Pittsburgh Supercomputing Center Application and UNIX Tutorial

By: Jimmy Scott

Advisor: Dr. Chuan Li

TABLE OF CONTENTS

Tutorial for Getting Allocation at PSC

This will be for getting an education allocation, which can be used for classes.

Step 1) Get an XSEDE account.

- You can create an account by going to XSEDE.org
- From there, click create account where it says XSEDE User Portal
- Once you fill out all the necessary information, you have your XSEDE account.

Step 2) Apply for allocation.

- To apply for allocation, click the ALLOCATIONS tab that should be at the top of your XSEDE USER PORTAL page.
- Click on Submit/Review Request
- Once here, you will see there are multiple options for the types of allocation you can apply for.
- If you click on Educational – Start a New Submission it will show you a page which highlights the documentation needed to apply for the allocation.
- The list will look like the following:
    o New Educational requests will require:
    o
        ▪ Title
        ▪ Abstract
        ▪ Keywords
        ▪ Principal investigator
        ▪ Field of science
        ▪ Documents (Resource Justification, Syllabus, PI CV (2 pages limit))
        ▪ Resources (Grants/funding)
            ● Typbically they use the NSF format for this but it is not limited to NSF.
- Once these documents are submitted, they will review it in a reasonable amount of time (about a week) and pending acceptance, you will have your allocation available.

MENTIONABLES:

- You do not need to be the PI to apply for the allocation, however; you will need a PI in order for the allocation to be approved.
- Co-PI's can be assigned and given almost equal access to the allocation as the PI.
    o This could be helpful if the PI would want someone else handling the allocation.
- In order for students to gain access to the allocation, they too will need an account.
    o From this, they can be given access to the allocation by the PI or the Co-PI

- Any other information about the allocations can be found here https://portal.xsede.org/allocations-overview. Or you can contact PSC directly by emailing your questions/concerns.
    o The email is as follows: grants@psc.edu.
- For more of a general overview, you can visit the PSC website as well, www.psc.edu.

Login to bridges server using UNIX

In order to login to the Servers that we have allocated on the PSC, there are specific commands we must write in.

```
jimmy@jimmy-SVS1511AGXB:~$ ssh -l jscott login.xsede.org
Please login to this system using your XSEDE username and password:
password:
```

This is the command that we will use to login to your account in UNIX. The password will appear blank when you type it in. After you have logged onto your XSEDE account, you are ready to connect to your allocated server. For this example, we are using the bridges server. After you've successfully logged in, type:

```
[jscott@ssohub ~]$ gsissh bridges
```

As you can see, we are accessing the Bridges server from my account, jscott, from the general ssohub server they have. We specify the server we would like to go to by typing in the command gsissh, followed by the server we would wish to go to. If you have successfully logged in, your screen should look like this:

```
[jscott@ssohub ~]$ gsissh bridges
Last login: Tue Nov  1 19:42:54 2016 from ssohub.iu.xsede.org
******************************** W A R N I N G ********************************
You have connected to br006.pvt.bridges.psc.edu

This computing resource is the property of the Pittsburgh Supercomputing Center.

It is for authorized use only.  By using this system, all users acknowledge
notice of, and agree to comply with, PSC polices including the Resource Use
Policy, available at http://www.psc.edu/index.php/policies. Unauthorized or
improper use of this system may result in administrative disciplinary action,
civil charges/criminal penalties, and/or other sanctions as set forth in PSC
policies. By continuing to use this system you indicate your awareness of and
consent to these terms and conditions of use.

LOG OFF IMMEDIATELY if you do not agree to the conditions stated in this warning


Please contact remarks@psc.edu. with any comments/concerns.

******************************** W A R N I N G ********************************
[jscott4@br006 ~]$
```

From this step forth, we are able to access modules that the server provides and other neat features. To leave the server, simply type "exit", or press ctrl+d.
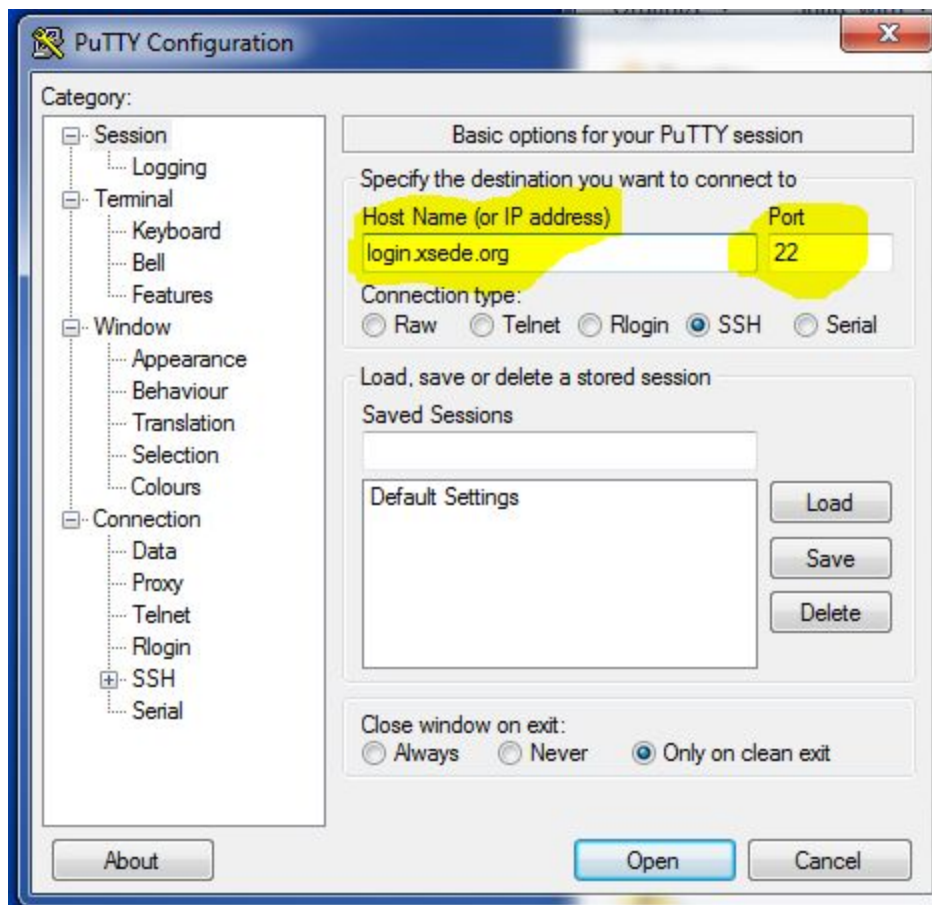
https://portal.xsede.org/web/xup/single-sign-on-hub

Logging into XSEDE using PUTTY
(For WINDOWS users)

What you need installed:
    PUTTY.exe
    Filezilla

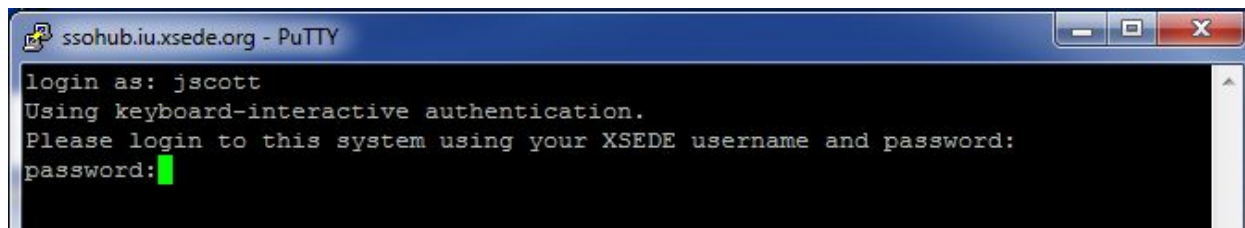Both of thesse programs are free to download from any computer with internet access.

