School of Electrical Engineering and Computer Science
The University of Newcastle
SENG1110/SENG6110 Object Oriented Programming

Lab Session 4 – Week 5

This week we will start to use a new environment to edit, compile, run and debug our Java codes. It is called **BlueJ**. Later, **after** this computer lab, you can visit http://www.bluej.org/ and download BlueJ software in your computer. Today, you will learn how to edit, compile and run using BlueJ. We will learn other aspects in future computer labs (for example, how to debug).

It will be very useful to use an IDE (as BlueJ) for SENG1110/6110. In the subsequent courses you should use a more sophisticated IDE (more information will be available in Blackboard in the end of our course). The aspect more important and more difficult is to know how to debug a code using a tool like BlueJ. You will see some of these aspects in future computer labs.

1. Listen the **first video** available in the 'computer lab 4' in Blackboard to follow/complete the following exercise.

2. Download the program Student.java and TestStudent.java from Blackboard. Now follow the instructions below.
   - **a. Execute Bluej**. Just write Bluej in the command line.
   - b. **Open a new project:**
     - i. select new Project from project menu
     - **ii.** write StudentProject (this will create a folder called StudentProject and all Java files related to this project will be inside this folder)
   - c. **Add a class from a file**
     - i. Select add 'class from file' from 'edit' menu
     - ii. Select the file Student.java
     - iii. Add TestStudent.java as well
   - d. To **edit** your code, just clique twice and the code will open
   - e. To **compile**, just choose the compile menu.
   - f. To **run** the code
     - i. go to the Bluej window (go to TestStudent.java icon).
     - **ii.** use the right button of the mouse and choose main(). Don't worry about the other options now.

   (you need to complete the g. and h. show, using BlueJ, the result to your demonstrator)
   - g. Check the Student.java code and implement the method **getHighScore** (in the end of the file Student.java you will find more details). At this point, if you have difficult in understanding how the classes/objects work, listen the **second video** before completing this exercise.
   - h. Write a line of code that use the method getHighScore in TestStudent.java (use your imagination).

3. Bank example
   - a. Using **BlueJ**, choose **new project**
     - i. select new Project from project menu
     - **ii.** write BankProject
   - b. **Add a class from a file**
     - i. Select add class from file from edit menu
     - ii. Select the file BankAccount.java
     - iii. Again, select add class from file from edit menu
     - iv. Select the file BankAccountManager.java
   - c. Compile/run the program in the Bluej.
   - d. Be sure that you understand the codes.
   - e. Notice that BankAccount.java has the instance variables name, idNumber and balance. The BankManager.java just manages the balance. Modify the program in such a way that it can work with the name and the id of the client.
   - f. Write the methods getName and setName in BankAccount.java. Use them in BankAccountManager.java
   - g. Document the program well
   - h. Listen the **third video** to understand better the concept of encapsulation and how to generate documentation.

Prof Regina Berretta
Mar-2017