Learning Outcomes

Upon completion of this lab, you should be able to do the following in Eclipse:
- Make a constructor.
- Make getters and setters.
- Make a `toString()` method.
- Make an `equals()` method.
- Use a test harness to aid in testing methods of a class.

General Lab Instructions to Help Labs Run Smoothly

- Read through the lab instructions before coming to the lab.
- Do the pre-lab preparation.

Preparation. Make sure you understand the parts of a class definition: instance variable declaration, constructor definition, method definition.

IMPORTANT: Make sure you attend your lab session and sign an attendance sheet, show the result(s) of each exercise to your TA, and submit your .java files through OWL by 11:55pm on the same day as your lab session to get your marks for the lab.

Exercise 1: Creating the Person Class

The `Person` class encapsulates a person using the private variables:
- `firstName`: the first name of the person (of type `String`),
- `lastName`: the last name of the person (of type `String`), and
- `email`: the email address (of type `String`).
and the methods:
- A constructor method, `Person()`, with 3 parameters: the person’s first name, last name, and email address. This method creates an instance (object) of the class `Person`.
- Getter methods: `getFirstName()`, `getLastName()`, and `getEmail()` that return the first name, last name, and email address, respectively.
- Setter methods: `setFirstName()`, `setLastName()`, and `setEmail()` that saves the first name, last name, and email, respectively.
- Method `toString()`, that returns a string containing the values of the object’s private variables.
- Method `equals()`, with 1 parameter: a person object. This method should return true if the first and last names are the same same for both people; otherwise, this method should return false.
Exercise 2: Adding a Test Harness to the Person Class

To make it easier to test code that you have written for a Java class you can add to that class a main method that acts as a "test harness". A test harness is a main method that includes calls to methods that you wish to test. For convention this main method is the last method of the class.

If you have a test harness, you do not need to have a separate application program in order to test your class: the test harness is part of the class you are testing. In this part of the Lab, you will get some experience in writing a test harness.

- Add a main method to your Person.java.
  - Create a Person object called friend1 using the constructor Person(). Use the arguments Mickey, Mouse, and "".
  - Set friend1’s email address to mickey@uwo.ca.
  - Print out to the screen the toString() method of friend1.
  - Print out to the screen friend1’s first name.
  - Print out to the screen friend1’s email address.
  - Create a Person object called friend2. Use Minnie, Mouse, and "".
  - Print out to the screen the toString() method of friend2.

- Run Person.java; it will execute the test harness (main method).

- Now add code to the test harness to test the equals() method: it should print a message like "same friend" if two objects of the class Person are the same according to the equals() method documentation (i.e. if they have the same first name and last name), or print "not same friend" if they are not the same. (Hint: you can use friend1 and friend2 to compare two objects of class Person that are not the same, but for thorough testing, you should also create a third Person object who has the same name as one of the first two.)

- Run Person.java again to execute the changed test harness.