CSCI 2134 Assignment 1

Due date: 11:59pm, Sunday, September 24, 2023, submitted via Git

Objectives

- Set up your laptop with all the tools that you will need for this course.
- Demonstrate basic use of Git and the IntelliJ IDE.

Task 1: Install the Required Tools

You will need to install Git, the Java JDK, and the IntelliJ IDE on your laptop.

Test Your CSID

Log into the Faculty Git server to test your CSID at the following URL
https://git.cs.dal.ca/
If you do not know your CSID or password, go to this URL
https://csid.cs.dal.ca

Install Git on Your Laptop

You will also need to install the Git client on your laptop. If you already have Git installed, you can skip this step. If not, you can download and install a Git client by going to the following URL for the corresponding OS:

- Mac OS X: https://sourceforge.net/projects/git-osx-installer/files/
- Windows: https://gitforwindows.org
  The default settings should be fine for the most part. **Note:** If you do not have Microsoft Visual Studio installed, select the second cmd.exe option instead of minTTY when asked what the default terminal window should be.
- Linux: https://www.atlassian.com/git/tutorials/install-git#linux and downloading the appropriate distribution for your platform

Note: If you configure your user id as the instructions in this tutorial suggest,
https://www.atlassian.com/git/tutorials/install-git
use your "Firstname Lastname” as the user name.

Create a `git` Directory on Your Laptop (Strongly Recommended)

It is very s**trongly** recommended that you create a **`git`** directory on your laptop in your home directory (in the same directory that your `Documents` directory and other main directories are located). This is just a regular directory called “git” created using your file browser or the mkdir command.

Inside the **`git`** directory it will be useful to create course subdirectories, such as `csci2134`, and in csci2134, you should create a directory for Assignment 1, e.g., `assn_1`.

Install Java on your Laptop

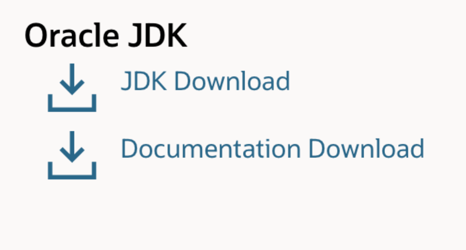
If you already have Java installed on your laptop, you can skip this step. To install Java:

1. Go to URL: https://www.oracle.com/technetwork/java/javase/downloads/index.html,
2. Select the "JDK DOWNLOAD" option, and
3. Download the package for your preferred platform. We recommend:

- Mac OS X: jdk-19_macos-x64_bin.dmg
- Windows: jdk-19_windows-x64_bin.exe
- Linux:
    - jdk-19_linux-x64_bin.deb
    - jdk-19_linux-x64_bin.rpm
    - jdk-19_linux-x64_bin.tar.gz

  Depending on your Linux flavour.

Note: The version number, such as 19.01, may be later than the version in these instructions. Any recent version of Java should work for this course.



There is a video available in Brightspace on how to install the JDK on Windows:

https://dal.brightspace.com/content/enforced/283393-10841.202410/Resources/Java_JDK_Windows.mp4

Installing IntelliJ IDE on Your Laptop

We will be using IntelliJ from JetBrains in this course. If you already have an up to date version of the IDE installed, you can skip this step. The professional version of this IDE is available free on an education license. To download and install the IDE go to the following URL

https://www.jetbrains.com/products.html#type=ide.

You should apply for the educational licence at this URL:

https://www.jetbrains.com/community/education/

Note: You MUST use your Dalhousie Net ID email when applying for this license.

There is a video available in Brightspace on how to install the IDE on Windows:

https://dal.brightspace.com/content/enforced/283393-10841.202410/Resources/IntelliJ_Windows.mp4

and on Mac:

https://dal.brightspace.com/content/enforced/283393-10841.202410/Resources/IntelliJ_macOS.mov

Note: The first time you run IntelliJ, you will be asked to register the license. You will need the login ID and password you created when you applied for the educational licence.

