Type out definitions for the following commands and shortcuts in your own words:
a. pwd – This command stands for print working directory. With this command, you are able to determine the current directory you are in.
b. mkdir – Short for make directory, this command enables you to make a directory or folder.
c. cd – stands for change directory, you use this command to change to another defined directory.
d. cd .. – Takes you to the previous directory one directory at a time.
e. ls – short for list; used to list the contents of a directory
f. rm and the -r flag – short for "remove"; rm by itself is used to remove a file, but with the -r flag you can remove/delete a directory. It's important to note that this action is not reversible.
g. cat – This command has multiple uses. You can use it to display the contents of a file **(cat file1.txt)**, combine two files (fileA.txt and fileB.txt) to make fileC.txt that contains the content of fileA.txt and fileB.txt **(cat fileA.txt fileB.txt > fileC.txt)**, create a file and add content to it all in the same command line **(cat > test_file.txt "Write content of the file")**, and also print out the contents of a file including the line numbers **(cat -n test_file.txt)**.
h. head – displays the **first** 10 lines of a file
i. tail – displays the **last** 10 lines of a file
j. scp – stands for secure copy; used to copy a file or directory into another location
k. nano (including Ctrl+o and Ctrl+x) – nano opens up a file to write content in. Crtl+o is referred to as "write out" and it saves the current content of the file. Crtl + x allows you to exit the file.
l. --help – used with commands to display a summary of what a certain command does (e.g. ls --help)
m. TAB – helpful key command to use to complete a command instead of having to write it out. For example, if I want to go to a directory named "Fall_2024_materials" I can just type "cd Fall_" then hit tab and then the path to the directory is autocompleted to "cd Fall_2024_materials/"
n. Ctrl+a – moves the cursor to the beginning of the current command line. Helpful because you can't use your mouse to click to the beginning of the command line.
o. Ctrl+e – does the opposite of crtl + a; moves the cursor to the end of the current command line.
p. Ctrl+r – used to start a reverse search through your command history. For example, if I do crtl + r "scp" an example command that would show up is "scp /mnt/c/users/anmol/downloads/1a6m.pdb $HOME/" or whatever is the most recent command you used containing that contains what you're searching for.
q. Ctrl+k – deletes everything from the cursor position to the end of the command line; similar to doing crtl + delete
r. Ctrl+u – deletes everything to the left of the cursor position, it's like doing crtl + backspace
s. Ctrl+l – clears the commands you written above the current line so all you see after "crtl + l" is just one command line to write in.

What command would you use to navigate to your Desktop from /Users/ using an absolute path? Relative path?
Absolute path: cd /mnt/c/users/anmols/Desktop
Relative path (if I'm starting from /mnt/c/users/anmols/Downloads): cd ../Desktop

How would you copy /Desktop/Example Folder/ with multiple documents inside to /Documents/?
scp -r /Desktop/Example_Folder /Documents

If you didn't know which folder you were in, how would you navigate back to /Documents/?
cd /mnt/c/users/anmols/Documents

Fill in the blank:

- To push your local changes to GitHub, use the following sequence of commands:
    1. **git status** to view any unsaved changes.
    2. **git add** to save all files, or to save a specific file/folder.
    3. **git commit -m "your message"** to commit files for saving. Use '-m' to include a message.
    4. **git push** to push your changes to GitHub.

Do it yourself!

- Use commands in Bash to add the completed homework file to your week3_bash_github folder in your local qbio_490_name repository
- Use GitHub to stage, commit, and upload your completed Bash and Github HW into your personal GitHub repo

Turn in your answers for this assignment by attaching a link to your personal GitHub repo on Brightspace for full credit.