

CSCI 274 - Intro to Linux OS

Week 2 - Interacting with the Command Line and ~/.bashrc Configuration File

Max Gawason (maxgawason@mines.edu) [A]
Rocco Marchitto (rmarchitto@mines.edu) [B/C/D]

WALK-IN ADVISING

Have a quick question? Walk-in advising is now available!

Who: Colleen Maroney, CS Academic Advisor

When: Tuesdays 2pm- 4pm
Wednesdays 3pm – 5pm

Where: CTLM 246F

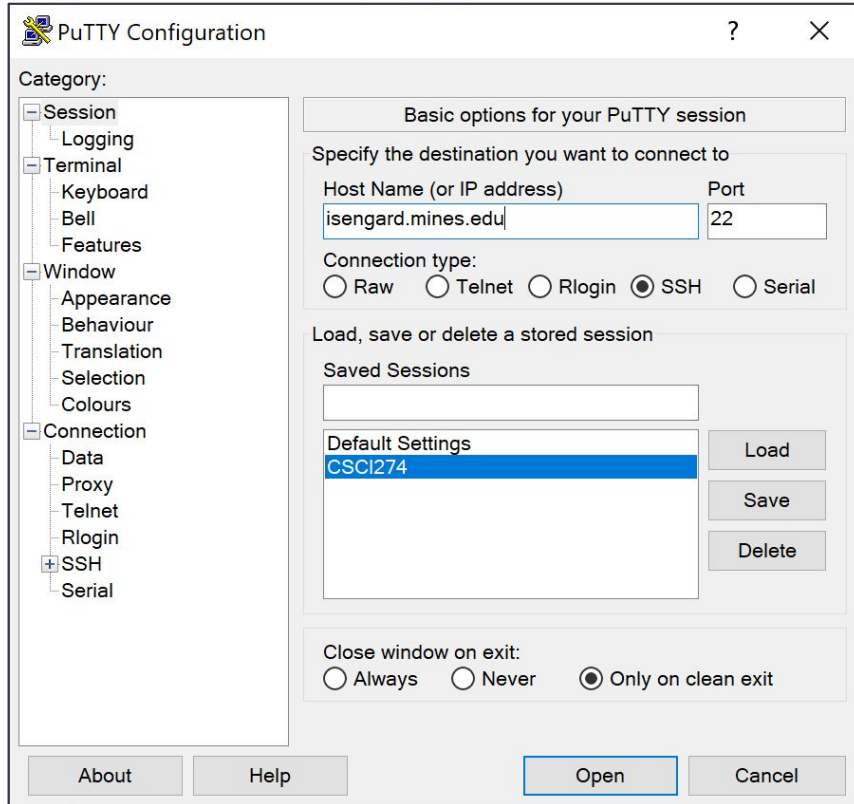


For a longer conversation please make an appointment with Colleen. Check your inbox for an email from her with the appointment link and additional advising resources. <https://casaappointments.as.me/colleen>

Overview

1. CHEATSHEET.LNX
2. Interacting with the Command Line
3. Understanding the meaning of `~`, `.`, `..`
4. `.bashrc`
5. Choose an Editor Assignment

PuTTY



Is an SSH and telnet client. Can be installed on most operating systems.

Opens a terminal on **isengard.mines.edu** server. Login via campus wide username and password.

Terminal emulator is a program that opens a window and lets you interact with the shell.

CHEATSHEET.LNX

Notes from the semester that students can use on the Midterm and Final exam.
Must be written in a Linux environment using a text editor of choice.

- Will submit for review twice throughout the semester
- You **can not** share/distribute; Collaboration Policy applies

Sample

```
Vim/Vi techniques {
  1. delete blank all blank lines [ :g/^$/d
  2. delete from the current cursor position to the end of line [
      a. enter Visual Mode via Ctrl-V
      b. do: Shift+d
         OR
      a. d$
    ]
  3. change a word, globally [ :%s/\<word>/newWord/g ]
File Navigation {
  * $ man <softwareName> = short for manual. It helps gets the manual pages for dif
  * $ apropos <keyword> = searches through all the command lines tools and prints
  * $ clear = makes the page blank
  * $ script <fileName> = captures everything in the terminal into a text file. It
  * $ pwd = present working directory; tells you where you are in the directories.
  * $ ls = list; list everything in the directory that you are currently in.
  * $ ls -l = shows a long list of everything in the directoty that you are current
  * $ ls -a = will show all files including hidden files.
  * $ ls <file/path> = will list the files and directories that exist at that path
  * $ cd <directroyName> = allows you to navigate to a different directory
  * $ cd .. = allows you to go to the parent directory
  * $ cd <file/path/name> = will take you the file name directory
  * $ cd = will take you back to your home directory.
  * $ cd ~/<directoryName = will brings you to the directory you want in the home d
  1. How to View Files {Using a text editor to view files is not the best option.}
      * $ less <fileName> = allows you to view the file. Most common way to view big fi
      * $ less is a program that was made to view files similar to the program "more".
  2. Information about a file
      * $ file simple.cpp = this will tell you information about the file
      * $ file exe = compiled c++ code. talking about how it is an exectuable.
      * You are able to run this commmand on a directory and it will give
      * $ wc - way to count things in a file
      * $ wc -l = counts the number of new lines that are in the file
      * SIDE NOTE: you can get vim to show you line number instead of the "-" character
      * $ wc -w = counts the number of words in a file
Common Pipeline Commands {
  1. read a file [ $ cat fileName ]
      * 'cat file' and 'cat < file' is the same thing
  2. tr { Automatically translates (subsitutes, or maps) one set of characters to another. It co
      input to the standard output with substitution or deletion of selected characters. }
      * FORMAT -> tr [Options] set1 [set2]
      * options list {
          a. '-d' = deletes chracters in set1 from the input
```

