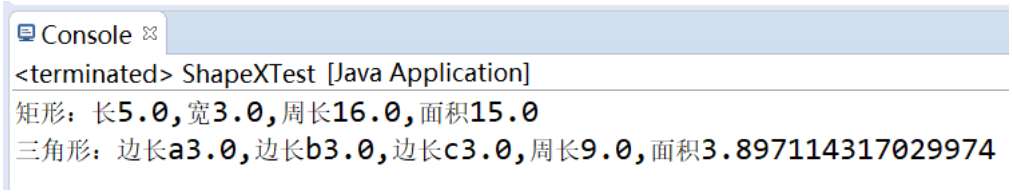
- Experiment 6 -

Create a Java project named Experiment6 and create the following classes in the project.

1. Define an interface ShapeX, which has two abstract methods: area() and perimeter().
Next, define two classes Rectangle and Triangle to implement the ShapeX interface respectively.
Finally, define a main class ShapeXTest, in the main method, create several objects, and output the perimeter and area of each object (each class must have at least one object).

[Sample]

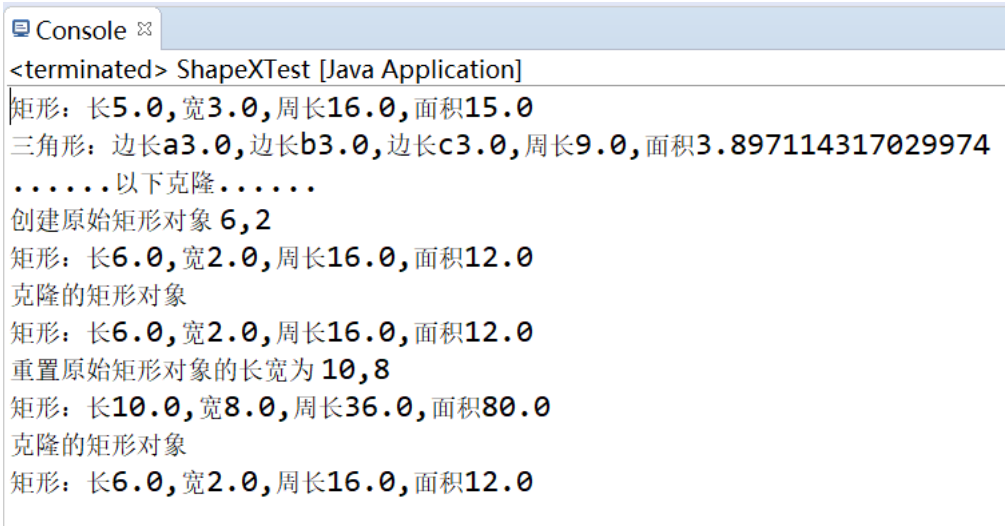


2. On the basis of above, add clone() method in class Rectangle.

//Refer to the code on 6.2.3 of the textbook.

In the main method, create an original Rectangle object, then use *original* to call clone() method to assign it to a new object variable *copy*, output the values of original object and copy object.

[Sample]

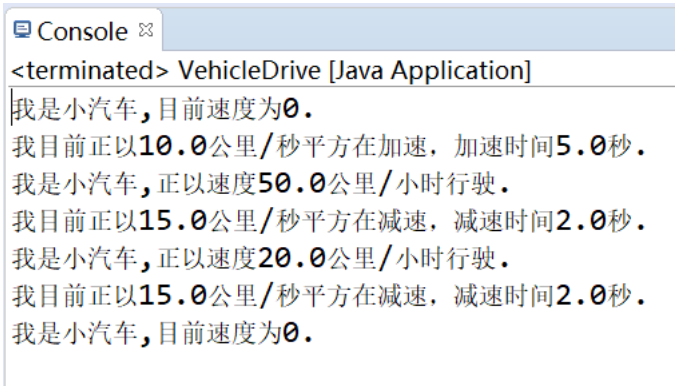


3. Given the following two interfaces and the public class VehicleDrive in the same file,

```
interface SpeedUp {
    void speedUp(double acceleration, double time);
}
interface SpeedDown {
    void speedUp(double slowDown, double time);
}
public class VehicleDrive {
    public static void main(String[] args) {
```

```
        Car car1=new Car();
        System.out.println(car1);
        car1.speedUp(10, 5);
        System.out.println(car1);
        car1.speedDown(15, 2);
        System.out.println(car1);
        car1.speedDown(15, 2);
        System.out.println(car1);
    }
}
```

add a class to the file so that the running result of your program is as follows:



4. Write a program named SanqiArtist.java to describe Sanqi artists(三栖艺人). Sanqi artists: they can act in movies, in TV dramas, and sing songs.

Implementation ideas:

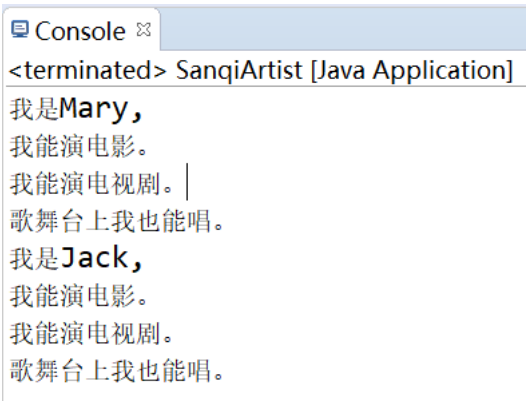
1) Define 3 interfaces

- Interface for acting in movies --- method: actInMovie
- Interface for acting in TV series --- method: actInTV
- Singing interface --- method: sing

2) Let class SanqiArtist implement 3 interfaces.

3) Create at least one object, and output the results.

[Sample]



5 Write a program which can make musicians play musical instruments. The musical instruments include piano and violin. One musician can play different instruments to produce different sounds, and many musicians can play the same type of instruments to produce wonderful sounds.

Realization ideas:

1) Define an interface Instrument, which contains a method -- makeSound();

2) Define two classes that implements Instrument: one is Piano, the other is Violin;

3) Define a class Musician for musicians, including the method play (*Instrument i*) to play various instruments;

4) Define a main class MyTest, give one musician different instruments to play; give the same kind of instrument to different musicians to play.

//**Tips:** makeSound() method prototype --- String makeSound()

[Sample]

6. Run the following codes, then answer the questions:

```
import java.awt.*;
import javax.swing.*;
public class SimpleFrameTest extends JFrame {
    public static void main(String[] args) {
        EventQueue.invokeLater( new Runnable() {
            public void run() {
                SimpleFrameTest frame = new SimpleFrameTest();
                frame.setBounds(100, 100, 400, 300);
                frame.getContentPane().setBackground(Color.BLUE);
                frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
                frame.setVisible(true);
            }
        } );
    }
}
```

```
import java.awt.*;
import javax.swing.*;
public class SimpleFrameTest extends JFrame {
    public static void main(String[] args) {
        EventQueue.invokeLater( () -> {
                SimpleFrameTest frame = new SimpleFrameTest();
                frame.setBounds(100, 100, 400, 300);
```

```
                    frame.getContentPane().setBackground(Color.ORANGE);
                    frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
                    frame.setVisible(true);
                } );
            }
        }
```

What are the two code segments marked in red above, and what are their characteristics?