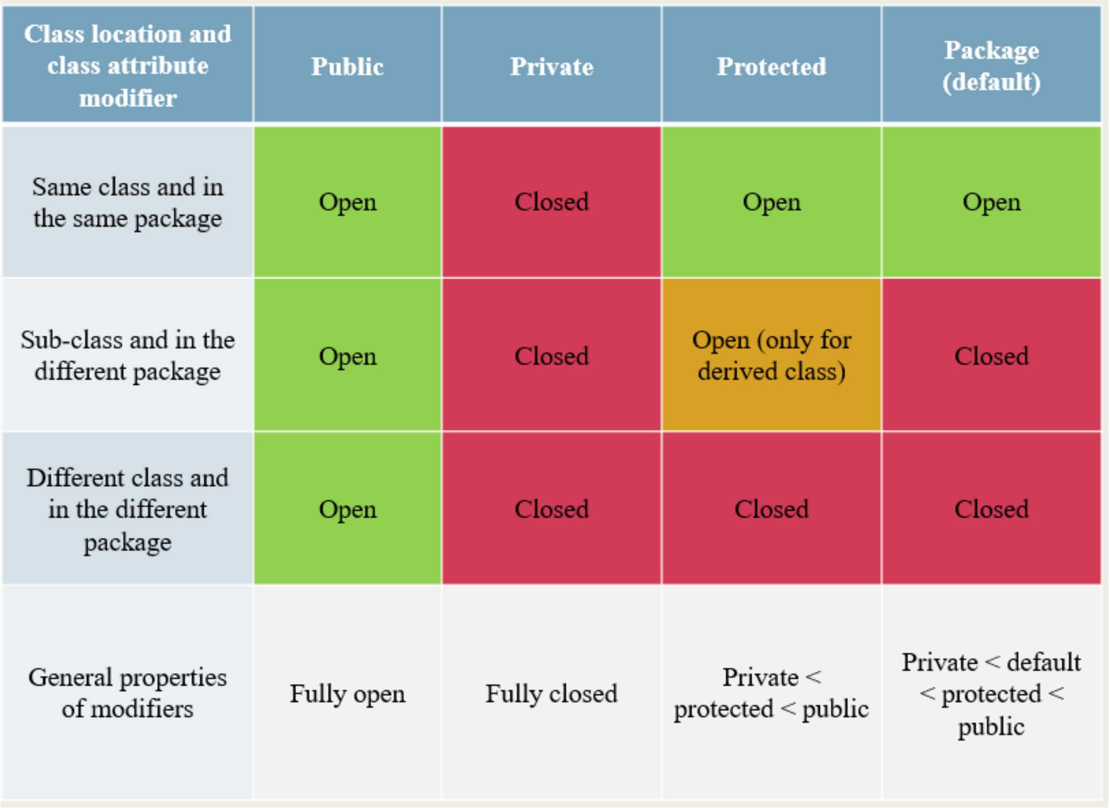
OOP | 08.04.2022

Using inheritance between classes in the Java programming language. Protected modifier

In this activity, student will learn creation of derived class in Java, types of inheritance, principles of connection between derived and base class, as well as using protected modifier between inherited classes. Also, students will learn managing access to protected attributes of base class from inherited and non-inherited classes which are located in different packages.

There are differences between access modifiers (*public, private, protected, default*) depending on whether the base class and the derived class are located in the same or different packages (see the table-1).

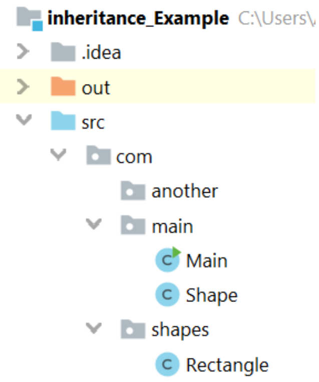
(Table-1). Difference between access modifiers



Class location and class attribute modifier	Public	Private	Protected	Package (default)
Same class and in the same package	Open	Closed	Open	Open
Sub-class and in the different package	Open	Closed	Open (only for derived class)	Closed
Different class and in the different package	Open	Closed	Closed	Closed
General properties of modifiers	Fully open	Fully closed	Private < protected < public	Private < default < protected < public

Classwork sample.

Create classes in the following hierarchy:



```
inheritance_Example C:\Users\
  > .idea
  > out
  v src
      v com
            another
          v main
                Main
                Shape
          v shapes
                Rectangle
```

There are 2 packages in the project that *main* package should contain **Main** and **Shape** classes, and *shapes* package should contain Rectangle class.

