CSD221 W18 Lab 1

Goals:

* Set up a Netbeans project using Maven
* Add dependencies to a Maven project
* Track program revisions on a project using git
* Write a simple program using OOP principles learned in class
* Discuss how the OOP principles from today’s class apply to your program

Submission: [How to Submit CSD221 Labs](https://docs.google.com/document/d/1HBW_otFOBU2tWDPexTCU8VjPpkaxyYVUXZOekEDbqhc/edit?usp=sharing)

# Documentation & Formatting [5 marks]

* For this lab, you really don’t need to comment much, if at all, because the program is very simple and easy to understand
* However, make sure to use appropriate class and variable names, and to format your code well to obtain all 5 marks

# Question 1 [0 marks]

Please complete the ‘Tell Me About You’ survey in LMS

# Question 2 [5 marks]

Note: This question gets you acquainted with the process of setting up a Maven based Java Application, and adding dependencies. Going forward in this course, unless otherwise requested, any lab questions that ask you to create a project should do so using a Maven based Java Application as in this question.

* Create a Maven-based Java Application project in NetBeans.
* Use the following format for your Project Name: CSD221\_W18\_<FirstnameLastname>
  + Example: CSD221\_W18\_RodneyMartin
* Use ca.saultcollege as your Group Id
* Add the junit library (Group Id: junit; Artifact Id: junit; Version: 4.12; Scope: test) as a dependency to your project

**Note: Do all the questions below in the project you just created in Question 2**

# Question 3 [10 marks]

Note: This question gets you acquainted with the process of setting up a git repository for a NetBeans project, and using it to track your project files.

* Initialize a git repository
* As you proceed with the questions in this lab, track ALL your project files in git.
  + Make sure to include in your repository the pom.xml file in your project root that Maven makes for you automatically
  + There will probably also be a .gitignore file in your project root as well—add it to your repository too
  + Dependencies do not need to be tracked
* To get full marks for this question, make sure your **most recent changes** have been **committed** to your repository for this lab (not just added to stage) and **pushed** to the project you set up for this lab in GitLab. (See [Configuring Your GitLab Account for CSD221](https://docs.google.com/document/d/1-Z3TFr1yKYlPEJ9GxWie_mQNTimWOA2KuX9NH0dBT3s/edit?usp=sharing) )
* See [How to Submit CSD221 Labs](https://docs.google.com/document/d/1HBW_otFOBU2tWDPexTCU8VjPpkaxyYVUXZOekEDbqhc/edit?usp=sharing) for instructions on how to submit your Lab using git.

# Question 4 [10 marks]

I would like a silly application that can print greetings displaying a range of emotions.

* In the main package of your project (ca.saultcollege.csd221\_w18\_<yourname>) create a class named ‘Greeter’ that has the following method:

public void greet() {

System.out.println("Hello, world!");

}

* Create at least two more classes that are subclasses of Greeter that override the greet method to print greetings specific to a particular emotion. Eg. A SadGreeter might print “Hello, cruel world!”; a HappyGreeter might print “Hello, wonderful world!”.
* In at least one of your subclasses, add a second method that prints a different message relevant to that class’s emotion. Eg. A SadGreeter might have a ‘cry’ method that prints ‘Waaaa!’
* Finally, create a class named ‘Main’ that contains the Java main method. In this method, instantiate one of each of the classes you created above (including the superclass Greeter) and invoke the greet method on each one.

# Question 5 [10]

* Create a new package in your project called ca.saultcollege.csd221\_w18\_<yourname>.discussion and to this package add a new file called Answers.txt
* In this file, describe how each of the following concepts apply to the set of classes you created in Question 4
  + Objects & Classes
  + Abstraction
  + Encapsulation
  + Generalization
  + Polymorphism

Your completed NetBeans project tree should look something like this:

