

# CLI (Command Line Interface) and First Assignment Tutorial

## Command Line Interface and First Assignment Tutorial

Intro: This tutorial is made to assist those who little to no experience using a command line interface (see following paragraph for description of what this is). It will consist of several parts addressing the following topics:

1. What a command line interface (CLI) is
2. How to access the LSU server
3. Some basic navigation techniques in a CLI/the LSU server
4. How to compile and submit on/to a server (In this case LSU's)
5. Answers to some frequently asked questions

## What is a CLI?

To best understand how to navigate and use the servers at LSU, a person has to understand the difference between a GUI (graphical user interface) and a CLI (command line interface). A GUI is an interface in which the user interacts with easy to understand buttons, icons, text boxes, etc. Windows and most applications that run on Windows are GUIs.



## GUI examples

In contrast to these pretty GUIs, we have a CLI (command line interface). To use a CLI, the user has to type in lines of text called "command lines" to perform a task. There are several different CLIs but the one we will be dealing with specifically is Linux (the interface used by LSU).

```

xterm
--revprop          : operate on a revision property (use with -r)

dave@bolsh:~/work/articles$ svn proplist -v *
Properties on 'svn_browser.png':
  svn:mime-type : image/png
Properties on 'svn_design.png':
  svn:mime-type : image/png
dave@bolsh:~/work/articles$ ls
CVS_linuxmagazine.txt      Subversion_resume_linuxmagazine.txt
CVS_resume_linuxmagazine.txt  svn_browser.png
Subversion_linuxmagazine.txt  svn_design.png
dave@bolsh:~/work/articles$ svn status
?      .xvpics
M      Subversion_linuxmagazine.txt
A      svn_browser.png
A      svn_design.png
dave@bolsh:~/work/articles$ rm -rf .xvpics/
dave@bolsh:~/work/articles$ svn commit -m "Almost finished draft"
Sending          Subversion_linuxmagazine.txt
Adding (bin)    svn_browser.png
Adding (bin)    svn_design.png
Transmitting file data ...
Committed revision 10.
dave@bolsh:~/work/articles$

```

### CLI example

Why Use CLI? although CLI may be more difficult to use than a GUI, it takes significantly less memory on the server (improving performance for each user) and can be substantially more powerful in certain instances.

## Logging Into and Using the LSU Servers

### Step 1:

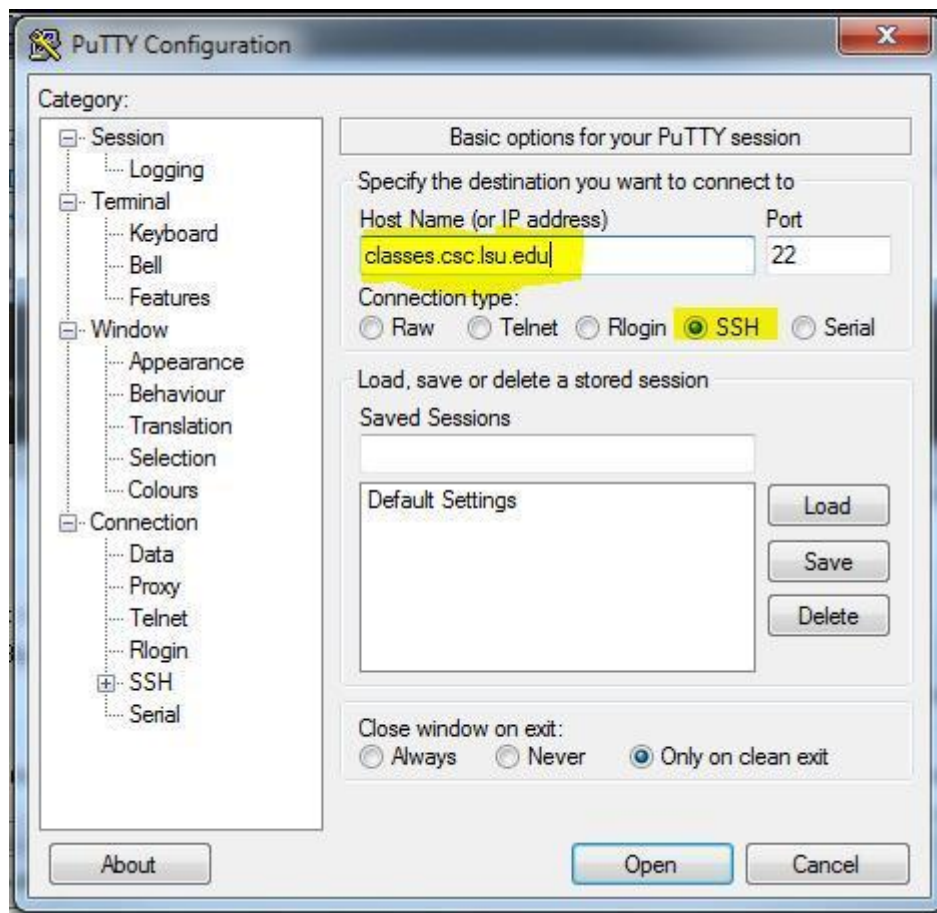
For Windows: If you are running Windows and have not done so already, you will need to download an SSH client to access the servers. A popular SSH client called PuTTY can be downloaded at this link:

<http://the.earth.li/~sgtatham/putty/latest/x86/putty.exe>

For OSX/Linux: If you are using a OSX (the mac operating system) or linux, you already have an SSH client by default. To access it, go to spotlight (the little magnifying glass in the top-right of your screen) and type in "terminal" (without the parentheses) in the search bar. Open terminal and proceed to the OSX/linux portion of step 2 for further instructions.

### Step 2:

For Windows: Open up Putty and type in classes.csc.lsu.edu using port 22. make sure that the SSH radio button is checked. Your client should match the one shown below when you finish. Click "Open" when you are done.



