COMP1618 Exercise 6 – File IO

Task 1: Reading and Writing Using Files

1. Use the files *FileStats.java* and *Numbers.txt*. You can compile FileStats.java. It will compile without errors so that you can use it to test out the StatsDemo class you will be creating.

2. Create a class called StatsDemo which consists of a main method to do the following:
   a) Create a DecimalFormat object so that we can format our numbers for output with 3 decimal places (Don't forget the needed import statement).
   b) Create a Scanner object to get the file name input from the user (Don't forget the needed import statement).
   c) Prompt the user and read in the file name (Remember to declare any needed variables).
   d) Create a FileStats object passing it the file name.
   e) Create a PrintWriter object passing it the filename "Results.txt" (Don't forget the needed import statement).
   f) Since you are using a PrintWriter object, add a throws clause to the main method header.
   g) Print the mean and standard deviation to the output file using a three decimal format, labelling each.
   h) Close the output file.Compile and run.

3. Compile, debug, and run. You should get no output to the console but running the program will create a file called Results.txt with your output. The output you should get at this point is: mean = 0.000, standard deviation = 0.000. This is not the correct mean or standard deviation for the data, but we will fix this in the next tasks.

Task 2: The calculateMeanMethod

1. Open *FileStats.java* for editing. You will notice that the calculateMean and calculateStdDev methods do not do any calculations yet. They simply return a 0 to the constructor to initialize the instance variables. We need to add lines to each of these methods to have them return the correct value. Let's work with the calculateMean method first.
2. Create a File object passing it the filename (Don't forget the needed import statement).
3. Create a Scanner object passing it the File object.
4. Since you are using a Scanner object to open a file, add a throws clause to the calculateMean method header as well as the constructor method header (since it calls the calculateMean method).
5. Declare local variables for an accumulator of type double, a counter of type integer, and line of type String. Initialize all number variables to 0.
6. Write a loop that reads values from the file until you are at the end of the file.
7. The body of the loop will
   a. read a double from the file and add the value to the accumulator
   b. increment the counter

8. When the program exits the loop close the input file.
9. Calculate and return the mean instead of 0. The mean is calculated by dividing the accumulator by the counter.
10. Compile, debug, and run. You should now get a mean of 77.444, but the standard deviation will still be 0.000.

Task 3: The calculateStdDevMethod

1. Do steps 2-6 as above in the calculateMean method but add another local variable called difference of type double.
2. The body of the loop will
    a. read a double value from the file, subtract the mean from that value, and store the result in difference
    b. add the square of the difference to the accumulator
    c. increment the counter
3. When the program exits the loop close the input file.
4. The variance is calculated by dividing the accumulator (sum of the squares of the difference) by the counter. Calculate the standard deviation by taking the square root of the variance (Use Math.sqrt () to take the square root).
5. Compile, debug, and run. You should get a mean of 77.444 and standard deviation of 10.021.

Congratulations! You completed the exercise.