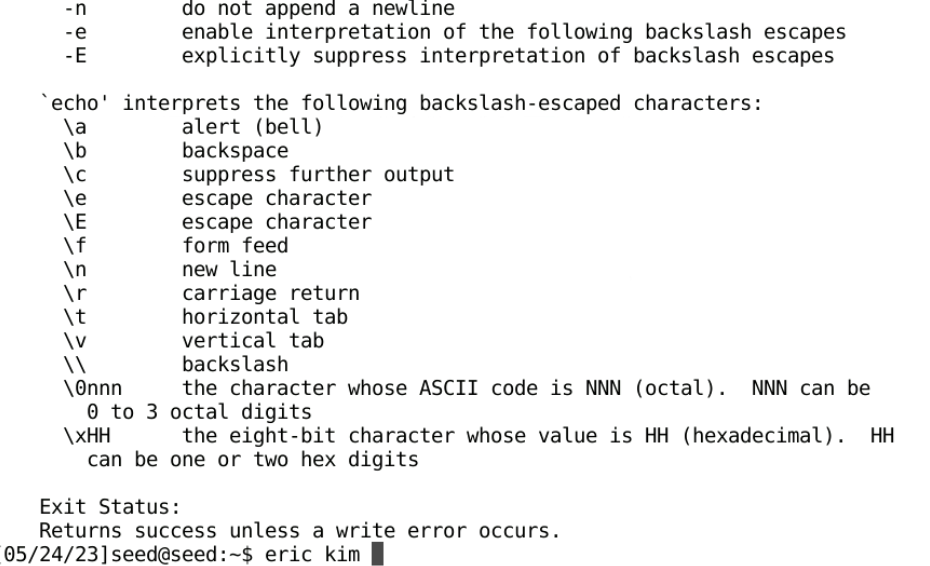
Laboratory Exercise 1-1 Bash Basics

The purpose of this lab introduces the Linux Command Line Interface and how to do basic navigation inside of the terminal. CLI, the interactive shell allows the user to interact and gives commands directly to the Operating Systems. In most default modern linux distributions, we would be using the default bash shell.

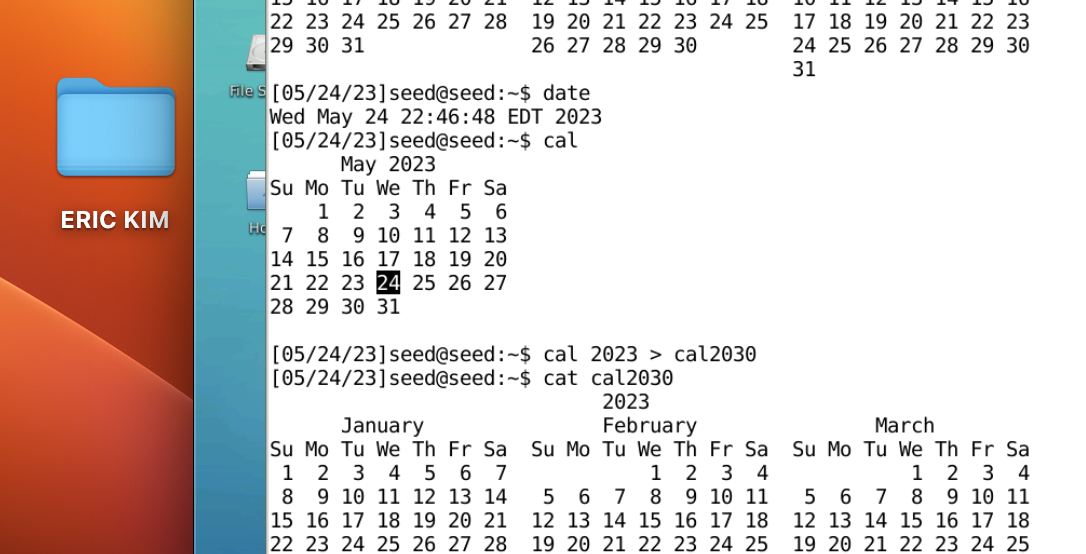
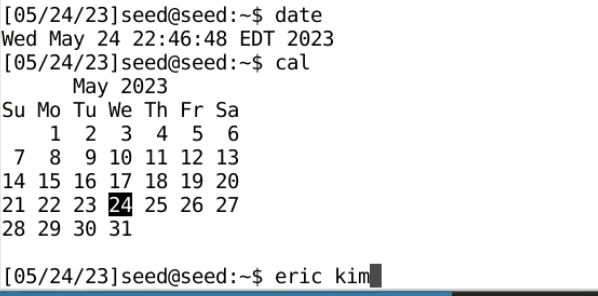
The task was to display the message that the user has inputted, in this case which would be Hello World! We would type echo and append any messages afterwards. We could also store the messages into the file. Then, we could also see the messages within the file using the cat command which stands for concatenate.



