Basic Linux (Unix) Commands
D.H.McIntosh 02/25/06
modified by G. H. Rudnick on 8/25/24
any item contained within angled brackets, e.g. <somefile> indicates that this is a placeholder for some file. The name can be anything you want.

commands are all written in this font and will be colored red

Any line preceded by a # is a comment

There are some commands that are followed by a "." this is important and means whatever directory you are in now.

Remember that all linux commands and filenames are case sensitive. That means that LS is not the same as ls

#===
Things you can do with directories:
when you login into one of the computer lab machines you start at the directory
/home/<username> . The starting directory for Macs is /Users/<username>
mkdir NewDir - makes new directory (i.e., folder) /home/username/NewDir/
rmdir NewDir - removes directory NewDir/ . This only works in an empty directory
cd NewDir - change to NewDir
cd .. - go back ("up") one dir.
cd ../Dir - go back ("up") one dir and then down into Dir
cd - return to home dir.
cd ~/ - return to home dir
cd - - return to previous directory
pwd - shows current dir.
ls - list contents of dir.

single letters preceded by a "-" can be used to specify options.
ls -l - list contents of the directory with a verbose listing
ls -lt - list contents of the directory with a verbose listing, ordering the list in order of most recently modified to oldest file

moving, copying, and removing files. This works for all files
cp SomeFile NewFile - copies file to another file
cp SomeFile NewDir/ - copies file to new dir.
cp Dir/SomeFile . - copies file in Dir to current dir. (.)
cp ../SomeFile . - copies file in one dir. up to current dir. (.)
mv SomeFile NewDir/ - moves file to new dir.
mv SomeFile ../../ - moves file up 2 dirs.
mv *.txt NewDir/ - moves all files ending in .txt to new dir.
rm SomeFile - deletes file

`rm *` - deletes ALL files in dir. (Careful!)

This works only on ascii (data) and text files. Do not use these on other file types, e.g. pdf.
`wc -l SomeFile` - how many lines in file
`more SomeFile` - scroll thru file, carriage return for 1 line at a time, space bar for many lines at a time
`less SomeFile` - another version of “more” but with different options and more versatile scrolling.

`tail -9 SomeFile` - shows last 9 lines of file
`head -9 SomeFile` - shows first 9 lines of file
`grep string File` - finds every occurance of "string" in File

executable files: ones that do something
`which ExecFile` - shows dir. of known executable files
`chmod a+x code.py` - required to make a code executable (necessary only once)
`./code.py` - running an executable like a python code

using tar: for making archives of many related files
`tar cvf TarFile.tar .` - copies entire contents of current dir. (.) into the tar archive named TarFile.tar
`tar cvf TarFile.tar File1 File3 File11` - puts 3 files in an archive
`tar xvf TarFile.tar` - unpacks tar archive
`tar xvfz TarFile.tar` - unpacks and unzips gzipped tar archive

misc.
`gzip SomeFile` - compresses file, becomes SomeFile.gz
`ungzip SomeFile` - uncompresses file
`man <command>` - gives the manual page for that command

external machines
`ssh -X User@Machine` - login (secure shell) to another machine, will be prompted for your password; -X means you can display files to local monitor
`rsync -u -a -v File User@Machine:~/Dir/` - copy File TO Dir on another machine. This will only copy the file if the version in your current machine (the source) is newer than the one on the destination maching.
`rsync -u -a -v Dir/ User@Machine:~/` - copy the contents of an entire Dir TO the home dir. on another machine
`rsync -u -a -v Dir/ User@Machine:~/` - copy a directory **and its contents** TO the home dir. on another machine
`rsync -u -a -v User@Machine:~/Dir/File .` - copy File FROM another machine to current machine only if the file on the other machine is newer than on your machine.

MANY more commands
see http://www.ss64.com/bash/
or use www.google.com to search e.g. "basic linux commands"