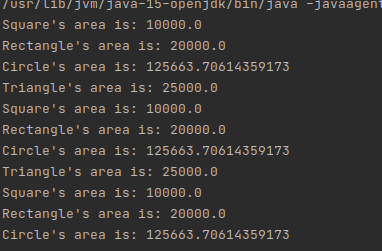
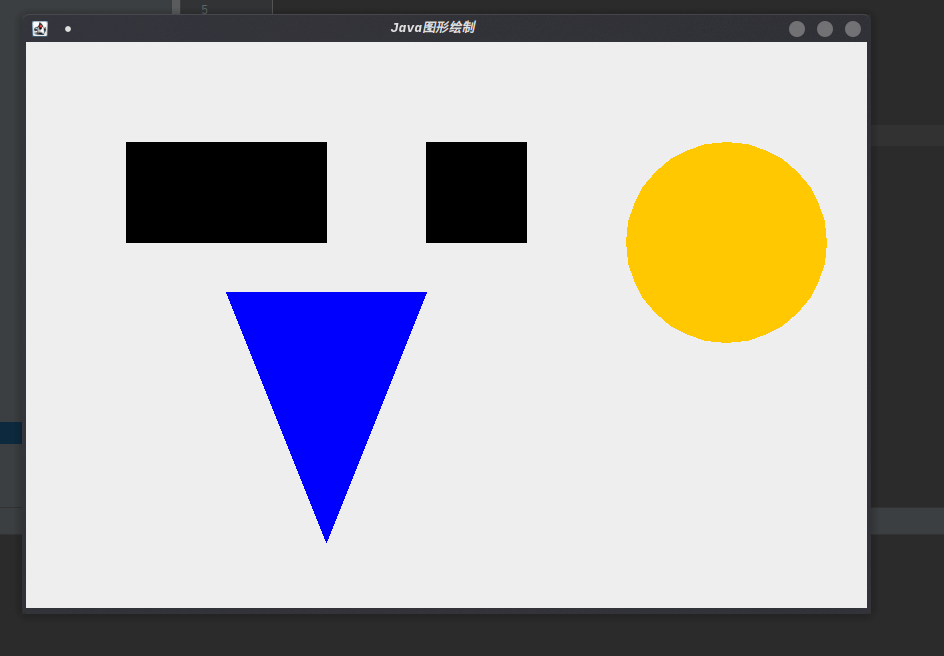
1

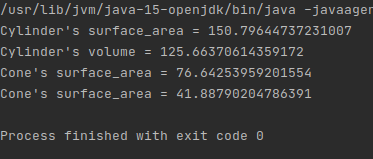
import javax.swing.\*;  
import java.awt.\*;  
  
*/\*\**  
 *\** ***@author*** *liukanglai*  
 *\** ***@date*** *4/7/21 - 4:27 PM*  
 *\**  
 *\*/*  
  
// 代码参考自https://blog.csdn.net/Phage625/article/details/88990859  
// 其又参考自https://blog.csdn.net/xietansheng/article/details/55669157  
// 修改了Triangle的area求法  
  
public class T1Shape {  
}  
  
abstract class Shape {  
 public int x, y;  
 public Color c;  
 public Graphics g;  
  
 public Shape() {  
 }  
  
 public Shape(int x, int y, Color color) {  
 this.x = x;  
 this.y = y;  
 c = color;  
 }  
 public abstract void draw(Graphics g);  
 public abstract void area();  
  
 public static void main(String[] args) {  
 MyFrame frame = new MyFrame();  
 frame.setVisible(true);  
 }  
  
 public static class MyFrame extends JFrame {  
  
 public static final String *TITLE* = "Java图形绘制";  
 public static final int *WIDTH* = 800;  
 public static final int *HEIGHT* = 600;  
  
 public MyFrame() {  
 super();  
 initFrame();  
 }  
  
 private void initFrame() {  
 // 设置 窗口标题 和 窗口大小  
 setTitle(*TITLE*);  
 setSize(*WIDTH*, *HEIGHT*);  
 // 设置窗口关闭按钮的默认操作(点击关闭时退出进程)  
 setDefaultCloseOperation(WindowConstants.*EXIT\_ON\_CLOSE*);  
 // 把窗口位置设置到屏幕的中心  
 setLocationRelativeTo(null);  
 // 设置窗口的内容面板  
 MyPanel panel = new MyPanel(this);  
 setContentPane(panel);  
 }  
 }  
  
 public static class MyPanel extends JPanel {  
  
 private MyFrame frame;  
  
 public MyPanel(MyFrame frame) {  
 super();  
 this.frame = frame;  
 }  
  
 public void paintComponent(Graphics g) {  
 super.paintComponent(g);  
 Square a = new Square(400,100,100,100);  
 a.draw(g);  
 a.area();  
 Rectangle b = new Rectangle(100,100,200,100);  
 b.draw(g);  
 b.area();  
 Circle c = new Circle(600,100,200);  
 c.draw(g);  
 c.area();  
 Triangle d = new Triangle(200,250,400,250,300,500);  
 d.draw(g);  
 d.area();  
 }  
 }  
}  
  