1) First you will want to run putty.exe on your computer. Once run, you will get a window that looks like the following.

The important pieces to recognize are the highlighted areas. The Host Name is going to be filled in with the login webside for XSEDE. This login cite is login.xsede.org.
**Keep note of the Port number as that will be needed for Filezilla. It should automatically fill in.

After all the proper information is applied to the PuTTY Configuration, click Open.

You should get a terminal-looking window that asks for your login information.

At first, it will prompt you with

      login as:

All this prompt is asking for is for your XSEDE username that you have previously created. It will then ask for your password. NOTE: Your password will not show up as you type it in the terminal. Make sure you enter your password correctly and then hit Enter and you should be logged into the general server for XSEDE.
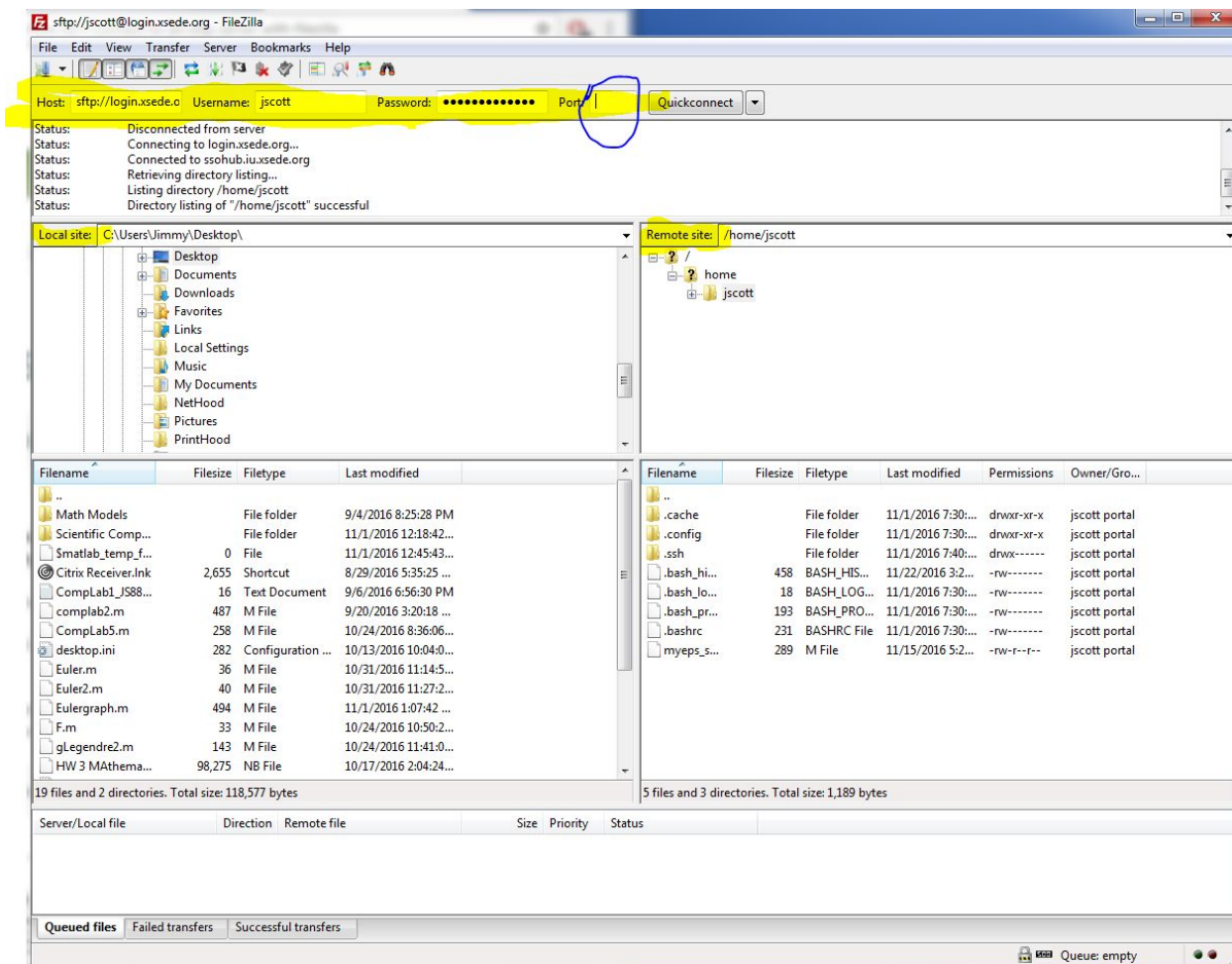
If you are experiencing difficulty logging in, it may be due to the security of the WIFI you are on. Make sure the wifi you are connected to is secure and allows you access to the servers. For example, I've had users have difficulty logging on when using the guest network for WCU.

