CS 5000 – Summer 2025
Assignment #3, 50 Points

Math Methods, Characters, and Strings – Chapter 4

Note: If you re-upload the files, you must re-upload **ALL** files as the system keeps the most recent uploaded submission only. No zip files!

Develop a complete Java program for each of the following problems. Please name the programs as indicated and add proper program headers and output labels as specified. ***Do not use loops, arrays, methods, or other concepts we did not discuss to date***, ***please build on what we have covered to date.***

Make sure you **include a header for each program (see previous assignments).**

Program #1 (25 points): Write a Java program, named ***`StringMethods`***, which reads from the user two strings (say `string_1`, and `string_2`). Do not hardcode the inputs, allow the user to enter any name. The program uses String class methods to manipulate these input strings as follows and prints out the outcome as shown below.

a) Determine the length of string_1.
b) Determine the length of string_2.
c) Concatenate both strings, separated by space.
d) Check if the two strings have same set of characters with regard to case (i.e., equal).
e) Convert string_1 to upper case.
f) Convert string_2 to lower case.
g) Extract a valid sub-string of multiple characters from string_1.

Make sure to properly label your output prompts for each manipulation above and print the outputs on separate lines. Use the escape character (\t) to line-up the outputs after the labels as follows (for inputs: `John` and `Amy`):

```
a) Length of String 1:    4 characters
b) Length of String 2:    3 characters
c) Concatenation:         John Amy
d) Equal Strings?         Not equal
e) Uppercase String 1:    JOHN
f) Lowercase String 2:    amy
g) Valid substring:       oh
```

Make sure your code displays the outputs following the test data format. Separate your code into sections with proper in-line comment before each section, such as

```
// Part A: Determine the length of string_1
```

Program #2 (25 points): Write a Java program, named ***`RandomNumbers`***, that generates random numbers as follows.

a) A random integer number between 20 and 80 (inclusive).
b) A random integer number between -20 and 20 (inclusive).
c) A random integer number between -50 and -20 (inclusive).
d) A random floating-point number between 0.0 and 21.9999 (inclusive).

Properly label output for each part above, print the outputs on separate lines, and use the tab escape character (\t) to line-up the outputs after the labels as shown in this sample run:

```
a) A random integer between 20 and 80 (inclusive):       51
b) A random integer between -20 and 20 (inclusive):      -9
```

```
c) A random integer between -50 and -20 (inclusive):     -39
d) A random float between 0.0 and 21.9999 (inclusive):   19.9048
```

Make sure your code displays the outputs following the test data format. Separate your code into sections with proper in-line comments such as

```
// Part A: Generate random integer number between 20 and 80 (inclusive)
```

Submission:

1. Before submitting your programs, make sure you review the assignment submission requirements and grading guidelines posted in D2L. The grading guidelines explain some of the common errors found in programming assignments.
2. The assignment due date is posted in D2L.
3. Please compile, run, and test your code right before you upload your java files to the assignment submission folder in D2L.
4. Please upload only the .java files (total 2 files).