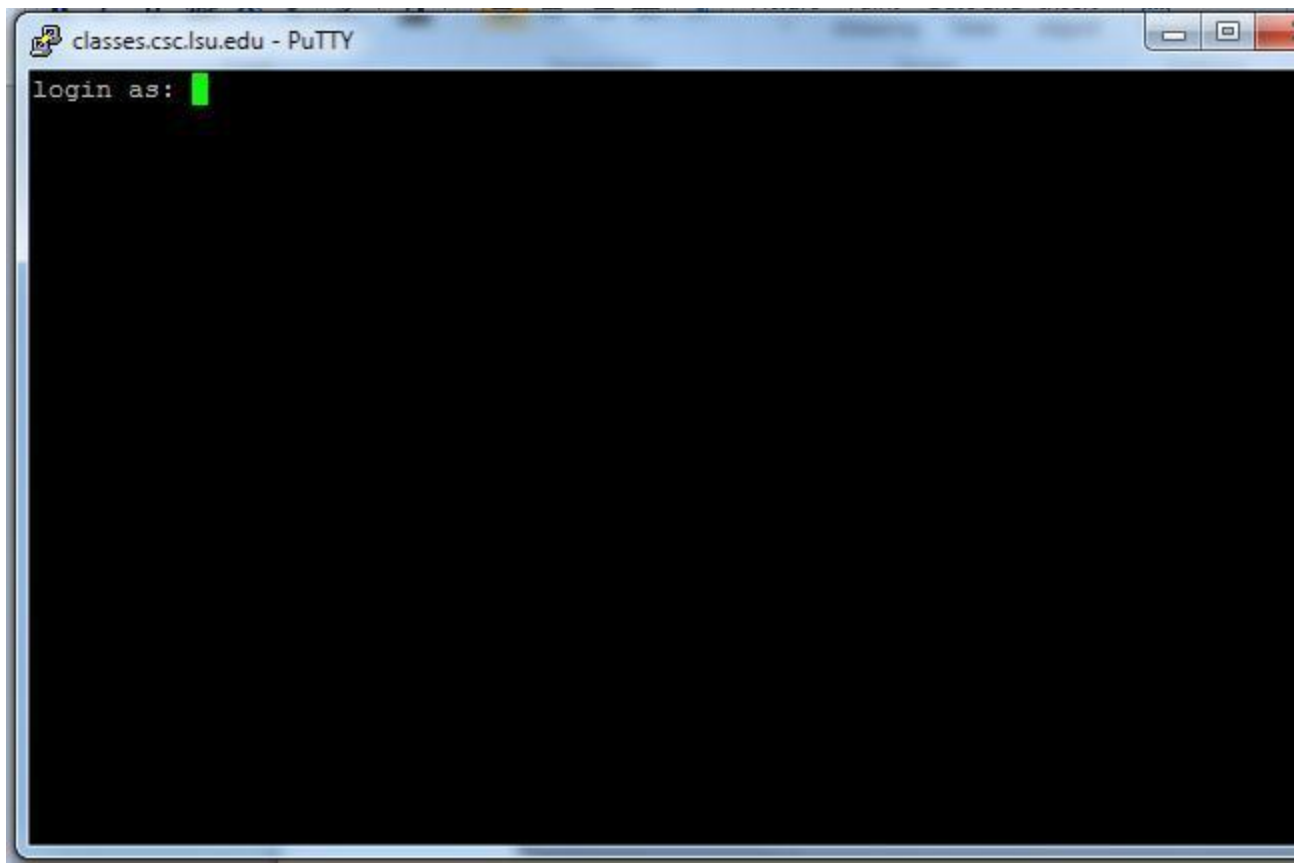
For OSX/Linux: in the terminal type in the following:

```
ssh -l (your username) classes.csc.lsu.edu
```

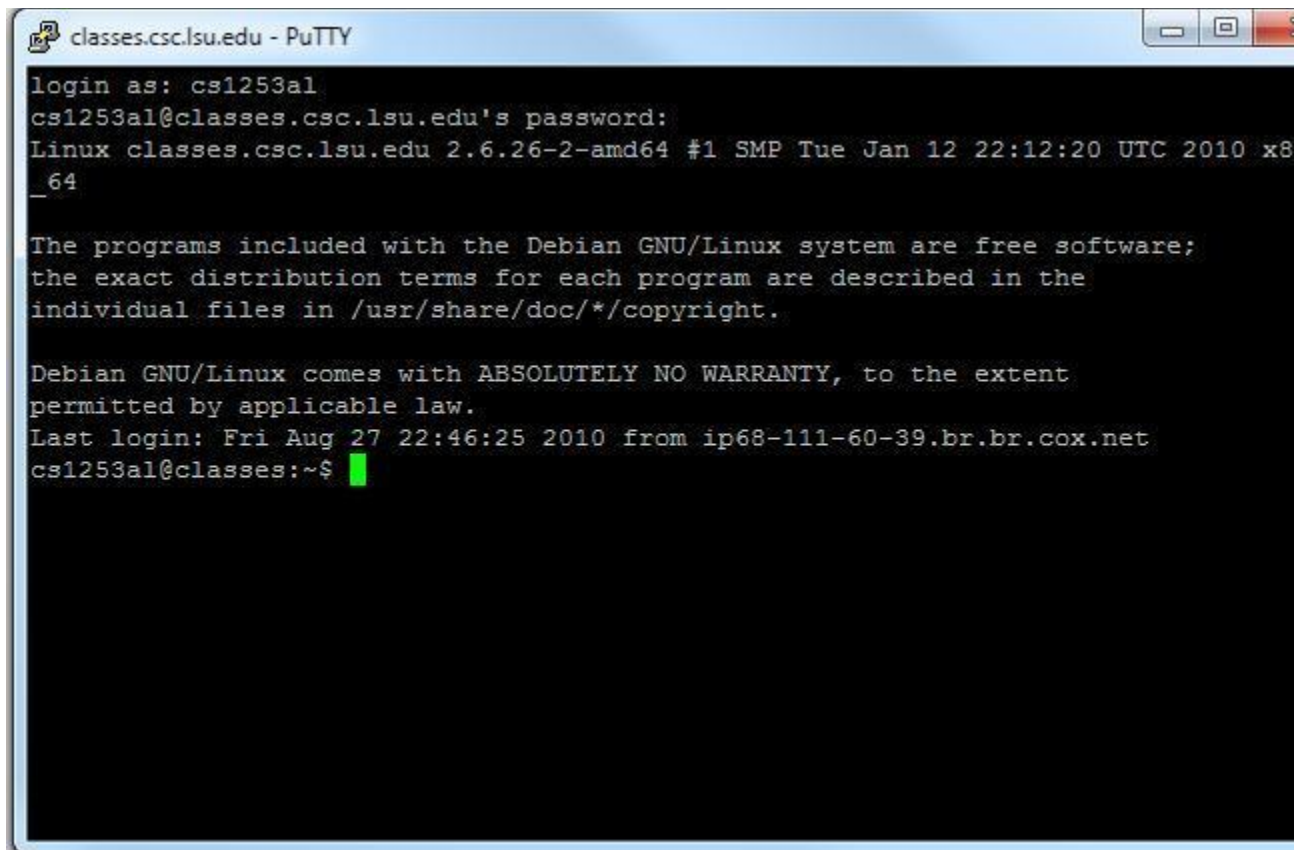
in place of "your username", type the username provided to you earlier this week. hit enter.

### Step 3:

For Windows: You should now see a screen resembling the one below



In this window, type in your username provided to you earlier this week. After doing this, the PuTTY client should ask for a password. Once it does, enter the password provided to you earlier this week. If you did everything correctly, you should see a screen resembling the one below.

A screenshot of a PuTTY terminal window titled "classes.csc.lsu.edu - PuTTY". The terminal shows a login process for user "cs1253al". The prompt "login as: cs1253al" is followed by "cs1253al@classes.csc.lsu.edu's password:". Below that, the system information is displayed: "Linux classes.csc.lsu.edu 2.6.26-2-amd64 #1 SMP Tue Jan 12 22:12:20 UTC 2010 x86\_64". A message about Debian GNU/Linux software licenses follows, stating that programs are free software and that the distribution terms are in /usr/share/doc/\*/copyright. Another message states that Debian GNU/Linux comes with absolutely no warranty. The last login information is shown: "Last login: Fri Aug 27 22:46:25 2010 from ip68-111-60-39.br.br.cox.net". The terminal ends with the prompt "cs1253al@classes:~\$" and a green cursor.

For OSX/Linux: You should now see a prompt to enter your password. Enter the password provided to you earlier this week. If you have done everything correctly up to this point, you should see a window similar to the one above (except with black text and a white backdrop).

## Using/Navigating the Server

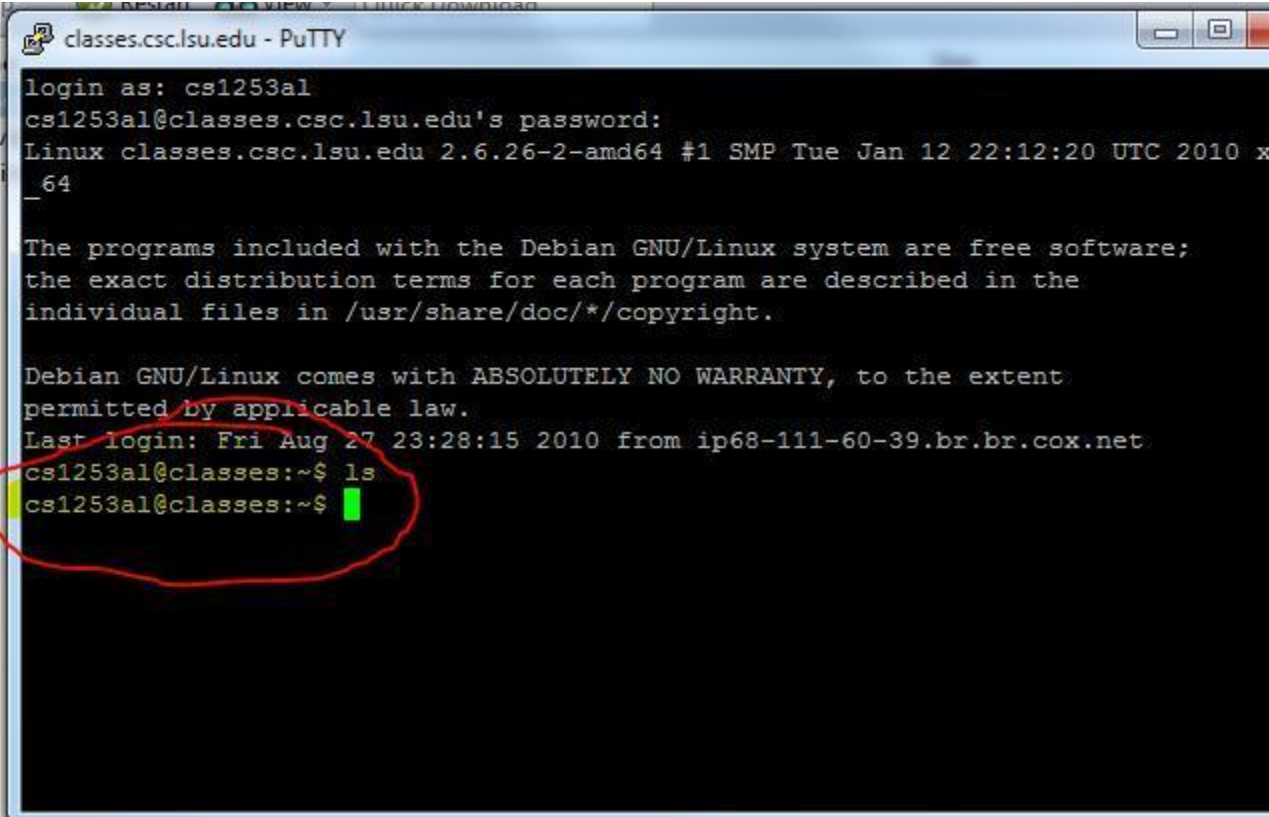
Now that you are logged in to the server, we can begin learning to navigate it properly. The navigation will work the same for Windows/OSX/Linux as we are now using the LSU server and NOT using our own computers (to a certain extent). When you logged in to the server, you were put into your "home" folder. the server is made up of several levels or "tiers" of folders. within each folder is either files, more folders, or both. This can sound confusing at first, but it becomes clearer as you learn to navigate the server. At this point, we will begin learning how to navigate. Let's start with three basic navigation commands

