实验五： 继承与抽象

## 实验目的：

1. 了解继承的概念，掌握派生类的定义。
2. 掌握派生类构造方法的执行过程。
3. 掌握方法的重载与覆盖。
4. 掌握抽象类的概念

## 实验内容：

定义一个抽象类图形类Shape，类中定义一个属性name，两个抽象方法：显示面积和周长的方法。定义两个子类：圆形类、三角形类，继承Shape类，并根据子类的特性增加自己的属性。并重写shape类中的方法。在主类中创建子类对象，设置其属性值，将子类对象的引用赋值给Shape类的对象，用上转型对象调用子类重写的方法，输出其面积与周长。

## 实验代码：

Shape.java

package com.shf.demo5;  
  
public abstract class Shape *{* public static String *name*;  
 public abstract double getGirth*()*;  
 public abstract double getArea*()*;  
*}*

Circular.java

package com.shf.demo5;  
  
import java.math.BigDecimal;  
import java.math.RoundingMode;  
  
public class Circular extends Shape*{* public String name;  
 public double radius;  
  
 public Circular*() {  
 }* public Circular*(*String name, double radius*) {* Shape.*name* = name;  
 this.radius = radius;  
 *}* public double getArea*(){* return new BigDecimal*(*Math.*PI* \* Math.*pow(*radius, 2*))*.setScale*(*2, RoundingMode.*UP)*.doubleValue*()*;  
 *}*;  
 public double getGirth*(){* return new BigDecimal*(*2 \* Math.*PI* \* radius*)*.setScale*(*2, RoundingMode.*UP)*.doubleValue*()*;  
 *}*;  
*}*

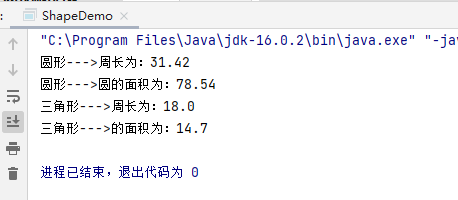
Triangle.java

package com.shf.demo5;  
  
import java.math.BigDecimal;  
import java.math.RoundingMode;  
  
public class Triangle extends Shape*{* public String name;  
 public double a;  
 public double b;  
 public double c;  
  
 public Triangle*() {  
 }* public Triangle*(*String name, double a, double b, double c*) {* Shape.*name* = name;  
 this.a = a;  
 this.b = b;  
 this.c = c;  
 *}* public double getArea*(){* double p = getGirth*()* / 2.0;  
 return new BigDecimal*(*Math.*sqrt(*p\**(*p-a*)*\**(*p-b*)*\**(*p-c*)))*.setScale*(*2, RoundingMode.*UP)*.doubleValue*()*;  
 *}*;  
 public double getGirth*(){* return a + b + c;  
 *}*;  
*}*

ShapeDemo.java

package com.shf.demo5;  
  
public class ShapeDemo *{* public static void main*(*String*[]* args*) {* Shape circular = new Circular*(*"圆形",5*)*;  
 System.*out*.println*(*Shape.*name*+"--->周长为："+circular.getGirth*())*;  
 System.*out*.println*(*Shape.*name*+"--->圆的面积为："+circular.getArea*())*;  
  
 Shape triangle = new Triangle*(*"三角形",5,6,7*)*;  
 System.*out*.println*(*Shape.*name* + "--->周长为："+triangle.getGirth*())*;  
 System.*out*.println*(*Shape.*name* + "--->的面积为："+triangle.getArea*())*;  
 *}  
}*

## 实验结果截图：



## 实验小结：

通过本次实验我掌握了继承的概念，掌握方法的重载与覆盖，掌握抽象类的概念，学会向上转型的多态性。抽象类和接口都不能被实例化，抽象类的子类必须给出抽象类方法的具体实现，除非该子类也是抽象类。