Observation: The task allowed to display the message into the terminal and redirect that message into a file. The > command overwrites the message that was previously there. The >> appended the message that has already existed. I found that it was interesting that you could concatenate multiple files from one command. 

This command was from the help command. It shows the help command for the command echo. It gives out the details of the command.

Observation: I thought that it was very similar to the man command I was used to. I have searched the difference and noticed that the help command is a built-in command and the man command is the pager program to read the manual.

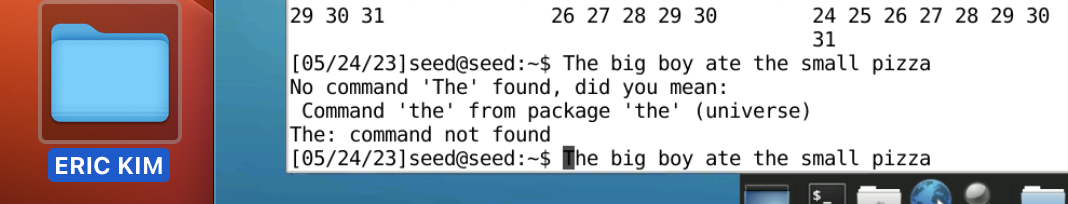


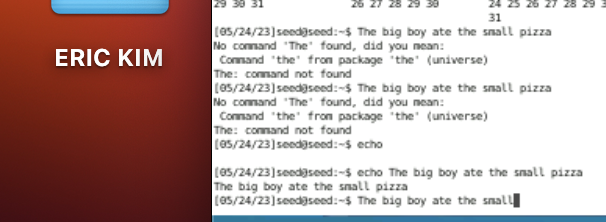
This was the date command and cal command. Date command shows the date in that format as the picture. The cal command shows the date in the second screenshot.

Observation: I have tried to figure out what it would be useful for specifically. I noticed that you could append dates into files which would be useful to keep track of the dates.