"ls" -This command shows all folders and files that are contained within the folder you are in.

"cd (file/folder name)" -This command enters the file or folder designated. replace "(file/folder name)" with the file or folder you wish to enter.

"cd .." -This command Brings you up a level or "tier" of folders

Now that we have these basic commands, lets play around with them a little. Using window from the "Logging Into and Using the LSU Servers" section, try typing in the command "ls" (without the parentheses)

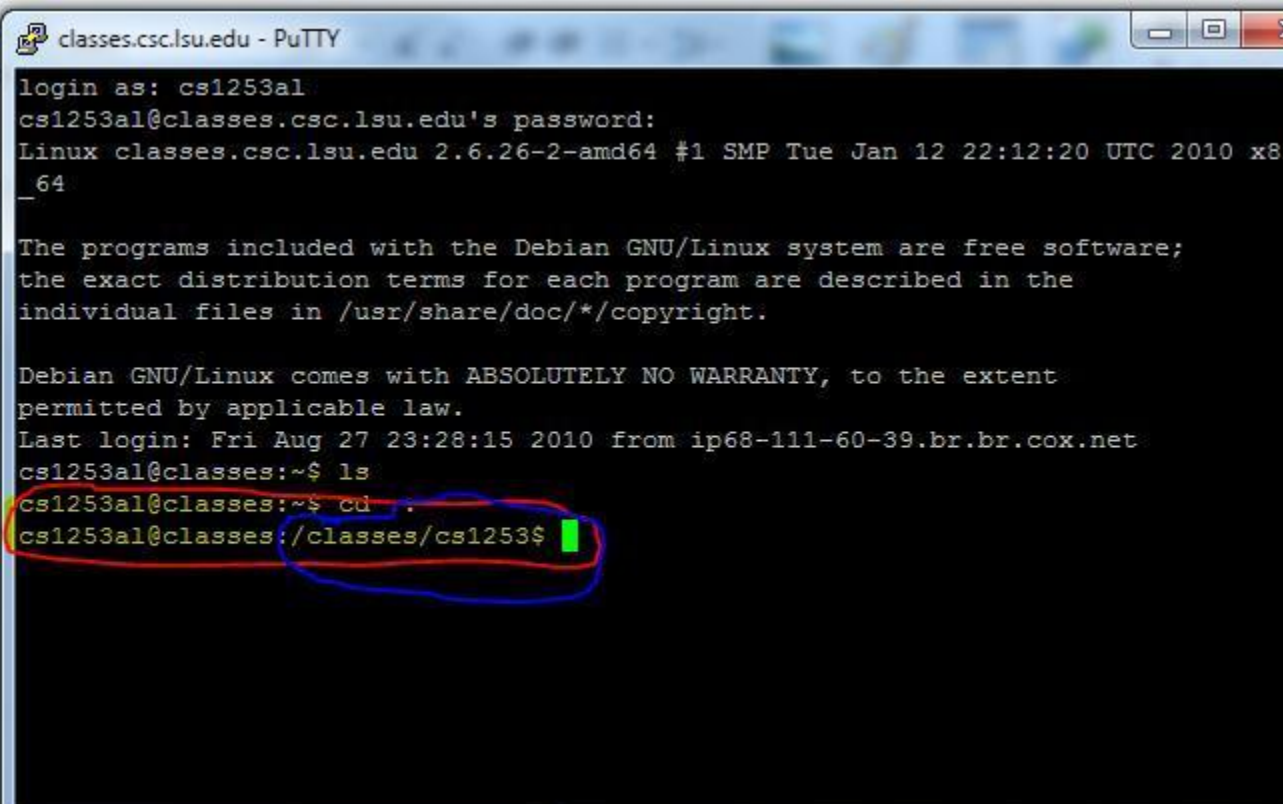


```
classes.csc.lsu.edu - PuTTY
login as: cs1253al
cs1253al@classes.csc.lsu.edu's password:
Linux classes.csc.lsu.edu 2.6.26-2-amd64 #1 SMP Tue Jan 12 22:12:20 UTC 2010 x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Fri Aug 27 23:28:15 2010 from ip68-111-60-39.br.br.cox.net
cs1253al@classes:~$ ls
cs1253al@classes:~$
```

You probably noticed that nothing happened. This is because there are currently no files or folders within your account. Let's see what happens if we type in the "cd .." command. (remember that the "cd .." command brings us up a level of folders)

