The Hong Kong Polytechnic University
Department of Electronic and Information Engineering

EIE3320 Lab 1: Object-Oriented Programming using Java

(Deadline for Submission: Check the course information)

Important Note: This is a group project. TWO persons form a group. You may use Blackboard Discussion to find partner. If you prefer to do it on your own, one person group is also fine but please specify you are one-person group at the cover page.

Expected Outcomes

- Understand the principles of Object-Oriented design.
- Apply Java in Object-Oriented software development.
- Apply UML in Object-Oriented software modelling.
- Apply Object-Oriented approach to developing computer software.
- Learn independently and be able to search for the information required in solving problems.
- Present ideas and finding effectively.
- Work in a team and collaborate effectively with others.

Assessment Criteria

Your submission should contain (but not limited to) the following:

1. Complete **source code** (inside project folder) of your solutions (include the parts that you have modified). Use 7-z or zip to compress all your whole project folders into a single file. Type your student name and student ID at the beginning of each source code (.java file) to identify yourself.

2. The format of **report** should include the followings:
   2.1 Cover Page: Course name/code, student IDs/names, assignment name and date.
   2.2 Introduction: A detailed description of the objectives and requirements of the program, and a brief description of the methodology.
   2.3 Methodology: The methodology when implementing the program. It contains
   - How your team divides the work among the team members?
   - The schedule of implementing the program
   - The program structure of the program developed, including
     - The specifications of the classes defined, and the public/private member functions/variables included
     - The flow of execution such as class diagram or flow chart.

   2.4 Program Testing
   - The validations of your program and confirmed that it is running correctly.
     - Include the execution results of your program captured from the screen.

   2.5 Conclusion
   - Summarize the experience gained in the program

   2.6 Future Development
   - Indicate how your program can be extended.

3. Use screen recorder software such as “Screen-o-matic”, “Screeny: Screenshot & Screen Video Recorder” from Chrome Web Store, mobile phone to do a **video** demonstration. Remember to capture your student ID & name from your program code at the beginning of your video clip to identify yourself. Please show how you can run your program. Demonstrate your program with different choices or values to proof that you meet all the requirements. The video size should be less than 120 MB and the video length should not last longer than 15 minutes. It is optional for you to add any narration inside the video. It is not required to do any explanation on program code inside the video.

General Descriptions

- After finished the program, all students should upload their program(s), report and video for demonstration to Blackboard.
- Please state clear title, date and ALL members’ name and student ID at the cover page. If you fail to indicate your group detail at the cover page, both of you will score no mark.
- Each team member should declare his/her responsibility in the report if you are not one-person group. Each member will be individually assessed based on the declared responsibility and the result obtained.
- The report should be in Word or PDF format. It is NOT required to include the complete source code in the report.
- It is compulsory to use a word processing tool to write your report. The font size must not be bigger than 12 or smaller than 10. Use 1.5 lines spacing on both sides of a page. The length of the report should not be shorter than 5 pages.

A. Brief Overview of Abstract Classes and Interface

Study the following URL

http://java.sun.com/docs/books/tutorial/java/IandI/abstract.html
http://java.sun.com/docs/books/tutorial/java/IandI/createinterface.html

B. Problem Statement

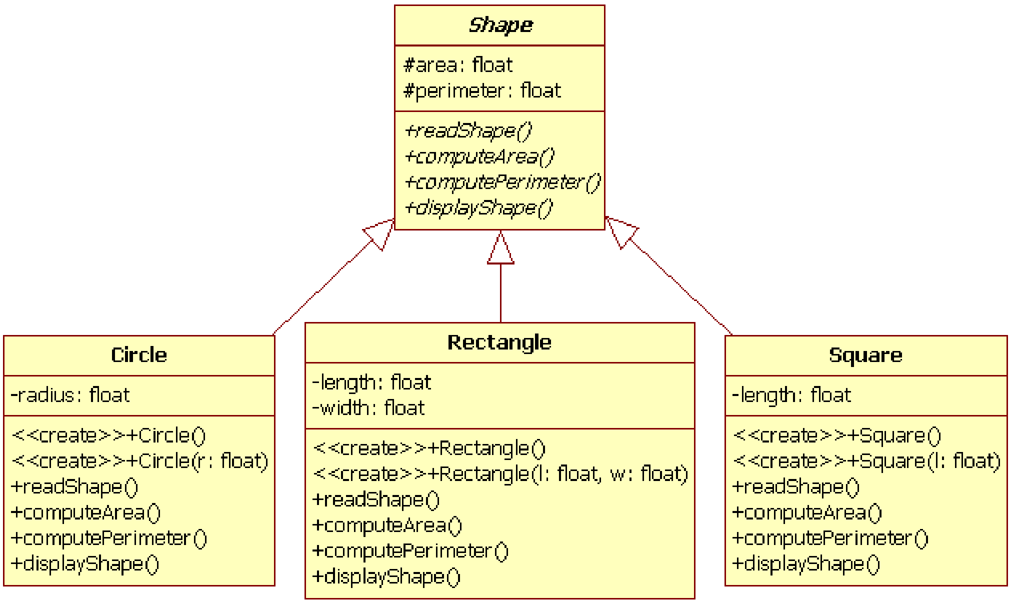
Write a program that can compute and display the area and perimeter of circles, squares, and rectangles. To achieve this task, you may create an abstract class called **`Shape`** that contains the following abstract methods.

```
// To read the shape information from users
abstract public void readShape();

// To compute the shape’s area
abstract public void computeArea();

// To computer the shape’s perimeter
abstract public void computePerimeter();

// To display the area and perimeter of the shape
abstract public void displayShape();
```