Shape.java (in the **main package**)

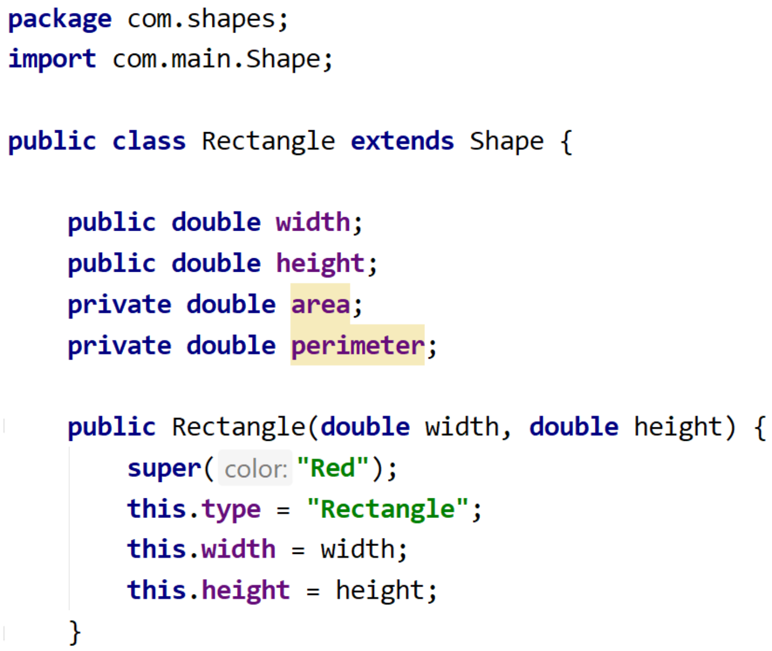
```java
package com.main;

public class Shape {
    protected String color;
    protected String type;

    public Shape(String color){
        this.color = color;
        this.type = "unknown shape";
    }

    public void show(){
        System.out.println("Shape of color: " + this.color);
        System.out.println("Type of shape: " + this.type);
    }
}
```

Rectangle.java (in the **shapes package**)

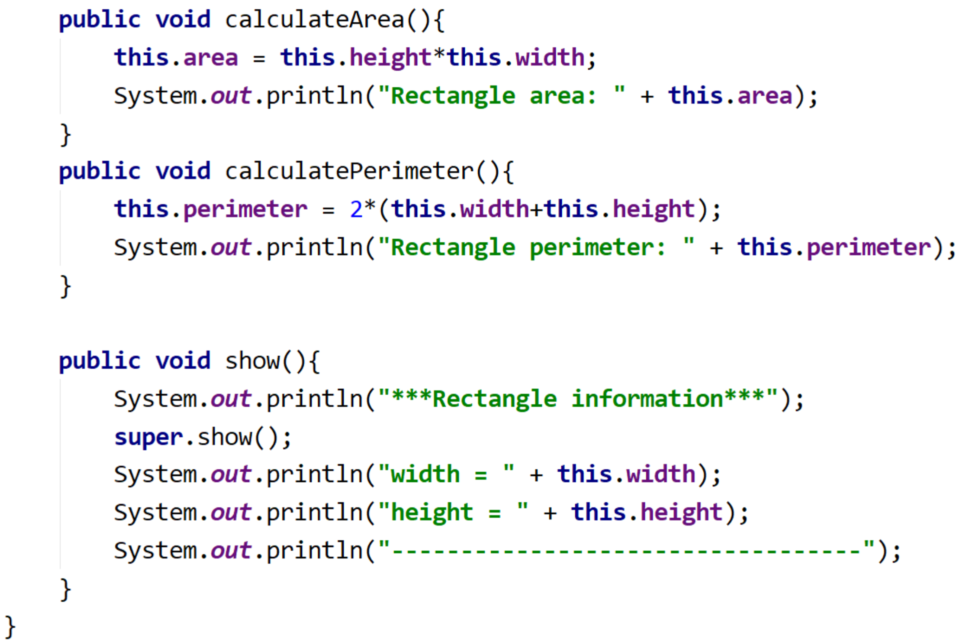


```
package com.shapes;
import com.main.Shape;

public class Rectangle extends Shape {

    public double width;
    public double height;
    private double area;
    private double perimeter;

    public Rectangle(double width, double height) {
        super( color: "Red");
        this.type = "Rectangle";
        this.width = width;
        this.height = height;
    }
```

