Getting Ready to Code

Linux Commands

An important part of using Linux is becoming familiar with the command line. The following exercise will get you started with a few commands you'll find useful in the coming week.

1. ls is the command to list the contents of the current directory. When you first open the terminal and have a command prompt, you will be in your home directory. The home directory is denoted by a ~. Try typing ls into the command line now and hitting Enter to see what the contents of your home directory are.
2. cd is the command to change directories. Choose one of the directories you saw listed when you entered the ls command and type cd <chosen_directory>into the command line (without angle brackets). Hit Enter.
3. *(Note that the command line has the idea of tab-complete. If you start typing the name of a unique file, directory, or command and hit tab, it will finish the name for you. If there is more than one match, all possible options are displayed.)*
4. When you're using a terminal to navigate the filesystem, you may want to go back up a directory level to the *parent directory*. Just as your home directory has the ~ alias, the parent directory of the directory you are currently in has the alias .. (yes, that is a dot dot). Type cd .. into the command line and hit Enter to return to the parent directory.
5. pwd will tell you what directory you are currently in. If you started this little exercise in your home directory, typing pwd at the command line and hitting Enter will tell you that you ended in your home directory, though not with a simple tilde; pwd gives you the full, absolute path. Likely, your terminal has already been configured to display this information at the start of its command prompt, but if not, pwd is a good way to keep from getting lost. Also remember that at any point, you may enter the command cd ~ to return to the home directory.

The above is essentially a command-based version of what you're used to doing by double clicking on folders. The command line is capable of a great deal more -- in fact, you will be using it to compile your tutorial program! But more on that later. First, you'll need to write some code.

Text Editor

In this lab, we'll be using a text editor called **gedit** to write our tutorial program in C++. We'll use a different environment for Project Illuminate. To open gedit, open up a terminal window, type in gedit & and hit Enter. The & tells the computer to allow you to continue to use that terminal while the editor is running. What pops up is a basic text editor, but between it and the command line, you will be able to write and execute your tutorial programs.

Source