From here on, the steps are all the same for logging onto any other server for XSEDE.

Using Secure File Transfer Protocal (SFTP)

For this, we are actually going to use Filezilla, so make sure you have the program opened.

Here we are going to make sure Filezilla is set up properly.

All the highlighted areas are of importance.

First off, you're going to enter your Host, username, password and port number.

If you don't know your port number, to find it, you're going to use the command, netstat.

Now this command will likely output a lot of information, but we are only concerned with the first line of output. Underneath Local Adress, we see the ssohub.iu.xsede.org:22. Then 22 would be our port number.

Information to be input into Filezilla will be as follows.

Host: login.xsede.org
Username: <your username>
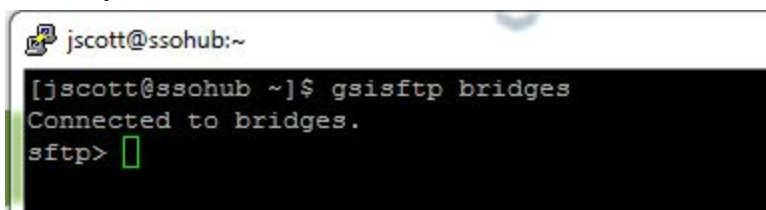Password: <your password>
Port: <The port you found using the netstat command>

If all this information is entered correctly, then you'll be able to see the files you have saved on the remote server. Notice that you have a local site, which is your computer, and a remote site, which is the server.

Now we are ready to transfer files from your computer to the remote server. Make sure you are on the general XSEDE server in your terminal.

From the general server terminal, you can type the following command, along with what server you would like to transfer your files to.
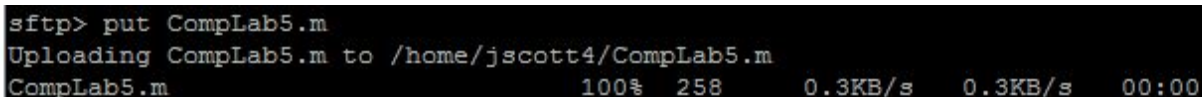

We include gsi at the beginning of sftp to make sure that it is secure. Now we are ready to transfer files onto the bridges server. To do so, we go to Filezilla, and underneath where it says Local site, we access the folder in which our file is located. Underneath that box, we can see another box that will have the files within the folder you just selected. Click and drag the file you wish to transfer over to the bottom right box which is your remote server. After a few moments, the file is then transferred. To complete the transfer, you will need to use the following command.

For whatever file you chose to transfer, you will use the put command, along with the filename in order to transfer over to the server. Congratulations, you now have successfully transferred your file to the XSEDE server.