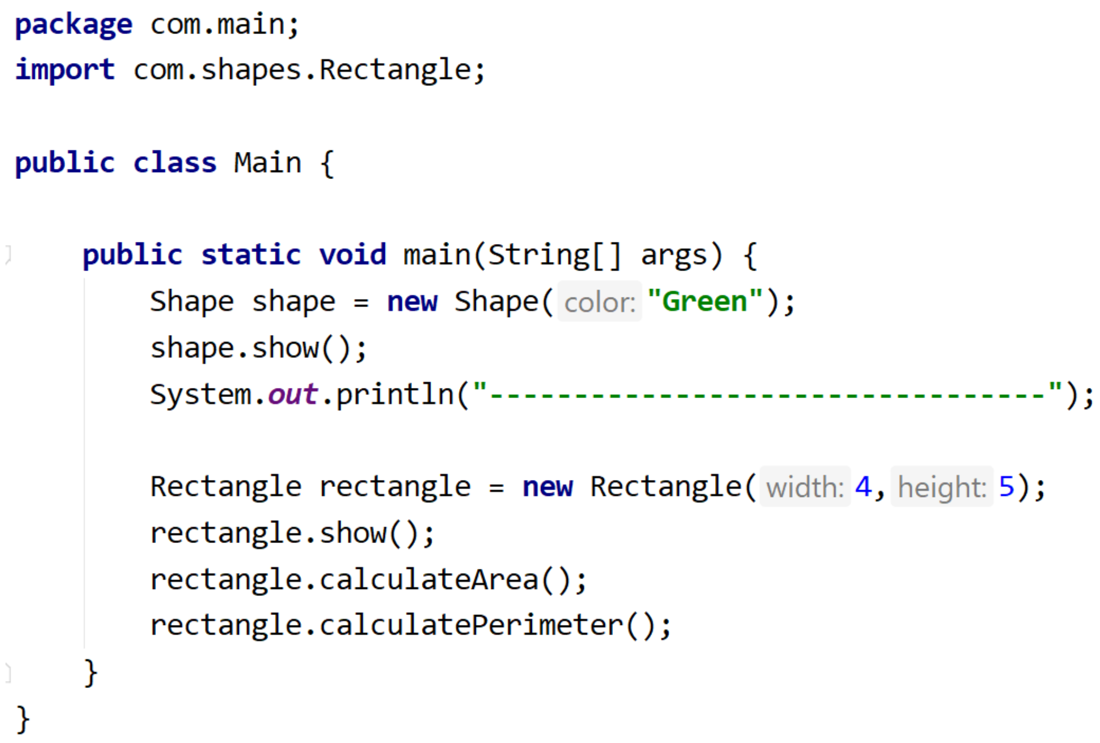


```
    public void calculateArea(){
        this.area = this.height*this.width;
        System.out.println("Rectangle area: " + this.area);
    }
    public void calculatePerimeter(){
        this.perimeter = 2*(this.width+this.height);
        System.out.println("Rectangle perimeter: " + this.perimeter);
    }

    public void show(){
        System.out.println("***Rectangle information***");
        super.show();
        System.out.println("width = " + this.width);
        System.out.println("height = " + this.height);
        System.out.println("----------------------------------");
    }
}
```

In this example Rectangle class is a derived class of Shape, so Shape class's protected attributes are accessible for Rectangle class because of it is inherited from Shape class.

Main.java (in the main package)



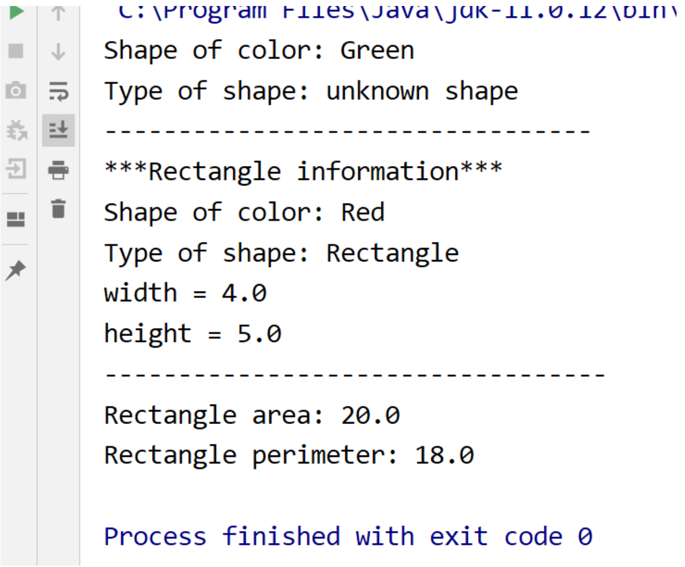
```java
package com.main;
import com.shapes.Rectangle;

public class Main {

    public static void main(String[] args) {
        Shape shape = new Shape( color: "Green");
        shape.show();
        System.out.println("--------------------------------");

        Rectangle rectangle = new Rectangle( width: 4, height: 5);
        rectangle.show();
        rectangle.calculateArea();
        rectangle.calculatePerimeter();
    }
}
```

