```java
//TIP To <b>Run</b> code, press <shortcut actionId="Run"/> or
// click the <icon src="AllIcons.Actions.Execute"/> icon in the gutter.
public class Main {
    public static void main(String[] args) {
        //TIP Press <shortcut actionId="ShowIntentionActions"/> with your caret at the highlighted text
        // to see how IntelliJ IDEA suggests fixing it.
        System.out.printf("Hello and welcome!");

        for (int i = 1; i <= 5; i++) {
            //TIP Press <shortcut actionId="Debug"/> to start debugging your code. We have set one <icon src="AllIcons.Debugger.Db_set_breakpoint"/> breakpoint
            // for you, but you can always add more by pressing <shortcut actionId="ToggleLineBreakpoint"/>.
            System.out.println("i = " + i);
        }
    }
}
```

```java
public abstract class Shape {
    protected String color;
    protected double height;
    protected double width;
    protected double base;

    public void setColor(String color){
        this.color=color;
    }
    public void setHeight(double height){
        this.height=height;
    }
    public void setWidth(double width){
        this.width=width;
    }
    public void setBase(double base){
        this.base=base;
    }
    public abstract double getArea();
    public String toString(){
        return "Shape[color=" +color+"]";
    }
    public void displayShapeName(){
        System.out.println("I am a Shape.");
    }
}

```

```java
public class Circle extends Shape {
    protected double radius;
    private final double PI = Math.PI;

    public Circle(double radius){
        this.radius=radius;
    }

    public Circle(double radius, double height){
        this.radius=radius;
        super.height=height;
    }
    public double getArea(){
        double area = PI*Math.pow(this.radius,2);
        return area;
    }

    @Override
    public void displayShapeName(){
        System.out.println("Drawing a Circle of radius " + this.radius);
    }

    @Override
    public String toString(){
        return "Circle[ radius=" +radius + super.toString()+ "]";
    }
}

```

```
 1 public class myRunner {
 2     public static void main(String[] args){
 3         Circle c = new Circle(100);
 4         System.out.println("Area of Circle " + c.
   getArea());
 5
 6         System.out.println("++++++++++");
 7         Shape shapeCircleObj = new Circle(100);
 8         shapeCircleObj.displayShapeName();
 9         System.out.println("Area of Circle " +
   shapeCircleObj.getArea());
10         System.out.println(shapeCircleObj);
11         System.out.println(shapeCircleObj
   instanceof Circle);
12         System.out.println("++++++");
13
14         Shape shapeRectangleObj = new Rectangle("
   Red");
15         shapeRectangleObj.displayShapeName();
16         shapeRectangleObj.setHeight(2);
17         shapeRectangleObj.setWidth(4);
18         System.out.println("Area of Rectangle is "
    + shapeRectangleObj.getArea());
19         System.out.println(shapeCircleObj
   instanceof Rectangle);
20
21         System.out.println("--------------");
22         Shape shapeTriangleObj = new Triangle("Blue
   ");
23         shapeTriangleObj.displayShapeName();
24         shapeTriangleObj.setHeight(10);
25         shapeTriangleObj.setBase(15);
26         System.out.println("Area of Triangle is " +
   shapeTriangleObj.getArea());
27         System.out.println(shapeTriangleObj);
28     }
29
30 }
31
```

```java
public class Triangle extends Shape{
    public Triangle(){}
    public Triangle(String color){
        super.color =color;
    }
    public void setBase(int base){
        this.base=base;
    }
    @Override
    public double getArea(){
        return .5*super.base *super.height;
    }
    @Override
    public void displayShapeName(){
        System.out.println("I am a triangle");
    }

    @Override
    public String toString() {
        return "Triangle[base="+super.base+",height
  ="+super.height +"," +super.toString() + "]";
    }
}

```

```java
public class Rectangle extends Shape{
    public Rectangle(String color){
        super.color=color;

    }
    public Rectangle(){

    }
    public Rectangle(String color, double width, double height){
        super.height = height;
        super.width =width;
        super.color =color;
    }
    @Override
    public double getArea(){
        return super.width *super.height;
    }
    @Override
    public void displayShapeName(){
        System.out.println("I am a Rectangle");
    }

    @Override
    public String toString(){
        return "Rectangle[height=" +height+",width =" +width+"," + super.toString()+"]";
    }
}

```