Exercise 12 Answers

Run-time type information

Here is the `ShapeTest.java` source file after all modifications have been made:

```
abstract class Shape {
}

class Circle extends Shape {

   // Properties of the class...
   public double radius;

   // Constructor of the class...
   public Circle(double aRadius) {
      radius = aRadius;
   }
}

class Triangle extends Shape {

   // Properties of the class...
   public double base;
   public double height;

   // Constructor of the class...
   public Triangle(double aBase, double aHeight) {
      base   = aBase;
      height = aHeight;
   }

}

class Rectangle extends Shape {

   // Properties of the class...
   public double width;
   public double length;

   // Constructor of the class...
   public Rectangle(double aWidth, double aLength) {
      width  = aWidth;
      length = aLength;
   }

}
```

```
class ShapeTest {

   public Shape[] myShapes;

   public void printAreas() {

      for (int i=0; i<myShapes.length; i++) {

         System.out.print("Shape " + i + " has area: ");

         if (myShapes[i] instanceof Circle) {
            Circle c = (Circle)myShapes[i];
            System.out.println(Math.PI * c.radius * c.radius);
         }
         if (myShapes[i] instanceof Triangle) {
            Triangle t = (Triangle)myShapes[i];
            System.out.println(0.5 * t.base * t.height);
         }
         if (myShapes[i] instanceof Rectangle) {
            Rectangle r = (Rectangle)myShapes[i];
            System.out.println(r.width * r.length);
         }
      }
   }

   public void printNames() {

      for (int i=0; i<myShapes.length; i++) {

         System.out.print("Shape " + i + " is a: ");

         if (myShapes[i] instanceof Circle) {
            System.out.println("circle");
         }
         if (myShapes[i] instanceof Triangle) {
            System.out.println("triangle");
         }
         if (myShapes[i] instanceof Rectangle) {
            System.out.println("rectangle");
         }
      }
   }

   public void doStuff() {

      // create an empty shapes array...
      myShapes = new Shape[4];

      // fill in the values of the elements...
      myShapes[0] = new Circle(12.0);
      myShapes[1] = new Circle(6.3);
      myShapes[2] = new Triangle(3,8);
      myShapes[3] = new Rectangle(10,10);
```

```
      printNames();
      printAreas();
   }


   // The main method is the point of entry into the program...
   public static void main(String[] args) {

      ShapeTest me = new ShapeTest();
      me.doStuff();

   }
}
```