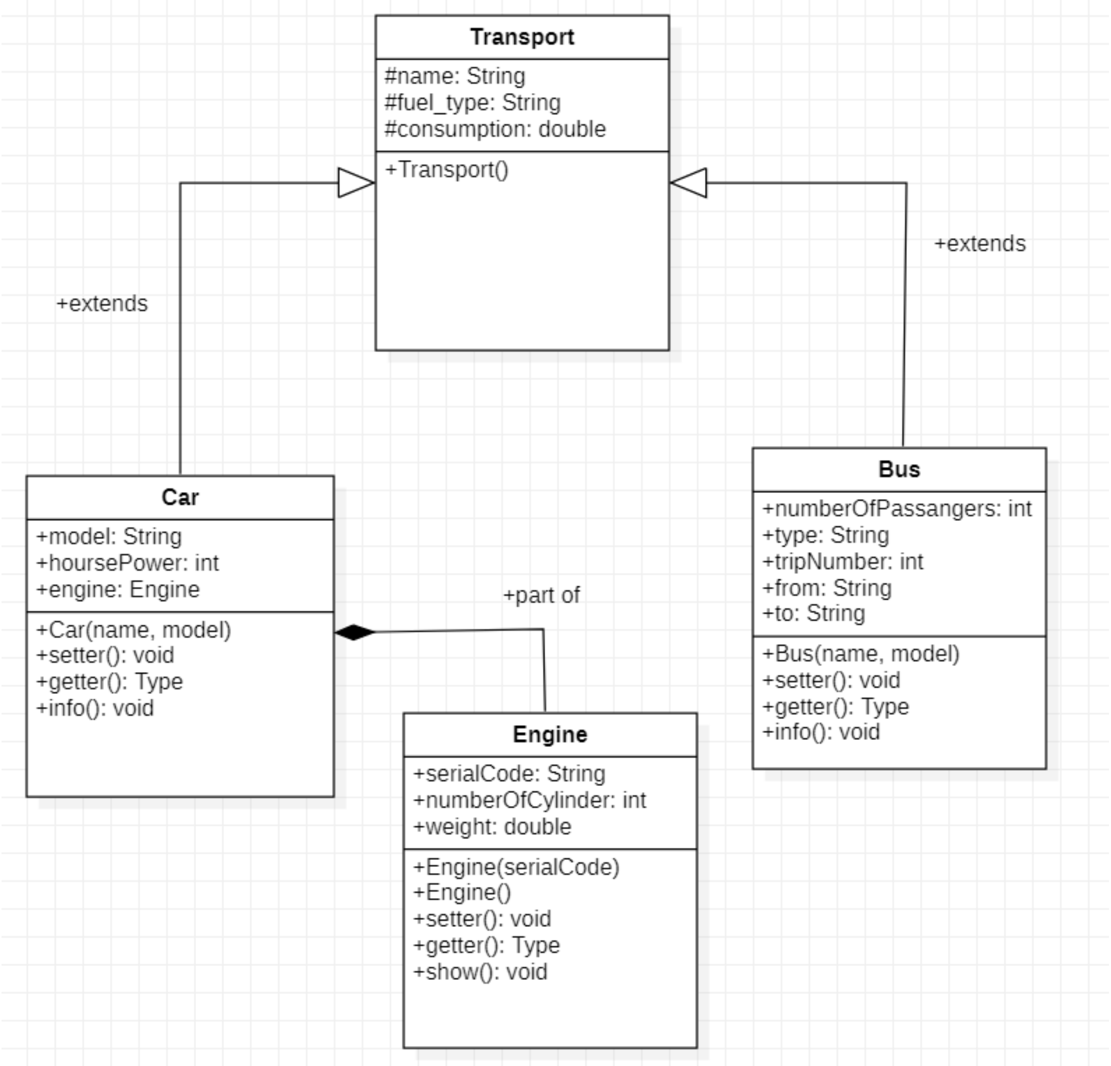
Result of the program:



```
 C:\Program Files\Java\jdk-11.0.12\bin\
Shape of color: Green
Type of shape: unknown shape
--------------------------------
***Rectangle information***
Shape of color: Red
Type of shape: Rectangle
width = 4.0
height = 5.0
---------------------------------
Rectangle area: 20.0
Rectangle perimeter: 18.0

Process finished with exit code 0
```

Task for students:



1. Intially, see the class diagram and create the classes in the same package. Regarding UML diagram Car and Bus classes are inherited from Transport class (all attributes are default). Assume that there is part of (composition) relationship between Engine and Car classes.
2. In the next stage, create following packages and move the classes to the appropriate package
   a. **main** package (Main, Transport)
   b. **trasnports** package (Car, Bus)
   c. **other** package (Engine)
3. Now Transport class attributes are not accessiable for derived (Car, Bus) classes, please change base class attributes modifier as a protected.
4. Print the result