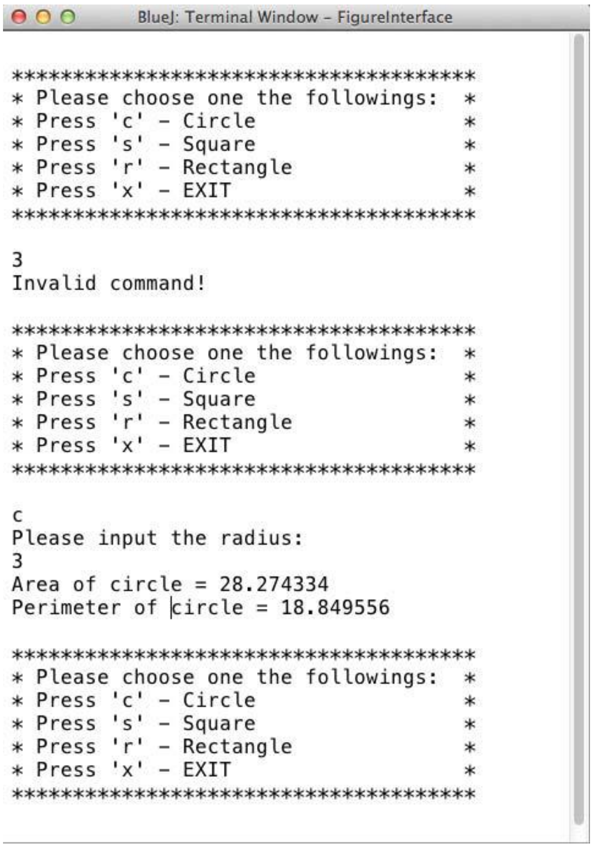
The **`Shape`** class also contains the protected member **`area`** and **`perimeter.`** You should use inheritance and polymorphism to make the code reusable and to reduce code complexity. Refer to the lecture notes on “Polymorphism” for the details of protected members.

C. Procedure

1. Use BlueJ or Eclipse or other tool using JDK to create the classes: **`Shape`**, **`Rectangle`**, **`Circle`**, and **`Square`**. These classes should have the relationship shown as follows. ***Hint*:** Some attributes may be unique to some classes only.



2. Create a Java file called "ShapeTester.java" to test the implementation. Your program output should look something like the following.



```
**************************************
* Please choose one the followings:  *
* Press 'c' - Circle                 *
* Press 's' - Square                 *
* Press 'r' - Rectangle              *
* Press 'x' - EXIT                   *
**************************************

3
Invalid command!

**************************************
* Please choose one the followings:  *
* Press 'c' - Circle                 *
* Press 's' - Square                 *
* Press 'r' - Rectangle              *
* Press 'x' - EXIT                   *
**************************************

c
Please input the radius:
3
Area of circle = 28.274334
Perimeter of circle = 18.849556

**************************************
* Please choose one the followings:  *
* Press 'c' - Circle                 *
* Press 's' - Square                 *
* Press 'r' - Rectangle              *
* Press 'x' - EXIT                   *
**************************************
```

3. Create a Java file called “Picture.java”. The class `Picture` should contain an `ArrayList` called `shapes` that stores a collection of `Circle`, `Square`, and `Rectangle` objects.

a) Write a public method with signature and return type as follows:

```
void addShape(Shape s);
```

The method shall add a `Shape` object `s` to the ArrayList `shapes`;

b) Write a public method with signature and return type as follows:

```
void computeShape();
```

The method shall compute the areas and perimeters of all objects in the ArrayList `shapes`;

c) Write a public method with signature and return type as follows:

```
public void listAllShapeTypes();
```

The method shall invoke `displayShape()` to display the areas and perimeters of all objects in the ArrayList `shapes`.