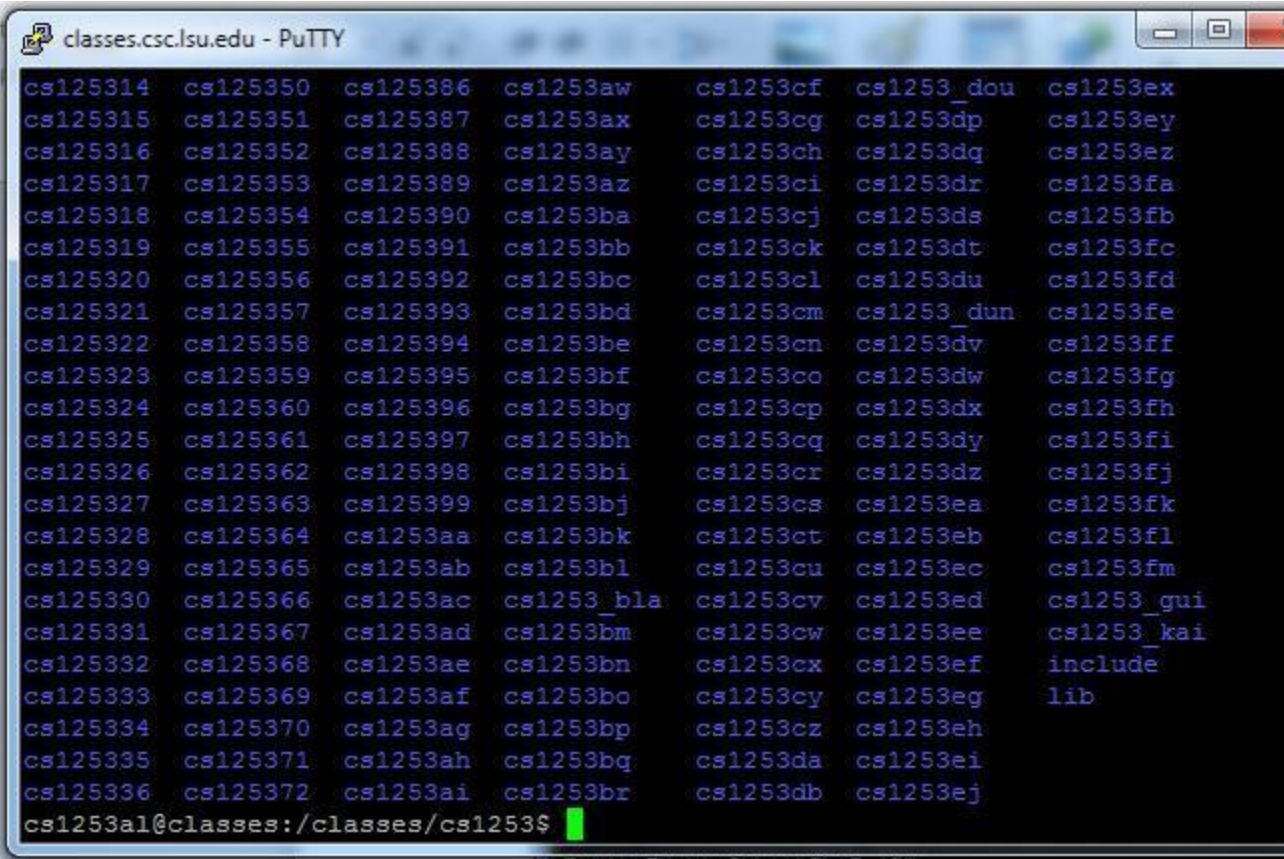


```
classes.csc.lsu.edu - PuTTY
login as: cs1253al
cs1253al@classes.csc.lsu.edu's password:
Linux classes.csc.lsu.edu 2.6.26-2-amd64 #1 SMP Tue Jan 12 22:12:20 UTC 2010 x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

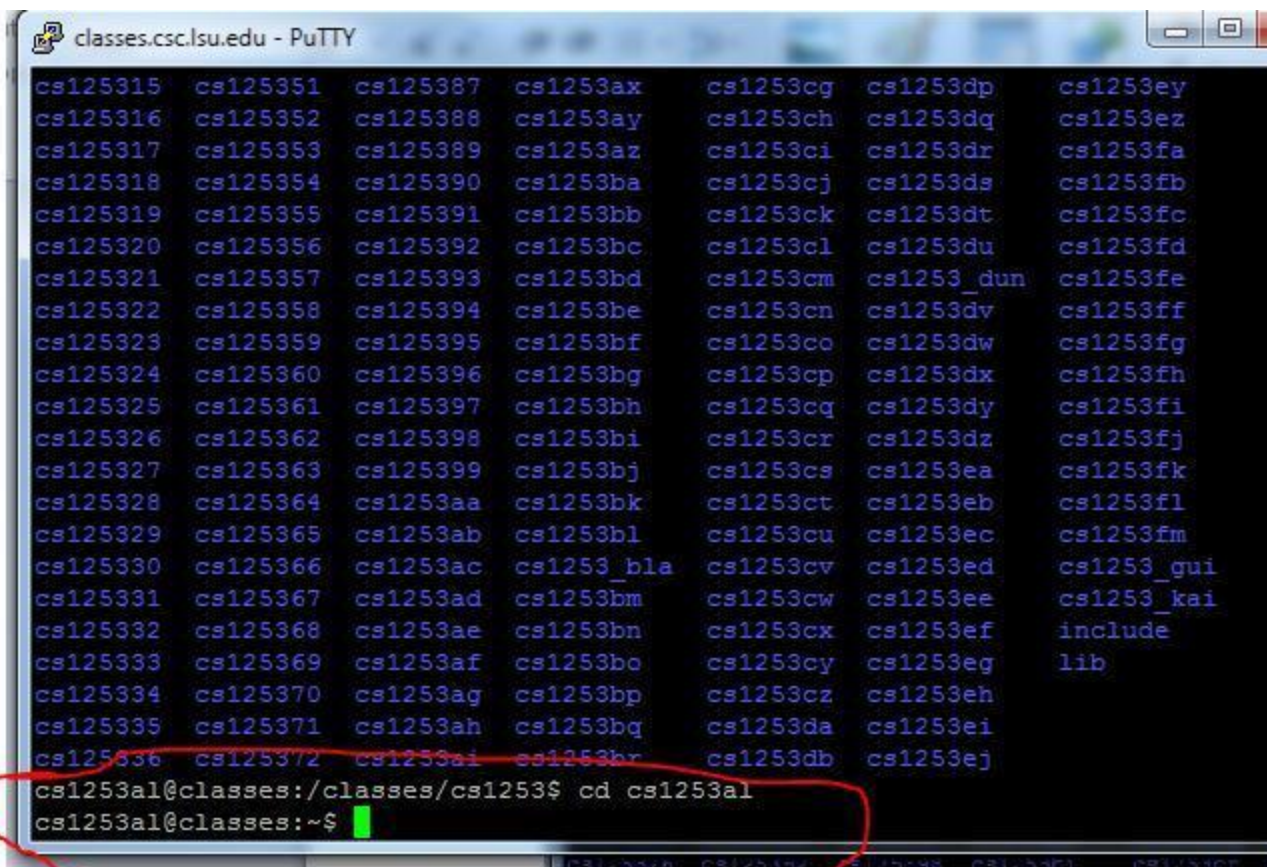
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Fri Aug 27 23:28:15 2010 from ip68-111-60-39.br.br.cox.net
cs1253al@classes:~$ ls
cs1253al@classes:~$ cd ..
cs1253al@classes:~/classes/cs1253$
```

