

Basic Terms:

Directory: A folder on your computer that contains files

Root: The highest-level directory on the computer (denoted '/')

Filepath: The "address" to a file or directory on your computer

Forward slashes divide levels in the nested hierarchy of directories:

/top_level_directory/second_level_directory

Standard output: Where the results of a program are written to, i.e., what you see.

Basic commands:

Commands are typed at the prompt, and executed by hitting 'enter.' Modifying text typed after the command is called an 'argument.' Many commands require at least one argument to specify input. See below for examples.

Getting around:

cd: change directory

mkdir: make a directory

cp: copy a file to a location (creates a second copy of the file)

mv: move a file to a location (does not create a copy; the original file will be absent from its original location - will also overwrite files with the same name, so be careful)

ls: list files within a directory

rm: delete a file

pwd: get your working directory

man: get the manual for a command

echo: print output to screen

Text editing:

grep: print any line that contains text which matches input text

cat: concatenate input and write to standard output (screen)

head/tail: print first or last lines to file

wc: print number of lines, words, and bytes in a file.

Common arguments, tips and tricks:

-r: Recursive. Can be combined with many commands. For example, "rm -r" allows for the removal of directories.

-l: Long. Gives more information. For example, "ls -l" shows more information about files in a folder than does ls, which simply lists them.

*****: A wildcard. Matches characters in given text. "ls *.csv" shows all .csv files in a directory.

|: Pipe. Send output of one command to another as input.

"ls *.txt | wc -l" : counts number of files in directory ending in '.txt'

tab key: Autocomplete. Once enough unique characters are entered, hit tab to finish the text.

Up arrow: Last command. Hit the up arrow to repeat the last command you typed.

Ctrl + C: Escape. Pressing Ctrl + C will terminate a Unix process.

> and >>: Redirect will put your output in a file. > will overwrite the file; >> will add output to the end of the file. "cat file1.txt >> file2.txt" will copy file1.txt to a new file, file2.txt, or append file1.txt's content to file2.txt if it already exists in that directory.

