

CPSC 1181 - Assignment 4 [50 marks]

Submission:

- Submit a **zip file** containing **only** the .java files to Brightspace prior to the **due date set in Brightspace**. Do not include class or other files
- Submissions that are less than 24 hours late receive a 1% per hour late penalty. Submission that are more than 24 hours late will not be accepted.
- You can use one of your 2 day extensions.
 - If you submit more than 24 hours late, one will be used automatically
 - **Add a comment to your submission in Brightspace IF**
 - You want to use an extension for a submission less than 24 hours late
 - You are submitting more than 24 hours late, but **don't** want to use an extension...you just want feedback.
- Submissions that are unzipped or that contain .class or other unneeded files will be penalized.
- **IMPORTANT: If you are unsure of your submission for any reason, submit it AND email it to me.**

Working With a Partner

- You must tell me **in-person** with your partner that you are working as partners to have permission. This must be done before the assignment is posted.
- If one partner no longer has an extension, neither partner can use an extension.
- Only one group submission is required: submitted by either partner
 - If both submit, we will mark whichever we open first
- **Add a second @author to your JavaDoc**

Exercise 1

Create an abstract class called GeometricShape. A GeometricShape may or may not be filled in. GeometricShape should contain a standard constructor that sets the data member and a no-argument constructor that creates a filled in shape, as well as getters and setters for the instance data and a toString method.

Create two abstract methods: one for getting the area of a shape, and one for getting the perimeter of a shape.

Exercise 2

Modify GeometricShape so that it implements the Comparable interface. Implement the compareTo() method, comparing shapes by their areas. **The Comparable interface is part of the Java library, do not try to create the Comparable interface.**

Exercise 3

Create a child class of GeometricShape, called RectangleShape. RectangleShape has a width and a height. Those values should be positive integers. RectangleShape should have an appropriate constructor for initializing instance data. Provide getters and setters for the data members and a toString method.

In RectangleShape, be sure to implement the abstract methods of the parent class.

Exercise 4

Create an interface called ASCIIDrawable. The interface should have one unimplemented method, String drawAsASCII() which will return a String containing the object drawn using ASCII characters.

CPSC 1181 - Assignment 4 [50 marks]

Exercise 5

Add implements `ASCIIDrawable` to the `GeometricShape` class, but implement the `drawAsASCII ()` function in the `RectangleShape` subclass. See the examples below for the expected output.

The `toString` method should not be used to draw anything, just to display the basic info of an object.

Exercise 6

Create a new class, `Billboard`, that represents a sign containing a message. It contains one `String` data member, `message`, a setter and getter for it, and a `toString`. It should implement the `ASCIIDrawable` interface. Have the message **on a single line with a space separating the message from the surrounding line**. See the example below for the expected output. **A billboard does not extend any of the previous classes. It does not use the previous class to draw the rectangle around the message**

Exercise 7

Create a tester class called `GeometricShapeTester` (not `jUnit`). Create an Array of `GeometricShapes` that contains several `RectangleShapes`.

- Use **`Arrays.sort`** to demonstrate that your array of shapes is sorted ascendingly by area (it functions the same was as `Collections.sort`)

Create an array with the `ASCIIDrawable` interface type and use it to test the draw function of `RectangleShapes` and `Billboards`.

Exercise 8

Create a comparator called `PerimeterComparator` that compares `GeometricShape` objects by perimeter in decreasing order. In `GeometricShapeTester`, use the comparator to sort the array in descending order by perimeter.

Examples (In order, unfilled Rectangle with height /width 3, filled rectangle with height /width 5, unfilled Rectangle with 7 width and 4 height, and a billboard)

```
Problems Javadoc Declaration Console Coverage
<terminated> Square [Java Application] C:\Program Files\Java\jdk-11.0.1\bin\javaw
####
# #
###

#####
#####
#####
#####
#####

#####
#   #
#   #
#####

#####
#           #
# Old Joe's Place #
#           #
#####
```

CPSC 1181 - Assignment 4 [50 marks]

Marking Rubric:

Style, Convention [5 marks]

Documentation [5 marks]

GeometricShape.java[10 marks]

PerimeterComparator.java[5 marks]

RectangleShape.java [9 marks]

ASCIIDrawable.java[2 marks]

Billboard.java[7 marks]

GeometricShapeTester.java[7 marks]