# COSC 1436: Programming Fundamentals I

# (Assignment 5)

* **DUE Time and Date**: **Midnight, Friday, 11/30/2018**
* **What to submit:** **LastNameFirstNamePersonalNameList.java**

**LastNameFirstNameSnowFallTotal.java**

**Note 1:** Unless otherwise mentioned, you are asked to upload ONLY your **java source files** (not the class files) through blackboard. Email submission is not accepted, because of confusion in grading.

**Note 2:** If your programs contain any syntactical errors, no points will be given. Thus, please make sure your programs are properly compiled with computers at the CS labs, not only in your laptop or desktop environments.

**Note 3:** No late submission will be accepted, thus keep the deadline.

**Note 4:** Grading will be divided into two categories, formatting and logic, where formatting compromise 25% of your total grade. Formatting will be based on the following rules.

**Rule 1:** Naming is an important issue in Java. Not only you need to define meaningful variable names, but also have to give appropriate names for the physical java file, which should be the same as your public class name that you edit.

Unless otherwise mentioned, you will follow **the industry standard for Java naming convention**:

(1) Java Classes start in uppercase and each individual word in the class name is capitalized;

(2) All Java methods and variables start in lowercase and each individual word in the method and variable is capitalized;

(3) Each final variable (known as a constant) should be written in all uppercase.

**Rule 2:** There should be a space around all operators (e.g., 3 + 5, not 3+5). In addition, spacing with regards to parentheses should be consistent.

**Rule 3:** In addition to the Java naming convention, you are asked to add your name in front of each class name like **LastNameFirstNameClassName.java.**

For instance, if your name is “John Doe” and the class name is “RightTriangle”, then your class name in your source code should be “DoeJohnRightTriangle” and your corresponding physical file name should be “DoeJohnRightTriangle.java”.

**Rule 4:** Everything nested inside of an open brace should be indented with regular-sized spaces (say, 4 or 8 spaces). The open brace for functions and classes should (1) come at the end of the line and be preceded by a space like

**public class DoeJohnRightTriangle {**

**public static void main() {**

**}**

**}**

or (2)start with the new line as shown below:

**public class DoeJohnRightTriangle**

**{**

**public static void main()**

**{**

**}**

**}**

**Rule 5:** Always type block comments to include title of the project, program’s purpose, your name, the date, and the version number as in the lectures or in the labs. For example,

**/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**@Title: LastNameFirstNameClassName**

**@Purpose: To verify the edit, compile, execute function in Textpad**

**@Author: (your last & first name)**

**@Date: (today’s date)**

**@Version: 1.0**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/**

**Or**

**/\*\***

**@Title: LastNameFirstNameClassName**

**@Purpose: To verify the edit, compile, execute function in Textpad**

**@Author: (your last & first name)**

**@Date: (today’s date)**

**@Version: 1.0**

**\*/**

## Question 1 (50 points): PersonalNameList

1. Companies acquire personal information from phone, World Wide Web, or email in order to sell or send an advertisement about their product. However, when this information is acquired, moved, copied or edited, the data may lose its quality. Often, the use of data administrators or a tool that has limited capabilities to correct the mistyped information can cause many problems. Moreover, most of the correction techniques are particularly implemented for the words used in daily conversations.
2. For this assignment question, write a program which will get the personal names from the user and store them in an **array list** called “personalNames”. If there is any other character besides “a-z” or “A-Z” in a record, those should be flagged as ill-defined data. Because of the stated reasons above, instead of trying to fix the error in personal names, we want to remove this ill-defined data from the original array list and move it to another array list called “errorProneNames”. In other words, error involving records should be saved in a separate array. At the end your solution should print out the “personalNames” and “errorProneNames” array list separately.
3. Your file will have the following documentation comments before the class header:
4. **/\*\***
5. @Title: LastNameFirstPersonalNameList
6. @Purpose: To practice ArrayList
7. @Author: (your last first name)
8. @Date: (today’s date)
9. @Version: 1.0
10. \*/

## Question 2 (50 points): Snowfall Total

The table below reflects the snowfall totals in centimeters for 5 different cities between the months of November and March.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | November | December | January | February | March |
| Detroit | 3.6 | 24.6 | 31.8 | 26.4 | 17.5 |
| Chicago | 4.7 | 29.1 | 33.0 | 29.2 | 17.7 |
| Boston | 5.5 | 23.3 | 41.0 | 26.7 | 14.4 |
| New York | 4.4 | 18.8 | 36.1 | 24.4 | 11.1 |
| Washington DC. | 2.1 | 10.1 | 18.8 | 18.8 | 8.9 |

Create a two dimensional array called snowfalltotal that will hold the values above.

1. Calculate and print: the average snowfall totals for each city.
2. Calculate and print: the average snowfall totals of these five cities for each month.
3. Calculate and print: In December, which city received the least amount of snowfall.
4. Calculate and print: Which city received the most amount of snowfall between the months of November and March.
5. Calculate and print: Which month has the most amount of snowfall of these five cities.
6. Your file will have the following documentation comments before the class header:
7. **/\*\***
8. @Title: LastNameFirstSnowFallTotal
9. @Purpose: To practice two dimensional arrays
10. @Author: (your last first name)
11. @Date: (today’s date)
12. @Version: 1.0
13. \*/