class Square extends Shape{  
 public int x,y;  
 public int length,height;  
 public Square(int x, int y, int l, int h){  
 this.x=x;  
 this.y=y;  
 length=l;  
 height=h;  
 }  
 public void area(){ // 均需int  
 double area = length\*height;  
 System.*out*.println("Square's area is: "+area);  
 }  
 public void draw(Graphics g){ ;  
 g.setColor(Color.*black*);  
 g.drawRect(x,y,length,height);  
 g.fillRect(x,y,length,height);  
 }  
}  
  
class Triangle extends Shape{ // Point 只能用int，，，  
 private Point a=new Point();  
 private Point b=new Point();  
 private Point c=new Point();  
 public Triangle(int x1, int y1, int x2, int y2, int x3, int y3){  
 a.x=x1;  
 a.y=y1;  
 b.x=x2;  
 b.y=y2;  
 c.x=x3;  
 c.y=y3;  
 }  
 public void area(){  
 double area = (a.x\*b.y - a.x\*c.y + b.x\*c.y - b.x\*a.y + c.x\*a.y - b.x\*b.y);  
 System.*out*.println("Triangle's area is: "+area);  
 }  
 public void draw(Graphics g){  
 Polygon p =new Polygon();  
 p.addPoint(a.x,a.y);  
 p.addPoint(b.x,b.y);  
 p.addPoint(c.x,c.y);  
 g.setColor(Color.*blue*);  
 g.drawPolygon(p);  
 g.fillPolygon(p);  
 }  
}  
  
class Rectangle extends Square{  
 public int length,height;  
 public Rectangle(int x, int y, int l, int h){  
 super(x,y,l,h);  
 this.length=l;  
 this.height=h;  
 }  
 public void area(){  
 double area = length\*height;  
 System.*out*.println("Rectangle's area is: "+area);  
 }  
 public void draw(Graphics g){  
 super.draw(g);  
 }  
}  
  
class Circle extends Shape{  
 public int radius;  
 public int x, y;  
 public Circle(int x,int y,int r)  
 {  
 this.x=x;  
 this.y=y;  
 radius=r;  
 }  
 public void area()  
 {  
 double area = Math.*PI*\*radius\*radius;  
 System.*out*.println("Circle's area is: "+area);  
 }  
 public void draw(Graphics g)  
 {  
 g.setColor(Color.*orange*);  
 g.drawArc(x,y,radius,radius,0,360);  
 g.fillOval(x,y,radius,radius);  
 }  
}

k

发现一个有意思的现象，我拖动图像界面大小的时候，输出会一直进行。。。

2

class T2Start {  
 public static void main(String[] args) {  
 Cylinder a\_cylinder = new Cylinder(10,2);  
 Cone a\_cone = new Cone(10,2);  
 a\_cylinder.calculateSurfaceArea();  
 a\_cylinder.calculateVolume();  
 a\_cone.calculateSurfaceArea();  
 a\_cone.calculateVolume();  
 }  
}  
  
public interface ThreeDimensionalGraphics {  
 void calculateSurfaceArea();  
 void calculateVolume();  
}  
  
  
class Cylinder implements ThreeDimensionalGraphics {  
 double hight, radius;  
 Cylinder(double hight, double radius){  
 this.hight = hight;  
 this.radius = radius;  
 }  
 public void calculateSurfaceArea() {  
 double surface\_area = hight \* Math.*PI* \* 2 \* radius + 2 \* radius \* radius \* Math.*PI*;  
 System.*out*.println("Cylinder's surface\_area = " + surface\_area);  
 }  
  
 public void calculateVolume() {  
 double volume = hight \* Math.*PI* \* radius \* radius;  
 System.*out*.println("Cylinder's volume = " + volume);  
 }  
  
}  
class Cone implements ThreeDimensionalGraphics {  
 double hight, radius;  
  
 Cone(double hight, double radius){  
 this.hight = hight;  
 this.radius = radius;  
 }  
 public void calculateSurfaceArea() {  
 double surface\_area = Math.*PI* \* radius \* Math.*pow*((hight \* hight + radius \* radius), 0.5) + Math.*PI* \* radius \*radius;  
 System.*out*.println("Cone's surface\_area = " + surface\_area);  
 }  
  
 public void calculateVolume() {  
 double volume = hight \* Math.*PI* \* radius \* radius / 3.0;  
 System.*out*.println("Cone's surface\_area = " + volume);  
 }  
}



3

class PayStart {  
 public static void main(String[] args) {  
 Bus a\_bus = new Bus();  
 Taxi a\_taxi = new Taxi();  
 a\_bus.收取费用();  
 a\_taxi.收取费用();  
 a\_taxi.AdjustTemp();  
 }  
}  
  
interface 收费 {  
 void 收取费用();  
}  
  
interface 调节温度 {  
 void AdjustTemp();  
}  
  
class Bus implements 收费 {  
 public void 收取费用() {  
 System.*out*.println("公交车收取费用，？？？");  
 }  
}  
  
class Taxi implements 收费 , 调节温度 {  
 public void 收取费用() {  
 System.*out*.println("出租车收取费用，？？？");  
 }  
 public void AdjustTemp() {  
 System.*out*.println("出租车调节温度，？？？");  
 }  
}