After you types in "cd ..", a new line should have appeared. This part is circled in blue in the picture above. these are an indication of which folder we are now located in. Now that we have gone up a level of folders, try typing in "ls" again.



```
classes.csc.lsu.edu - PuTTY
cs125314 cs125350 cs125386 cs1253aw cs1253cf cs1253_dou cs1253ex
cs125315 cs125351 cs125387 cs1253ax cs1253cg cs1253dp cs1253ey
cs125316 cs125352 cs125388 cs1253ay cs1253ch cs1253dq cs1253ez
cs125317 cs125353 cs125389 cs1253az cs1253ci cs1253dr cs1253fa
cs125318 cs125354 cs125390 cs1253ba cs1253cj cs1253ds cs1253fb
cs125319 cs125355 cs125391 cs1253bb cs1253ck cs1253dt cs1253fc
cs125320 cs125356 cs125392 cs1253bc cs1253cl cs1253du cs1253fd
cs125321 cs125357 cs125393 cs1253bd cs1253cm cs1253_dun cs1253fe
cs125322 cs125358 cs125394 cs1253be cs1253cn cs1253dv cs1253ff
cs125323 cs125359 cs125395 cs1253bf cs1253co cs1253dw cs1253fg
cs125324 cs125360 cs125396 cs1253bg cs1253cp cs1253dx cs1253fh
cs125325 cs125361 cs125397 cs1253bh cs1253cq cs1253dy cs1253fi
cs125326 cs125362 cs125398 cs1253bi cs1253cr cs1253dz cs1253fj
cs125327 cs125363 cs125399 cs1253bj cs1253cs cs1253ea cs1253fk
cs125328 cs125364 cs1253aa cs1253bk cs1253ct cs1253eb cs1253fl
cs125329 cs125365 cs1253ab cs1253bl cs1253cu cs1253ec cs1253fm
cs125330 cs125366 cs1253ac cs1253_bla cs1253cv cs1253ed cs1253_gui
cs125331 cs125367 cs1253ad cs1253bm cs1253cw cs1253ee cs1253_kai
cs125332 cs125368 cs1253ae cs1253bn cs1253cx cs1253ef include
cs125333 cs125369 cs1253af cs1253bo cs1253cy cs1253eg lib
cs125334 cs125370 cs1253ag cs1253bp cs1253cz cs1253eh
cs125335 cs125371 cs1253ah cs1253bq cs1253da cs1253ei
cs125336 cs125372 cs1253ai cs1253br cs1253db cs1253ej
cs1253al@classes:/classes/cs1253$
```

We can now see the folders that are contained within the folder we are now navigating. To enter another folder/file, all we need to do now is type in "cd (file/folder name)". Try entering our home folder again (the folder name is the same as your user name).

