L3b

<div style="border: 1px solid red; padding: 10px; display: inline-block;">Re-submit Assignment</div>

Due Sep 19, 2018 by 11:59pm   **Points** 10   **Submitting** a file upload   **File Types** zip
Available after Sep 10, 2018 at 8am

Please remember that Java API specifications can be found at
http://docs.oracle.com/javase/8/docs/api/   **(http://docs.oracle.com/javase/8/docs/api/)**

1. Get **Section_3b_FillInTheCode.java.** 📄Code the answers in java in the provided file.
   - See **ScannerConsoleInput.java** 📄 for an example for getting user input with the Scanner.
2. Download **PracticeMethods1.java** 📄 and **PracticeMethods2.java.** 📄
   a. complete `PracticeMethods1.java`
   b. after completing the program provide answers to the Discussion Questions 1 and 2 under Programming Activity 2 in Chapter 3 (Section 3.16)
3. In a new file write a program that prompts the users for **four integer** values: `x1, y1, x2, y2` that represent (x,y) coordinates for two points on the plane `p1` and `p2` respectively. Using the `Point` class from **Java Class Library**   **(http://docs.oracle.com/javase/8/docs/api/java/awt/Point.html)** (you need to look up the documentation) create two `Point` objects `p1` and `p2` with your input data, then print the data for both `Point` objects utilizing its `toString` method.
4. In a new file write a program that reads from the keyboard tha radius of a circle. Calculate and output the area and circumference of that circle. Use the following formulas: **area = π * r^2** and **circumference = 2 * π * r**
   - The radius should be a `double`.
   - Use `DecimalFormat` class to format the result, with two digits printed to the right of the decimal point (see example 3.7).
   - Use `Math.PI` for the value of pi
5. In a new file write a program that using a `Scanner` reads a file name input by the user. You should expect that the file name has one . (dot) character in it, separating the file name from the file's extension. Extract the file extension (without the dot) and print it in lowercase letters only, and then print it again in uppercase letters only, and finally output it again the way it was originally entered. Your code must not assume the length of the input. For example if the user inputs `MyFile.txt`, the program should extract `txt`, if the user inputs `index.html`, the program should extract `html`.
6. Complete `PracticeMethods2.java`

Please remember to zip all .java files together and submit for grading:

- your project must **compile**; if the project does not compile an F will be awarded
- each file must contain your **name;** otherwise 1 point will be deducted from your score

- the submitted source code must be **formatted;** otherwise 1 point will be deducted from your score