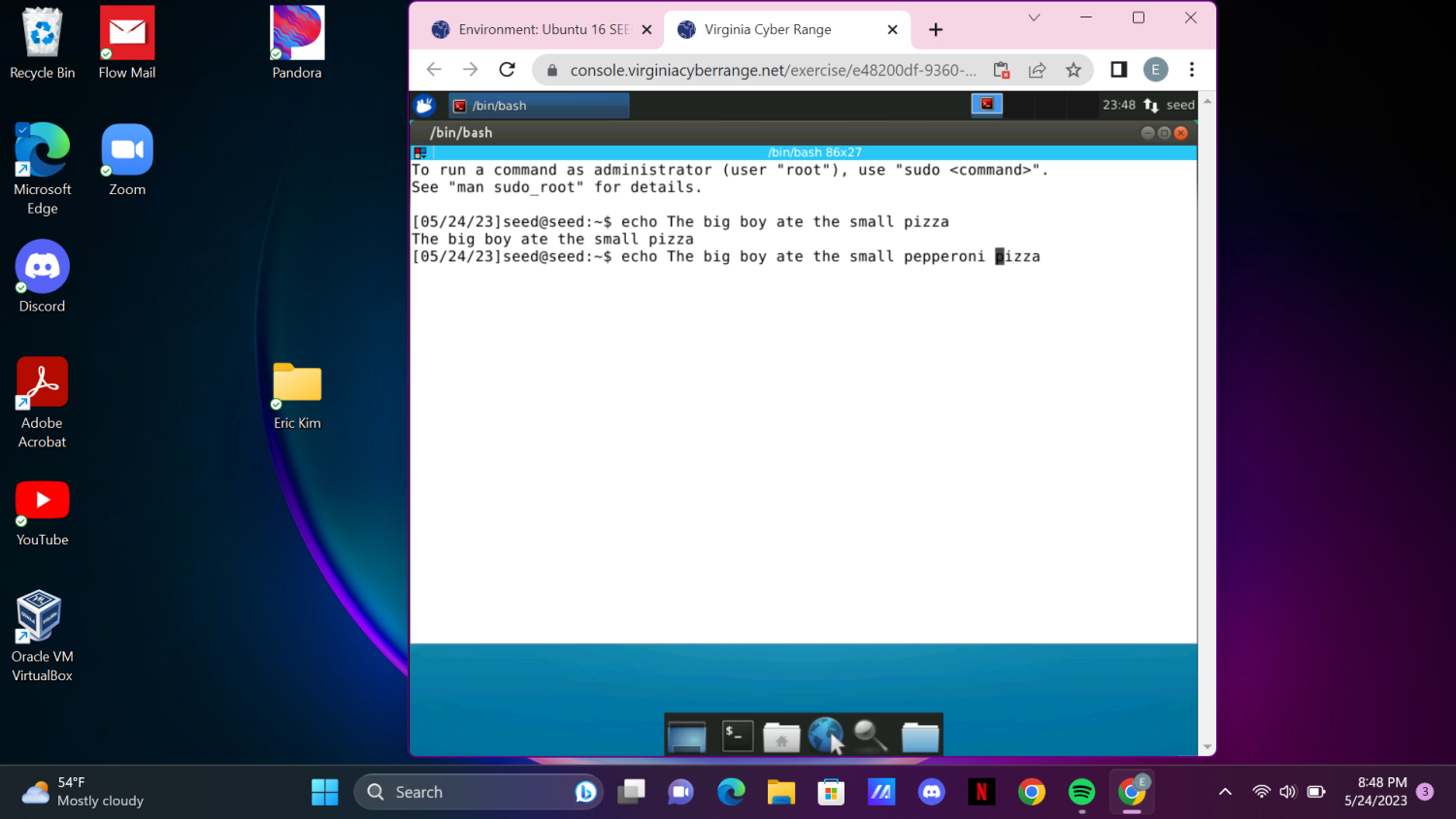
Bash Basics 1-1 Exercise

In the Linux Command Line Interface, there are shortcuts that make it easier for the user to use the terminal. In this exercise, I have used ctrl-A to access the beginning of the message without having to use the left arrow key to navigate towards it. There are other shortcuts such as Ctrl + H to remove the last character of the line, Control + - to undo the last character back, and the Control--U to clear the line that the cursor is on.