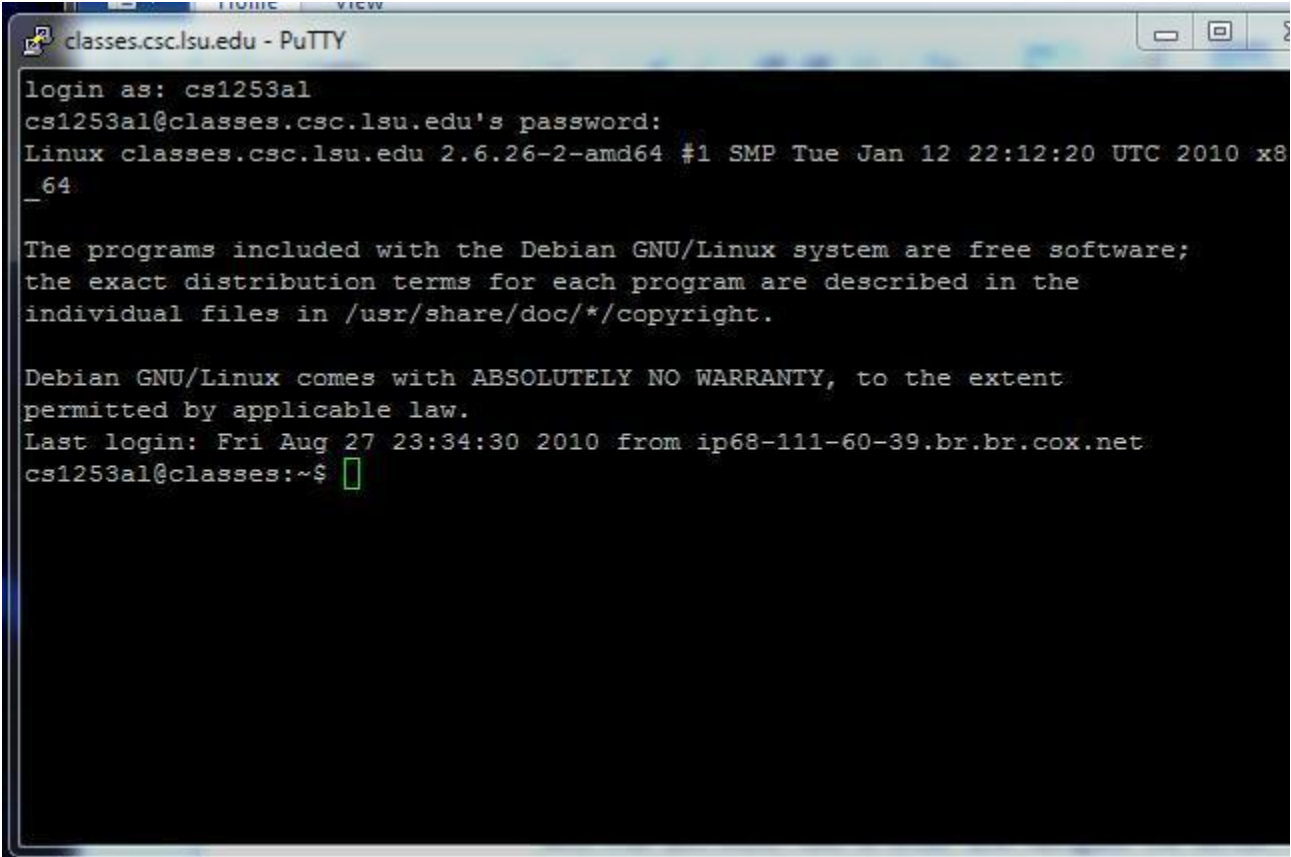


```
classes.csc.lsu.edu - PuTTY
cs125315 cs125351 cs125387 cs1253ax cs1253cg cs1253dp cs1253ey
cs125316 cs125352 cs125388 cs1253ay cs1253ch cs1253dq cs1253ez
cs125317 cs125353 cs125389 cs1253az cs1253ci cs1253dr cs1253fa
cs125318 cs125354 cs125390 cs1253ba cs1253cj cs1253ds cs1253fb
cs125319 cs125355 cs125391 cs1253bb cs1253ck cs1253dt cs1253fc
cs125320 cs125356 cs125392 cs1253bc cs1253cl cs1253du cs1253fd
cs125321 cs125357 cs125393 cs1253bd cs1253cm cs1253_dun cs1253fe
cs125322 cs125358 cs125394 cs1253be cs1253cn cs1253dv cs1253ff
cs125323 cs125359 cs125395 cs1253bf cs1253co cs1253dw cs1253fg
cs125324 cs125360 cs125396 cs1253bg cs1253cp cs1253dx cs1253fh
cs125325 cs125361 cs125397 cs1253bh cs1253cq cs1253dy cs1253fi
cs125326 cs125362 cs125398 cs1253bi cs1253cr cs1253dz cs1253fj
cs125327 cs125363 cs125399 cs1253bj cs1253cs cs1253ea cs1253fk
cs125328 cs125364 cs1253aa cs1253bk cs1253ct cs1253eb cs1253fl
cs125329 cs125365 cs1253ab cs1253bl cs1253cu cs1253ec cs1253fm
cs125330 cs125366 cs1253ac cs1253_bla cs1253cv cs1253ed cs1253_gui
cs125331 cs125367 cs1253ad cs1253bm cs1253cw cs1253ee cs1253_kai
cs125332 cs125368 cs1253ae cs1253bn cs1253cx cs1253ef include
cs125333 cs125369 cs1253af cs1253bo cs1253cy cs1253eg lib
cs125334 cs125370 cs1253ag cs1253bp cs1253cz cs1253eh
cs125335 cs125371 cs1253ah cs1253bq cs1253da cs1253ei
cs125336 cs125372 cs1253ai cs1253br cs1253db cs1253ej
cs1253al@classes:/classes/cs1253$ cd cs1253al
cs1253al@classes:~$
```

to get a list of basic commands at any time, simply type "help"