Basic Unix Commands

ls (list)

When you first login, your current working directory is your home directory. Your home directory has the same name as your user-name.
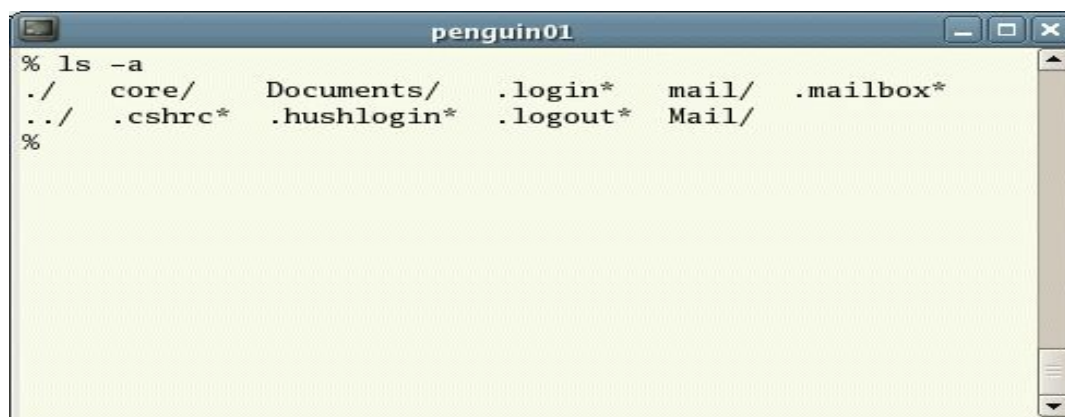
To find out what is in your home directory, type

% ls

Typing in lowercase LS will only show you the files and subdirectories that are not hidden. To view the hidden files, you'd have to type
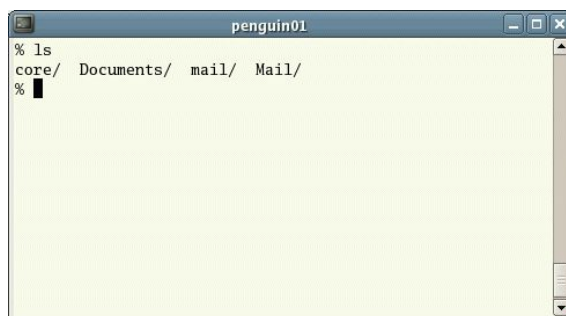
% ls -a

```
penguin01                                                    _ □ ×
% ls -a
./    core/     Documents/    .login*    mail/    .mailbox*
../   .cshrc*   .hushlogin*   .logout*   Mail/
%
```

ls is an example of a command which can take options: **-a** is an example of an option. The options change the behaviour of the command. (Note that I will talk about options later in the tutorial). As you can see, the option -a was part of the listed options that are seen when I entered the command man ls.

However, before even considering modifying any of the hidden files, you should make sure you know exactly what you're doing in Unix.

```
penguin01                                  _ □ ×
% ls
core/  Documents/  mail/  Mail/
% █
```

If you don't see any files, it may just mean that there are no files there to begin with.

man (Manual)

When using the man function, it allows you to see the options that go along with a command. For example,

man ls

This command will give you the following screen.

```
LS(1)                          User Commands                          LS(1)

NAME
       ls - list directory contents

SYNOPSIS
       ls [OPTION]... [FILE]...

DESCRIPTION
       List  information  about  the FILEs (the current directory by default).
       Sort entries alphabetically if none of -cftuvSUX nor --sort  is  speci-
       fied.

       Mandatory  arguments  to  long  options are mandatory for short options
       too.

       -a, --all
              do not ignore entries starting with .

       -A, --almost-all
              do not list implied . and ..

       --author
              with -l, print the author of each file

       -b, --escape
              print C-style escapes for nongraphic characters

       --block-size=SIZE
              scale sizes by SIZE before printing them; e.g., '--block-size=M'
              prints sizes in units of 1,048,576 bytes; see SIZE format below

       -B, --ignore-backups
              do not list implied entries ending with ~

       -c     with -lt: sort by, and show, ctime (time of last modification of
              file status information); with -l: show ctime and sort by  name;
              otherwise: sort by ctime, newest first

       -C     list entries by columns

       --color[=WHEN]
              colorize  the output; WHEN can be 'always' (default if omitted),
              'auto', or 'never'; more info below

       -d, --directory
              list directories themselves, not their contents
 Manual page ls(1) line 1 (press h for help or q to quit)
```