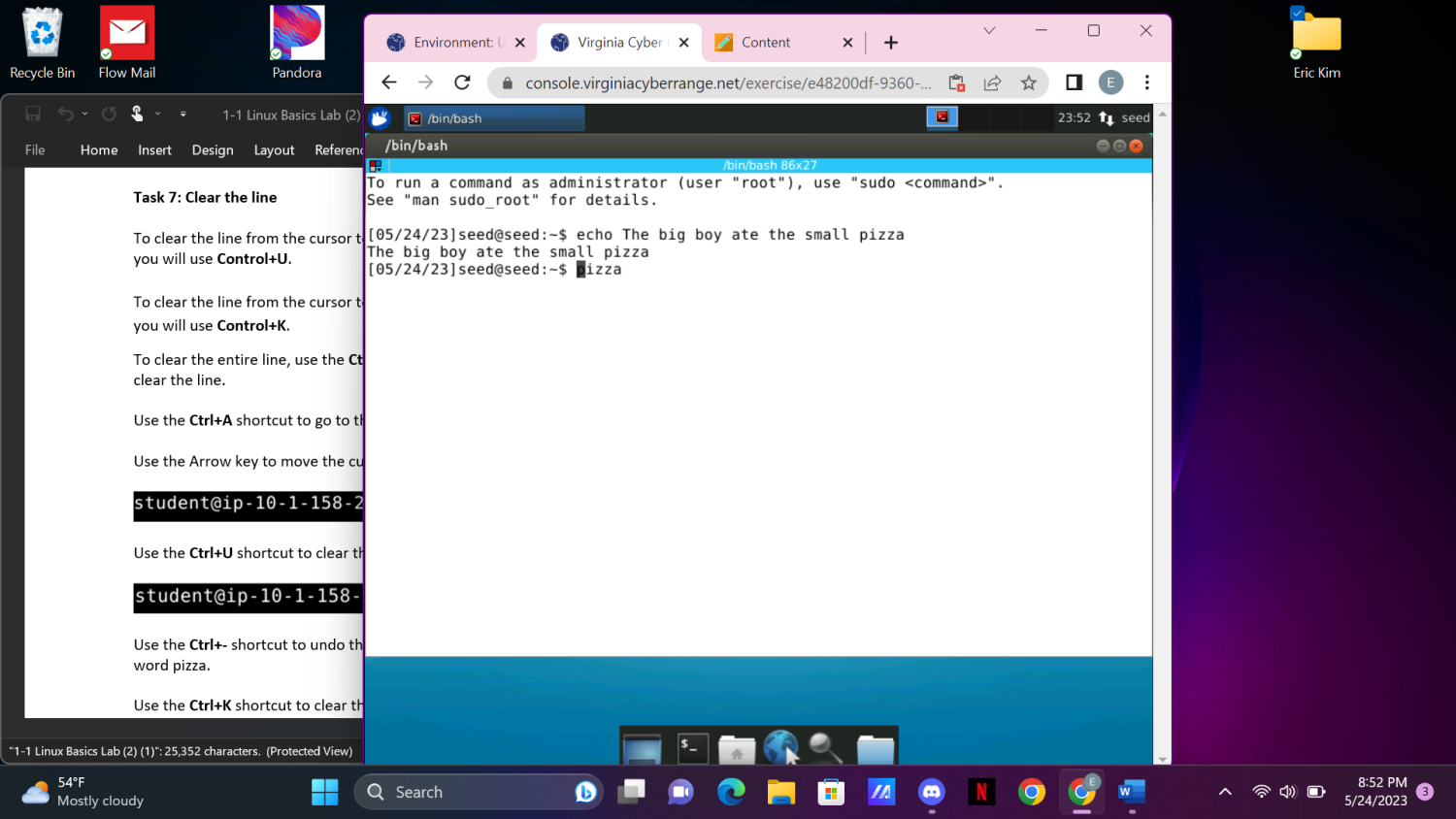
Observation: I notice that the control + - on the cyber range, did not allow me to do the command that I have intended to do. Instead it zoomed out of the screen, just like the screenshot below. When doing control + + it did not do anything. 

Control + L moves the last line up to the last screen and Control + p allows the same function as arrow up, which is the previous command. The down arrow equivalent command would be Control + N . I have used the Alt + B shortcut to move the cursor one word backwards and added the word pepperoni.



