Day 6 - Core Java / GENERICS

Topics:

Generics basics and implementation

Assignments:

Assignment #1

Create classes circle and Cylinder as shown in the UML diagram below:

```
Circle
-radius:double = 1.0
-color:String = "red"
+Circle()
+Circle(radius:double)
+Circle(radius:double,color:String)
+getRadius():double
+setRadius(radius:double):void
+getColor():String
+setColor(color:String):void
+getArea():double
+toString():String.
                                           "Circle[radius=r,color=c]"
                    \ superclass
         extends <sup>∠</sup>
                     subclass
                Cylinder
-height:double = 1.0
+Cylinder()
+Cylinder(radius:double)
+Cylinder(radius:double,height:double)
+Cylinder(radius:double,height:double,
   color:String)
+getHeight():double
+setHeight(height:double):void
+getVolume():double
```

In the main() function of a Program class, create an array of circle references with the initialization shown below:

```
Circle[] circles = {
    new Cylinder(12.34),
    new Cylinder(12.34, 10.0),
    new Cylinder(12.34, 10.0, "blue")
};
```

Print the area of the circular region of each cylinder along with the volume of the same.

Assignment #2

Create the classes Shape, Circle, Rectangle, and Square as shown in the *UML* diagram below:

```
Shape
     -color:String = "red"
     -filled:boolean = true
     +Shape()
     +Shape(color:String,filled:boolean)
     +getColor():String
     +setColor(color:String):void
     +isFilled():boolean
     +setFilled(filled:boolean):void
     +toString():String ◆ -
                                             "Shape[color=?,filled=?]"
                   extends
            Circle
                                                  Rectangle
-radius:double = 1.0
                                       -width:double = 1.0
                                       -length:double = 1.0
+Circle()
+Circle(radius:double)
                                       +Rectangle()
+Circle(radius:double,
                                       +Rectangle(width:double,
   color:String,filled:boolean)
                                          length:double)
+getRadius():double
                                       +Rectangle(width:double,
+setRadius(radius:double):void
                                          length:double, color:String,
+getArea():double
                                         filled:boolean)
+getPerimeter():double
                                       +getWidth():double
+toString():String。
                                       +setWidth(width:double):void
                                       +getLength():double
                                       +setLength(legnth:double):void
           "Circle[Shape[color=?,
                                       +getArea():double
           filled=?],radius=?]"
                                       +getPerimeter():double
                                      +toString():String
     "Rectangle[Shape[color=?,
                                                    Square
     filled=?],width=?,length=?]"
                                       +Square()
                                       +Square(side:double)
                                       +Square(side:double,
     The length and width shall be
                                          color:String,filled:boolean)
                                       +getSide():double
     set to the same value.
                                       +setSide(side:double):void
                                      +setWidth(side:double):void
 "Square[Rectangle[Shape[color=?,
                                      +setLength(side:double):void
filled=?],width=?,length=?]]"
                                       +toString():String
```

The tostring function of the above classes should return text as given below:

Classname	Sample return value from toString()
Shape	A Shape with color of xxx and filled/Not filled
Circle	A Circle with radius=xxx, which is a subclass of yyy (where yyy is the output of the toString() method from the superclass)
Rectangle	A Rectangle with width=xxx and length=zzz, which is a subclass of yyy (where yyy is the output of the toString() method from the superclass)
Square	A Square with side=xxx, which is a subclass of yyy (where yyy is the output of the toString() method from the superclass)

In the main() method of a Program class, create an array of 10 Shape references containing a mixture of Circle, Rectangle and Square objects of different dimensions. Using a loop, print the perimeter and area for all of them.