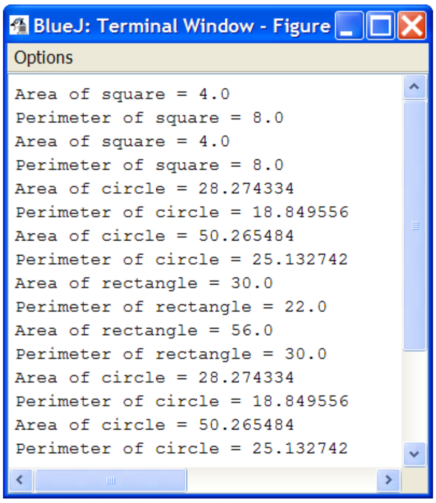
d) Write a public method with signature and return type as follows:

```
public void listSingleShapeType(String className);
```

The method shall display the areas and perimeters of all objects belonging to `className`. Note that `className` can be either `Circle`, `Square`, or `Rectangle`.

e) Create a class called `PictureTester` to test your codes in (a)–(d). Your “PictureTester.java” and console window should look like the following:

```
// PictureTester.java
public class PictureTester
{
  public static void main(String[] args) {
    Picture p = new Picture();
    p.addShape(new Square(2));
    p.addShape(new Square(2));
    p.addShape(new Circle(3));
    p.addShape(new Circle(4));
    p.addShape(new Rectangle(5,6));
    p.addShape(new Rectangle(7,8));
    p.computeShape();
    p.listAllShapeTypes();
    p.listSingleShapeType("Circle");
  }
}
```



```
Area of square = 4.0
Perimeter of square = 8.0
Area of square = 4.0
Perimeter of square = 8.0
Area of circle = 28.274334
Perimeter of circle = 18.849556
Area of circle = 50.265484
Perimeter of circle = 25.132742
Area of rectangle = 30.0
Perimeter of rectangle = 22.0
Area of rectangle = 56.0
Perimeter of rectangle = 30.0
Area of circle = 28.274334
Perimeter of circle = 18.849556
Area of circle = 50.265484
Perimeter of circle = 25.132742
```

Hints:

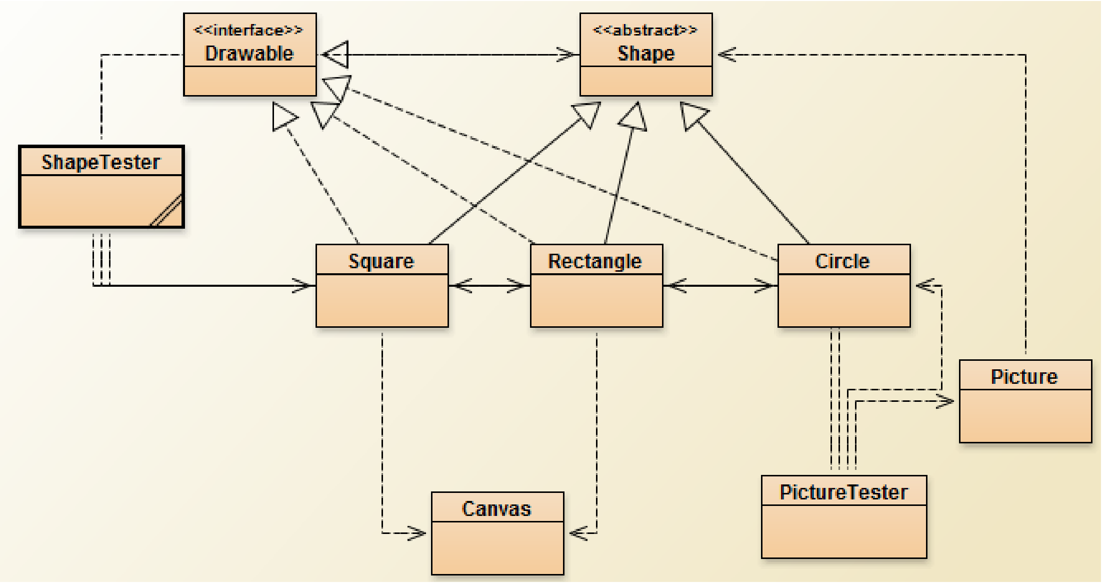
1. Use the class `Class` detailed in http://docs.oracle.com/javaee/6/api/ .
2. Search the Internet using keywords: Java classname
3. Search the Internet using keywords: Java string comparison

