

Assignment 5

Point Class	40 pts.
Date Class Improvements	20 pts.
Right Triangle Class	40 pts.

TOTAL: 100 pts.

General Requirements

- *All files you are turning in must be signed. Please add your name to the source code in comments. Unsigned work automatically loses 5 pts.*
- *Add comments to the source code you are writing:*
 - *Describe the purpose of every variable*
 - *Explain the algorithm you are using for solution*
 - *Add proper comments for all methods. Include @param, @return, and @throws tags*
- *Uncommented work loses up to 25% of points for the problem.*
- *Archive the complete projects using ZIP or RAR utility. Turn in the work into the digital drop box.*

Point Class

Write a definition of a class named `Point` that might be used to store and manipulate the location of a point on the plane. The point is stored as two coordinates: `x` and `y`. You will need to declare and implement the following methods:

1. Two constructors:
 - a. no-argument constructor that sets the point coordinates to (0,0), and
 - b. a constructor that takes `x` and `y` coordinate of the point and sets member variables.
2. Method `set` that sets the private data after an object of this class is created.
3. A method to move the point by an amount along the vertical and horizontal directions specified by the first and second arguments:
`move(double dx, double dy)`
4. The method to rotate the point by 90 degrees clockwise around the origin.
Hint: when point is getting rotated 90 clockwise around the origin the following changes happen to its coordinates:
 $X_{rotated} = y;$
 $Y_{rotated} = -X .$
5. two accessor methods to retrieve the coordinates of the point

Call your file `Point.java`.

`PointDemo.java` file contains test cases for the `Point` class. It is provided with the

assignment. make sure it works correctly with the class you wrote.

Date Class Improvements

Start your work with the class Date provided in **Date.java** file.

Make the following improvements to the class:

1. Replace all error-handling code that ends with `System.exit(0)` with a statement that throws `Illegal Argument Exception`. Please see detailed description in the commented sections in the code.
2. Add 3 methods (with different names) to print the date in 3 different formats:
12/25/16
December 25, 2016
25 December 2016

Call the source code files `Date.java`.

Write a class with `main()` that uses your class.

- Please do not collect input from the user. Instead call ALL the methods of the class with the data you choose to make sure the methods work properly. In the comments explain why you decided to use these particular test cases.
- Have try/catch blocks that handle `Illegal Argument Exception`. Make sure you write code that invokes methods throwing `Illegal Argument Exception` in `main()`. Please leave it in your code for me to see. You can comment it out if it interrupts the flow of your program.

Call the source code files `DateDemo.java`.

Right Triangle Class

Write a `RightTriangle` class that has the following fields:

- `legA`: a double
- `legB`: a double

The class should have the following methods:

- **No-argument constructor**. Sets fields to 0
- **Constructor**. Accepts legs of the right triangle as arguments. Constructor throws `Illegal Argument Exception` when one or both legs are set to 0 or negative number(s)
- **`setLegA()` , `setLegB()`** - mutator methods. Both methods throw `Illegal Argument Exception` when leg is set to 0 or negative number.
- **`getLegA()` , `getLegB()`** - accessor methods.
- **`getHypotenuse()`** – calculates and returns hypotenuse of the triangle.

- **GetArea()** – calculates and returns area of the right triangle
- **getPerimeter()** - calculates and returns perimeter of the triangle

Call the source code files `RightTriangle.java`.

Write a class with `main()` that uses your class.

- Please do not collect input from the user. Instead, call ALL the methods of the class with the data you choose to make sure the methods work properly. In the comments explain why you decided to use these particular test cases.
- Have try/catch blocks that handle `Illegal Argument Exception`. Make sure you write the call that invokes `Illegal Argument Exception` in `main()`. In addition, make sure to call class constructors and mutator methods with bad arguments to showcase exception throwing/catching in action.

Call the source code files `TriangleDemo.java`.