(This shows just some of the options that the list command can do)

Now you can see all the options that go along with your ls, or list command for this example.

mkdir (make directory)

We will now make a subdirectory in your home directory to hold the files you will be creating and using in the course of this tutorial. To make a subdirectory called unixstuff in your current working directory type

<div align="center">mkdir unixstuff</div>

To see the directory you have just created, type

<div align="center">ls</div>

rmdir (remove directory)

This does the exact opposite of what mkdir does. It removes the directory. Note: if you try to remove a directory with files within it, the directory will not be removed. The directory you wish to empty must be empty itself in order to remove, unless another option is used with rmdir.

<div align="center">rmdir unixstuff</div>

You can check your removed directory by seeing if it actually was removed using the list command.

cd (change directory)

The command **cd directory** means change the current working directory to *'directory'*. The current working directory may be thought of as the directory you are in, i.e. your current position in the file-system tree. Typing % cd by itself will take you back to your home directory. This is handy when you forget where you came from in the directories.

To change to the directory you have just made, type

<div align="center">cd unixstuff</div>

Type **ls** to see the contents (which should be empty)

Note you do not have to go through one directory at a time in the command line. If you wish to bypass typing the same command to get to a directory, you can simply type the specific file location, so long as that location is within the starting directory.

<div align="center">cd unixstuff/Stuff/morestuff/stuffmore</div>

This is gaining an understanding of the pathnames. If you were to type

<div align="center">cd stuffmore</div>

you would get the output:

cd: stuff: No such file or directory

It would show that no such file or directory exists. This is because you need to be in the current directory in order to access a file like that.

In the home directory, type

<div align="center">**ls -a**</div>

As you can see, in the home directory(and in all other directories), there are two special directories called (.) and (..)

The current directory (.)

In UNIX, (.) means the current directory, so typing

<div align="center">**cd .**</div>

means stay where you are (the unixstuff directory).

This may not seem very useful at first, but using (.) as the name of the current directory will save a lot of typing, as we shall see later in the tutorial.

The parent directory (..)

(..) means the parent of the current directory, so typing

<div align="center">**cd ..**</div>

will take you one directory up the hierarchy (back to your home directory). Try it now.

Note:Desktop typing **cd** with no argument always returns you to your home directory. This is very useful if you are lost in the file system.

This next example is directly from the website. The visual was too good not to use as part of this example.
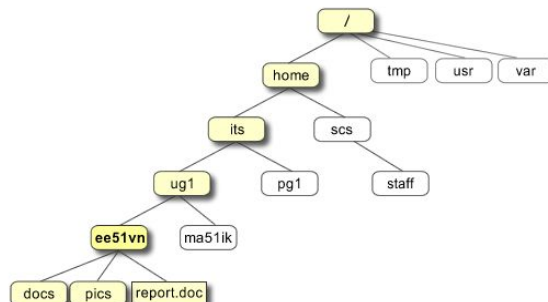
pwd (print working directory)

Pathnames enable you to work out where you are in relation to the whole file-system. For example, to find out the absolute pathname of your home-directory, type **cd** to get back to your home-directory and then type

<div align="center">**pwd**</div>

The full pathname will look something like this -

/home/its/ug1/ee51vn

which means that **ee51vn** (your home directory) is in the sub-directory **ug1** (the group directory),which in turn is located in the **its** sub-directory, which is in the **home** sub-directory, which is in the top-level root directory called " ⁄ " .