Task 2: Use the Required Tools

Get Assignment 1 Code

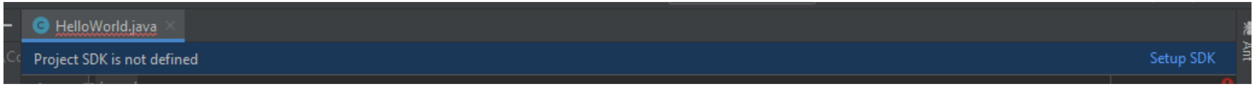
1. Open a command-line window (`Terminal` on a Mac or `gitbash` on Windows)
2. Change directories to where you would like to store a local copy of Assignment 1, e.g.,
   ```
   cd git/csci2134/assn_1
   ```
3. Clone the repository for your Assignment 1 using the following command and Git URL
   ```
   git clone https://git.cs.dal.ca/courses/2023-fall/csci-2134/assignment1/????.git
   ```
   where ???? is your CSID. **A directory with the same name as your CSID will be created, containing Assignment 1 code.**
4. Change directories into the cloned repository
   ```
   cd ????
   ```
   where ???? is your CSID.

Branch the Assignment 1 Code

5. Create a branch of the project called `development` on which you will make your modifications:
   ```
   git branch development
   ```
6. Checkout the branch, i.e., move to the branch you have created
   ```
   git checkout development
   ```
7. Check that you are on the right branch by performing a status check
   ```
   git status
   ```
   The first line of the output should state: **`On branch development`**

Modify Assignment 1 Code

8. Run IntelliJ and open the project you cloned.
9. Select **Project View** (Command-1 on Mac, Alt-1 on Windows) and expand the `src` directory
10. Select the `HelloWorld.java` file in the IntelliJ IDE and open it.
    Note: The first time you run IntelliJ you may need to set the JDK that IntelliJ should use.



Project SDK is not defined

    There will be an error status at the top of the IDE with a "**Setup SDK**" button for you to click and select a JDK. If you have a choice, select the latest one.
11. Click on the small green triangles beside the class name and the `main` method in the IDE to run the program.
12. Try running the program as is. You should see the following in the bottom console window:
    ```
    Hello ????!
    ```
13. Make the following modifications:
    a. Change line 18 so that `bannerNumber` is set to your banner number
    b. Change line 21 so that `csid` is set to your CSID
    c. Change line 24 so that `lastName` is set to your last name
    d. Change line 27 so that `firstName` is set to your first name
14. Compile and run the program. You should see the program output in the console window
    ```
    Hello _____!
    ```
    where _____ is your first name.

Add, Commit, Merge, and Push Changes

15. Switch back to the command line and prepare to commit the changes you made. This is called *staging*:
    ```
    git add -A
    ```
    This command stage all new or modified files.
16. Check to ensure that the files have been staged
    ```
    git status
    ```
    The modified files should be listed.
17. Commit the changes
    ```
    git commit -m “Set key variables in Assignment 1”
    ```
18. Push the branch back up to the repository
    ```
    git push -u origin development
    ```
    This ensures that your changes are not going to be lost if your computer dies.
19. Checkout the main branch
    ```
    git checkout main
    ```
20. Merge the `Development` branch back to the main branch
    ```
    git merge development
    ```
21. Push the merged main back to the repository
    ```
    git push origin main
    ```
22. **You’re done!**

Aside: the main or primary branch in git repositories was historically called the “master” branch. This has negative connotations so in this class we tend to use the name main instead, as are many large git repositories. You may occasionally see the main branch called master in this course or in other software repositories.

Assignment Submission

Step 21 of Task 2 submits your assignment to the Git server. You may submit as many times as you wish prior to the due date.

Grading

The following grading scheme will be used:

Grade	Level of Completion
100%	All steps of the assignment were completed and all software installed as required. This will be confirmed by checking the submission in the Git repository
50%	Software installed, but assignment was not completed. This must be confirmed by a TA or the professor in the lab or during office hours.
0%	Software was not installed on the laptop.

Note: Students will have opportunity to get help with Assignment 1 during Labs 0 and 1, as well as during office hours of the instructors and teaching assistants.