### How to Create/Compile/Save/Modify a Program (The First Assignment)

Now that you know how to enter and navigate the server, its time to learn how to create, compile, save, and modify a C++ program. To begin, lets navigate back to our home folders.



```
classes.csc.lsu.edu - PuTTY
login as: cs1253al
cs1253al@classes.csc.lsu.edu's password:
Linux classes.csc.lsu.edu 2.6.26-2-amd64 #1 SMP Tue Jan 12 22:12:20 UTC 2010 x86_64

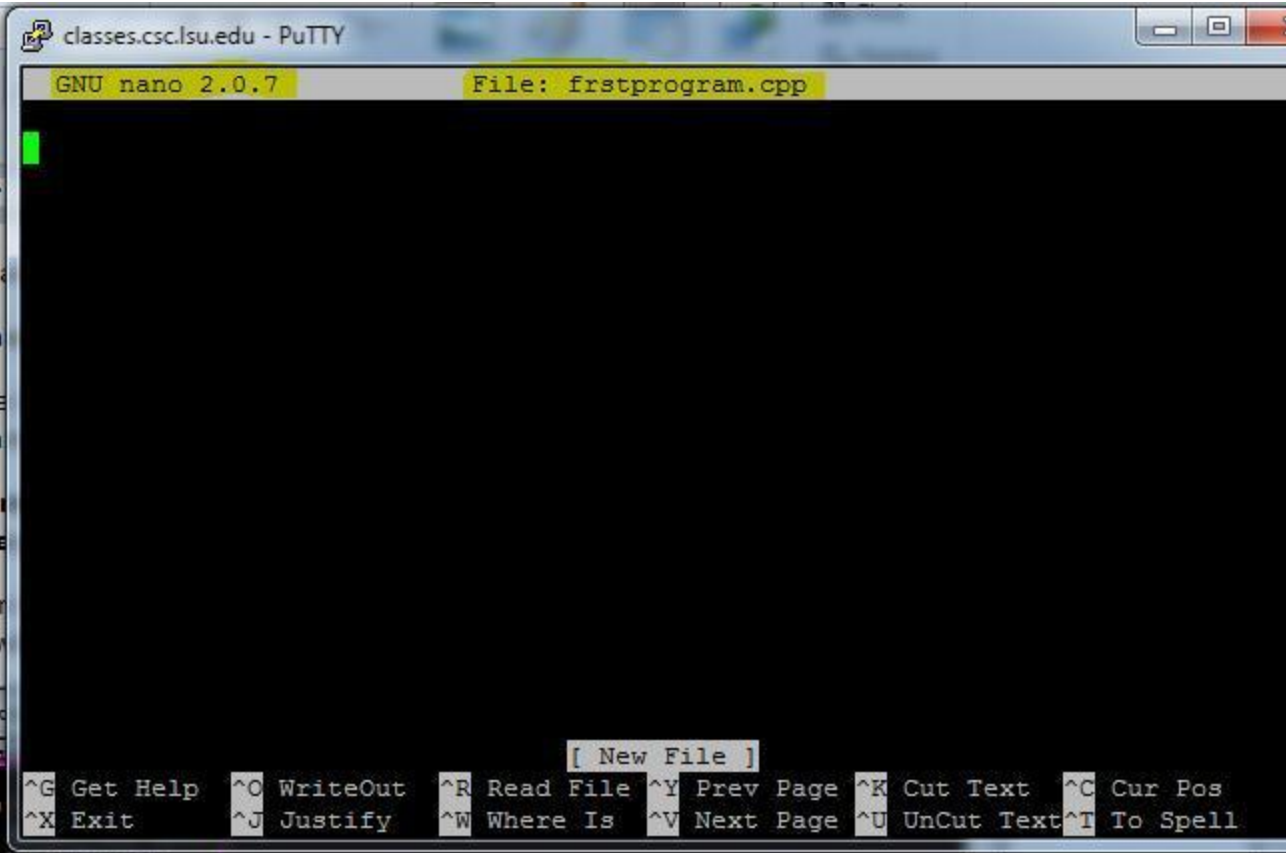
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Fri Aug 27 23:34:30 2010 from ip68-111-60-39.br.br.cox.net
cs1253al@classes:~$
```

Once you are in your home folder, you will need to open a text editor to begin coding. There should be a basic, easy to use text editor already built in to the server. This editor is called "pico" and can be evoked by typing:

```
pico (file name)
```

Replace the (file name) with what you want to name your program. I called mine "firstprogram.cpp". After hitting enter, you should see a screen like the one below.



```
classes.csc.lsu.edu - PuTTY
GNU nano 2.0.7 File: frstprogram.cpp
[ New File ]
^G Get Help  ^O WriteOut  ^R Read File  ^Y Prev Page  ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page  ^U UnCut Text ^T To Spell
```

Note: you must include the .cpp extension after every program name! If you do not do this, the program will not compile later! Now we can begin coding. Lets begin by typing the basic elements that (for practical purposes) will be needed in all of the programs we will be creating.

```

classes.csc.lsu.edu - PuTTY
GNU nano 2.0.7 File: firstprogram.cpp
//A Basic C++ Program
//Date: August 28, 2010
//Name: Nick Magazine
//#include "folder(s)"
int main()
{
return 0;
}
[ Read 10 lines ]
^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell

```

In pink: A header. It is a good habit to put a header at the start of every program. Remember that `"/"` denotes a comment. Comments are lines of text that are not read by the computer and therefore do not effect the code.

In green: The folders that will be imported. By importing certain folders, we gain access to more C++ functions. This can often save us the time of writing many lines of code to accomplish a redundant task. Because the program we will be creating is basic, this line will not be needed (this time anyway).

In red: The body and code container. Within this area is where we will be typing our code.

In blue: The return type. In this field we put a return value. Return values can be used in several different ways. In this case, the return value denotes whether or not the code executed properly. Returning 0 indicates to the computer that the code was executed without any errors. At this point, everything is set up and ready for us to begin typing our body of code. You can enter whatever you would like. Just make sure that your code follows the proper format. For my program, I made a simple currency converter. The converter will change Euros (€) to USD (\$). This is the code for the Euro to USD converter. When you make your program, try to integrate a couple of simple commands such as `"cin` and `"cout`". Reading and writing actual code is the best and usually easiest way to learn.

```

//A Basic C++ Program
//Date: August 28, 2010
//Name: Nick Magazine

#include "../include/std_lib_facilities.h"

int main()
{
    double var_euros = 0;
    double var_usd = 0;

    cout << "This Program converts Euros to USD.\n";

    do
    {
        cout << "Please enter the number of Euros: ";
        cin >> var_euros;
        var_usd = var_euros * 1.31;
        cout << var_usd << "USD\n";
        var_euros = 0;
    } while (var_euros = 0);
    return 0;
}

```

Now that you have typed your code, lets compile it. To save your work in pico, hit control+x

```

    var_euros = 0;
} while (var_euros = 0);
return 0;
)

```

^G	Get Help	^O	WriteOut	^R	Read File	^Y	Prev Page	^K	Cut Text	^C	Cur Pos
^X	Exit	^J	Justify	^W	Where Is	^V	Next Page	^U	UnCut Text	^T	To Spell

When pico asks if you would like to save, hit y to say yes.

```

cs1253al@classes:~$ firstprogram
This Program converts Euros to USD.
Please enter the number of Euros: 8
10.48USD

```

At this point, pico may or may not ask you if you want to change the files name. you can if you want, but you don't have to.