~ (your home directory)

Home directories can also be referred to by the tilde ~ character. It can be used to specify paths starting at your home directory. So typing

<p style="text-align:center">ls ~/unixstuff</p>

will list the contents of your unixstuff directory, no matter where you currently are in the file system.

cp (copy)

Now lets say you want to copy a file and rename it something else or place it in a different directory. The cp command does that for you.

```
jimmy@jimmy-SVS1511AGXB:~$ cp Desktop/UGH .
jimmy@jimmy-SVS1511AGXB:~$ ls
acroread-bin_9.5.5-1raring1_i386.deb  Pictures
Desktop                               Public
Documents                             Resource Justification.odt
Downloads                             Resource Justification.pdf.odt
examples.desktop                      Templates
Living Situation.ods                  UGH
Music                                 Videos
jimmy@jimmy-SVS1511AGXB:~$ 
```

In this example, I took some arbitrary file called UGH from my Desktop, and placed it in the directory I was currently in. Now there is also a way to rename a file within the same directory.

```
jimmy@jimmy-SVS1511AGXB:~/Desktop$ cp UGH OOH
jimmy@jimmy-SVS1511AGXB:~/Desktop$ ls
OOH   OOO   UGH
jimmy@jimmy-SVS1511AGXB:~/Desktop$ 
```

As we see here, we have within our current directory the file UGH. I copied it and called it OOH and now I have a file called OOH and UGH in my current directory.

mv (move)

A very similar command to copy, is the move command. The move command allows you to move a file to a particular destination rather than copy and paste it to a new place. This can be good for organization if you're looking to file particular files away in a tedious-to-get-to location.

Here we can see what I did. Originally, my home directory did not have the file OOH in it. So I moved OOH from my desktop to my current directory (which was my home directory). As we can see, the file no longer exists on the Desktop, and is not moved to my home directory.

rm (remove)

Now I have no need for some of these files in my Desktop directory. So what we can use is the remove command. It works just like it sounds by removing a file from your directory.

```
jimmy@jimmy-SVS1511AGXB:~/Desktop$ ls
OOO   UGH
jimmy@jimmy-SVS1511AGXB:~/Desktop$ rm OOO UGH
jimmy@jimmy-SVS1511AGXB:~/Desktop$ ls
jimmy@jimmy-SVS1511AGXB:~/Desktop$ █
```

We see I had my desktop with two files, OOO and UGH in it, then I removed them from the directory.

Similarly, we have **rmdir** which will remove a directory, but only if said directory is empty. It is ill-advised to remove a directory with anything in it anyways since there could be important files (which includes hidden files/directories) within any given directory.

cat (concatenate), less, head, and tail

We can use the cat, or concatenate command, to display the contents of a file on screen. less displays the contents of a file a page at a time. The head command can be used to look at the first few lines, and tail, the last few lines of a file.

```
jimmy@jimmy-SVS1511AGXB:~$ cat UGH
This is a note

with two lines.
jimmy@jimmy-SVS1511AGXB:~$ head -1 UGH
This is a note
jimmy@jimmy-SVS1511AGXB:~$ tail -1 UGH
with two lines.
jimmy@jimmy-SVS1511AGXB:~$ █
```

The command less, is a bit more involved. You can search for words within the file while using less by using /word.

```
This is a note

with two lines.
/is█
```

This will highlight the first instance of the word "is". You can move to the next instance by pressing n.

Similarly, the command grep allows you to search for specific words or patterns in a file. Grep will print out the every line that has the word in it.

```
jimmy@jimmy-SVS1511AGXB:~$ grep This UGH
This is a note
jimmy@jimmy-SVS1511AGXB:~$ grep -i this UGH
This is a note
jimmy@jimmy-SVS1511AGXB:~$
```

As you can see, I used the same command, but used an option with one of them. The -i means ignore, which ignores uppercase or lowercase within the word you are searching. Feel free to look at the manual for grep.