CSCB07 Lab 2 — Subversion, Eclipse

To earn lab marks, you must arrive on time and actively participate for the entire session.

1 Overview

This week, you are going to work with Subversion and Eclipse IDE.

2 Reminder: Choose a driver and navigator

In these labs, you are encouraged to work in pairs. You and your partner together will be able to figure out problems better than you would individually. Find yourself a partner. If you have trouble finding one, let your TA help you. This partnership is only for today's lab.

We strongly advise you to form a partnership with a colleague who has a similar level of background. For example, if this is your first time working with Eclipse, we suggest that you partner with someone who is also new to Eclipse.

In all the labs, we will use the terms *driver* and *navigator*. Here are the definitions of the two roles:

- **Driver:** Types at the keyboard. Focuses on the immediate task at hand.
- **Navigator:** Thinks ahead and watches for mistakes.

In lab handouts, we'll often refer to you as `s1` and `s2`, and `s1` will be the first driver.

Tip: If you are new to CDF labs, you should be `s1` to get practice logging in while `s2` helps.

3 Subversion

In this part of the lab, we continue to work with subversion. Recall that the URL for your repository is:

`https://markus.cdf.toronto.edu/svn/csc207-2015-01/your_CDF_username/`

1. Change to `s1`'s home directory.
2. Checkout `s1`'s repository in two different directories (you decide which directories and whether you need to create them). In the rest of these instructions, we will refer to them as `location1` and `location2`.
3. In your repository, you will find a newly created directory `lab2`.
4. In directory `lab2`, create a file called `testfile.txt`, and perform the necessary sequence of subversion commands to ensure that the two locations have the same copy of the file.
5. Modify the file in `location1`. Modify the file in `location2` differently. Make sure that the two modifications do not affect the same line in the file.
6. Perform a commit in `location1`.
7. Perform an update in `location2`. Observe that subversion was able to merge the changes without your help.
8. Perform the necessary steps to make sure `location1` and `location2` have the same version of the file — the result of subversions's merge.

   Switch roles!

9. Now modify the file in `location1` and `location2` in a way that will create a conflict.
10. Perform a commit in `location1`. Perform an update in `location2`. Resolve a conflict by editing the file.
11. Perform the necessary steps to make sure `location1` and `location2` have the same version of the file — the result of your edits.
12. If time permits, experiment with subversion: What kinds of modifications result in subversion being able to successfully merge? Which modifications create a conflict? For example, try to add to the end of file in one location and modify the first line in another location.

    Switch roles!

4 Eclipse

1. Open up Eclipse by issuing the command `eclipse-android &`. (The `&` specifies that this application should run in the background, which allows you to continue to issue command line instructions.)
2. Create a project named `Lab2`.
3. Create two packages, `lab2p1` and `lab2p2`.
4. In package `lab2p1`, create a Java class `MyLab2` with a `main` method.
5. In the `main` method, create and assign to variables of types `int`, `double`, `boolean`, `char`, `Integer`, `Double`, and whatever else you're familiar with in Java. Reassign some of these variables (give them new values).
6. Print out the values of these variables with appropriate messages.
7. Run your program.

    Switch roles!
8. Browse Eclipse's settings, figure out the contents of default panes, resize them, change font size to suit your preference, etc. Spend some time creating a comfortable working environment.
9. Run a debugger. Make sure you trace the values of all variables you created and observe all changes that result from assigning them new values.

Make sure you show your TA your work!

Looking for more practice?

For more practice using SVN, add your `Lab2` project to version control. Don't commit the files and directories generated by Eclipse, such as bin, .project, .class, etc.