Tip

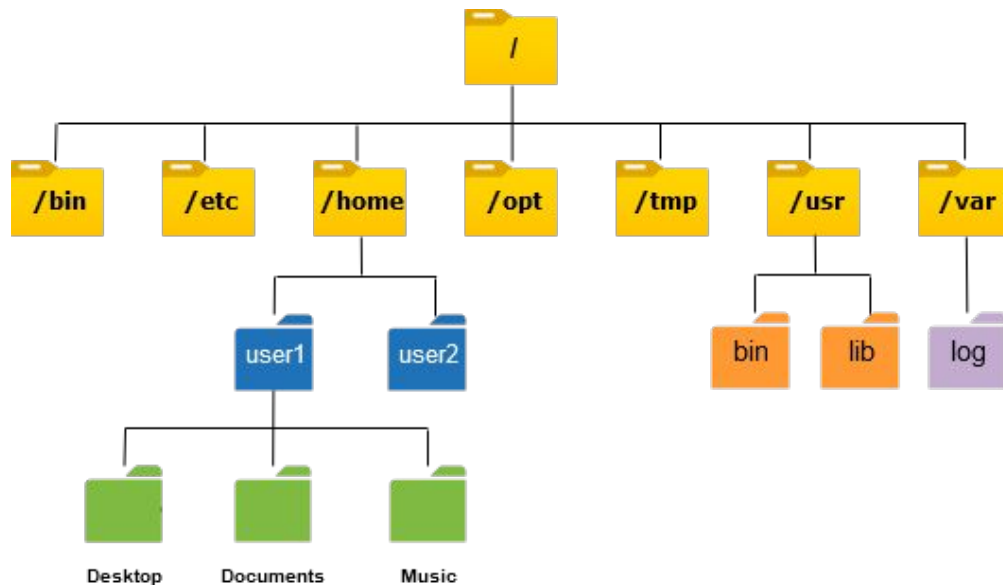
`script` is a UNIX command-line application that records a terminal session.

It stores the output as text file in the current directory and the default filename `typescript`.

```
[dwade@isengard ~]$ script
Script started, file is typescript
[dwade@isengard ~]$ exit
exit
Script done, file is typescript
[dwade@isengard ~]$
```

The Linux Filesystem

- File hierarchy similar to Windows and Mac
- Everything on Linux is a file including devices and pipes
- `/` = the root directory
 - Contains all files



Anatomy of a Linux Command

command [OPTIONS] [PARAMETERS]

- command = the name of the command
- options = modifies the behavior of the command
 - Either a single dash with a letter or a double dash with a word or phrase
- parameters = the arguments passed to the command

Interacting with Command line

man - used to display the user **man**ual of any command that we can run on the terminal

```
$ man [ -f | -w ] [COMMAND NAME]
```

pwd - **P**rint **W**orking **D**irectory

```
$ pwd [ -L | -P ]
```

ls - lists directory contents of files and directories

```
$ ls [ -t | -1 | -l | -h | -a ]
```

Understanding the meaning of [~], ., ..

[~] = The tilde is a Linux "shortcut" to denote a user's home directory

. = denotes the current directory

.. = denotes the “parent” directory

Interacting with Command line

mkdir - allows the user to **make** **directories**

```
$ mkdir [ -v | -p ] [DIRECTORY]..
```

rmdir - used **remove** empty **directories** from the filesystem

```
$ rmdir [ -p | -v ] [DIRECTORY]..
```

cd - used to **C**hange current working **D**irectory

```
$ cd [DIRECTORY]
```

Interacting with Command line

mv - used to **move** one or more files or directories

```
$ mv [-i | -f | -n | -b ] [SOURCE] [DESTINATION]
```

cp - used to **copy** files or group of files or directory

```
$ cp [-R | -i ] [SOURCE] [DESTINATION]
```

less - used to read contents of text file one page(one screen) per time

```
$ less [FILENAME]
```

apropos - helps the user when they don't remember the exact command

```
$ apropos [KEYWORD]..
```

.bashrc

Reminder: shell = an interpreter which can accept commands from the user and run them to perform operations

The **.bashrc** file is a shell script which is run every time a user opens a new shell. It is a good place to put commands you want to run every time you open a shell.

Typically, it contains configurations for the terminal sessions such as:

coloring	completion
shell history	command aliasing
And more	

Choose an Editor Assignment

=====
= Assignment - Choose an Editor =
=====

Description

=====

In class, we briefly introduced the vim and emacs editors. Now, you've got to pick (at least) one to get familiar with.

You will complete a Canvas quiz and answer the following questions:

1. Which editor you have chosen for the course
2. For an explanation of why you chose that editor
3. For a link to a "cheatsheet" you will use to help learn the editor throughout the course

The editor you choose to learn in this course must have all of the following features:

1. It must be able to run (with all of these features) in a terminal (or "console") window, which is to say "non-GUI mode."
2. It must support syntax highlighting.
3. It must support search and replace based on regular expressions.

Optionally, the editor you choose may have the following features:

1. You may be able to "script" or automate the editor.

Due Dates

- Quiz 2 due Sept 4 at 11:59pm
- Choose an Editor Assignment due Sept 4 at 11:59pm