4. Extend the program to draw the figure on the screen. You may use the "implements" keyword to produce the multiple-inheritance relationship as shown below. Note also that you may need to use the `Canvas` class included in the **BlueJProjects.zip** posted in our subject website in https://learn.polyu.edu.hk (under Projects/Chapter1/Shapes).
Note**:** To avoid name crashes between the AWT classes and the classes that you defined in this lab exercises, you may need to prefix the classes `Rectangle` and `Shape` with `java.awt.` in
"Canvas.java", i.e., replace `Rectangle` with `java.awt.Rectangle`.

The class Drawable has the code looks like the following:

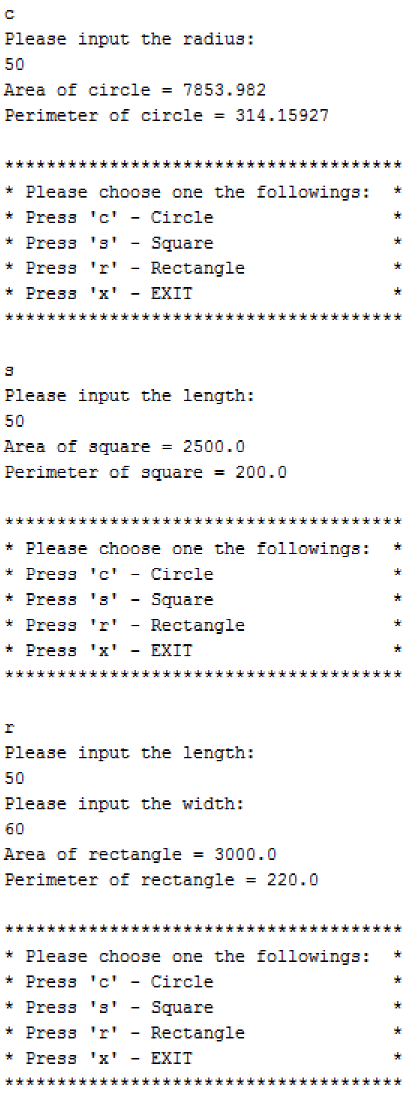
```
public interface Drawable
{
   void draw();
}
```

Note that you also need to change the code in "Shape.java" so that it implements the `Drawable` interface. The following shows the class diagram of program:

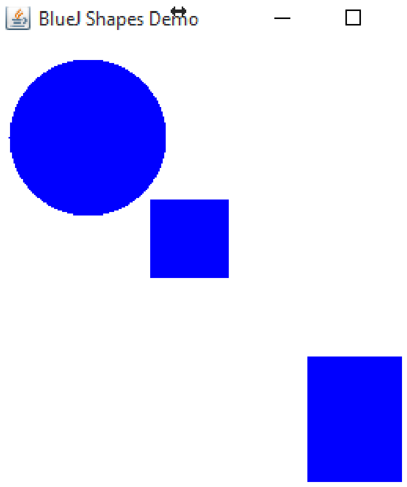


(If you use BlueJ to write your program, you will get the same class diagram.)

The result should draw the figure on the screen when input value in “ShapeTester.java”, the program output looks like the following. It is acceptable that your output figure overlaps with other figures.



```
c
Please input the radius:
50
Area of circle = 7853.982
Perimeter of circle = 314.15927

**************************************
* Please choose one the followings:  *
* Press 'c' - Circle                 *
* Press 's' - Square                 *
* Press 'r' - Rectangle              *
* Press 'x' - EXIT                   *
**************************************

s
Please input the length:
50
Area of square = 2500.0
Perimeter of square = 200.0

**************************************
* Please choose one the followings:  *
* Press 'c' - Circle                 *
* Press 's' - Square                 *
* Press 'r' - Rectangle              *
* Press 'x' - EXIT                   *
**************************************

r
Please input the length:
50
Please input the width:
60
Area of rectangle = 3000.0
Perimeter of rectangle = 220.0

**************************************
* Please choose one the followings:  *
* Press 'c' - Circle                 *
* Press 's' - Square                 *
* Press 'r' - Rectangle              *
* Press 'x' - EXIT                   *
**************************************
```


BlueJ Shapes Demo