**Points: 100**

**Individual Assignment**

**Due Tuesday, 01/15, at 11:59pm.**

**TASK:**

For this task you are to implement Example 4 from the C-to-Java-Examples.rtf as a Java Project.

You may cut and paste the code into the required class files.

You must create an appropriate header block (example below) for each class file.

STEP 1 – create a new Netbeans Project

* Open Netbeans
* From the menu bar select File -> New Project
* In the New Project dialogue
  + Select **Java** in Categories
  + Select **Java Class Library** in Projects
  + Press Next
* In the New Java Class Library dialogue
  + Enter the **Project Name** using the following naming convention:
    - Prefix-LastnameFM
    - Where prefix = the assignment prefix, Lab100 for this assignment
    - Followed by a hyphen
    - Followed by your last name with normal capitalization
    - Followed by the initial of your first name in upper case
    - Followed by the initial of your middle name in upper case
    - e.g. Lab100-LatimerJB
    - do not use the instructors name in your project
  + Use the default values for **Project Location** and **Project Folder** or changed them to your desired location and folder.
  + Press Finish

STEP 2 – create the Employee Java Class file

* In the Projects pane on the left of the Netbeans window select the Lab100 project (if it is not already selected).
* From the menu bar select File -> New File
* In the New File dialogue
  + Select **Java** in Categories
  + Select **Java Class** in File Types
  + Press Next
* In the New Java Class dialogue
  + Enter **Employee** for the Class Name
  + Press Finish
* In the Employee.java edit window
  + Delete the first comment block at the top of the page
  + Change the @author tag so that it shows **your name**
    - e.g. @author Joe Latimer
    - do not use the instructors name in your file
  + Add an @version tag followed by the creation or last modified date
    - e.g. @version August 23, 2017
  + Add a brief description of the purpose of the file
* Your comment block should look something like:

/\*\*

\*

\* @author Joe Latimer

\* @version August 23, 2017

\*

\* The Employee class contains the information about an employee.

\*

\*/

* Copy the Employee class from the C-to-Java Examples into your Employee class file
* Check for complier errors

STEP 3 – create the main Class file

* From the menu bar select File – New File
* In the New File dialogue
  + Select **Java** in the Categories
  + Select **Java Main Class** in File Types
  + Press Next
* In the New Java Main Class
  + Enter **Client** for Class Name
  + Press Finish
* In the Client.java edit window
  + Delete the first comment block at the top of the page
  + Change the @author tag so that it shows **your name**
    - e.g. @author Joe Latimer
    - do not use the instructors name in your file
  + Add an @version tag followed by the creation or last modified date
    - e.g. @version August 23, 2017
  + Add a brief description of the purpose of the file
* Your comment block should look something like:

/\*\*

\*

\* @author Joe Latimer

\* @version August 23, 2017

\*

\* A main class to test the Employee class.

\*

\*/

* Copy the Employee class from the C-to-Java Examples into your Employee class file
* Change the class name from Example4Client to **Client**
* Check for complier errors

STEP 4 – Run your program

* From the menu bar Select Run -> Project (Lab00)
* In the Run Project dialogue
  + Select the **Client** for the Select the main class
  + Press OK
* Check your output in the Output window

STEP 5 – Create your project zip archive

* Be sure the project is selected in the Projects window
* From the menu bar select File -> Export Project -> To Zip
* In the Export Project(s) to Zip dialogue
  + Use the Build Zip Browse button to have the zip file built in a convenient location such as the Desktop.
    - In the Save dialogue enter the zip file name as:
      * Lab100-LastFM.zip
      * Note you must specify the **.zip** extension
    - Press Save
  + Press Export
* Note that the File Explorer opens inside the zip archive. You need to go up one level to see your desktop and the zip file that you just created.

STEP 6 – Create your Word document (it must be a Microsoft Word .docx file)

* Open Microsoft Word whose name is the same as your project
  + **Lab100-LastFM**
* Copy and paste the contents of the Client.java file into the Word document
* Copy and paste the contents of the Employee.java file into the Word document
* Copy and paste the contents of the Output Window file into the Word document
  + Use <CTRL>A to capture all of the contents of the Output Windows.

STEP 7 – Submit both files as a single assignment submission on Blackboard.