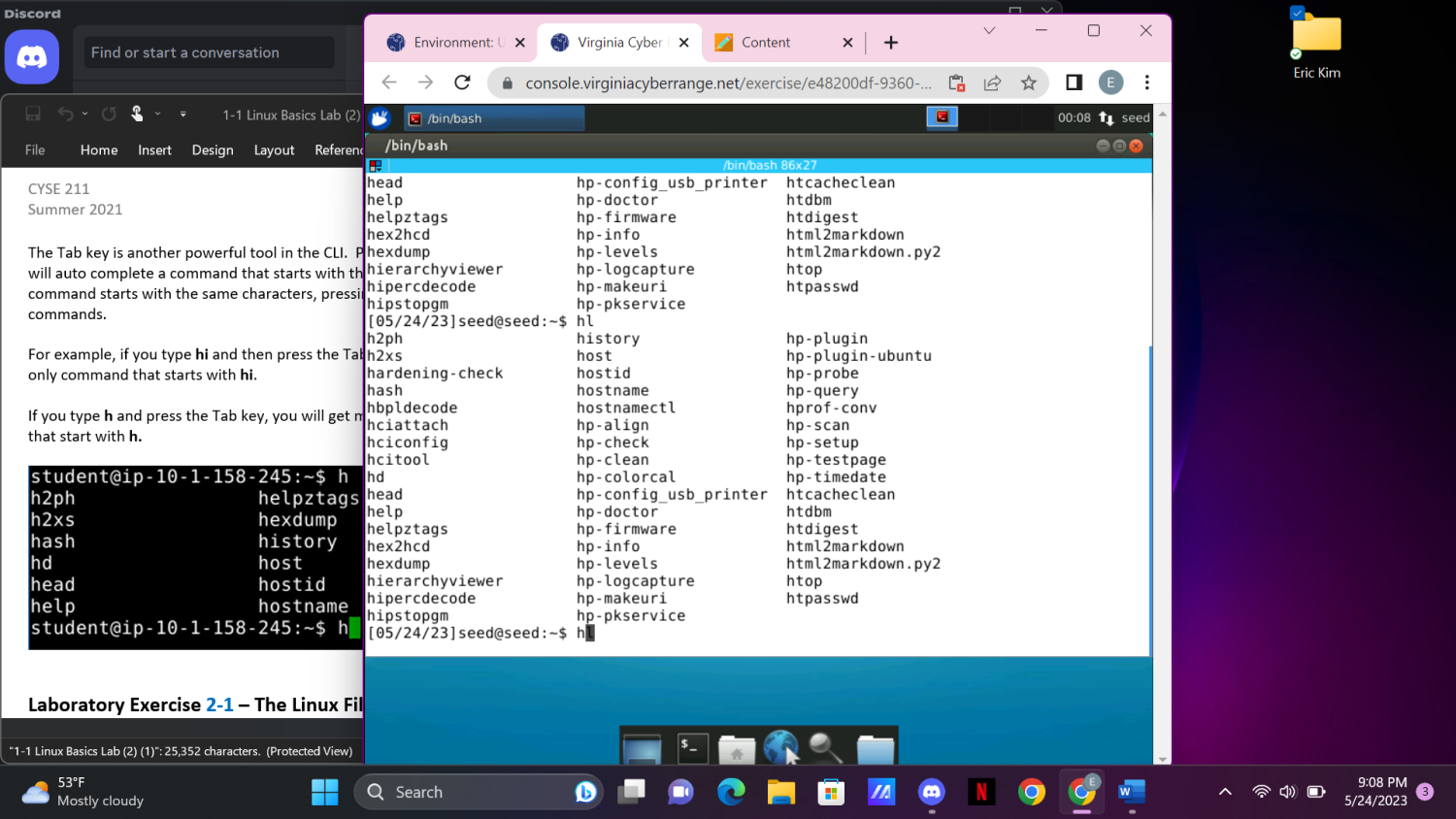
Observations:

I have tried to move onto windows, and see if the ctrl + - command works, but unfortunately it did not. I believe that it’s just the function with the cyber range, as well as the ctrl + N opens a new page instead in the cyber range.

I have used the control + U command and it deleted the lines before pizza. 

I have used Alt + b and then used control + d shortcut to delete the words under the cursor

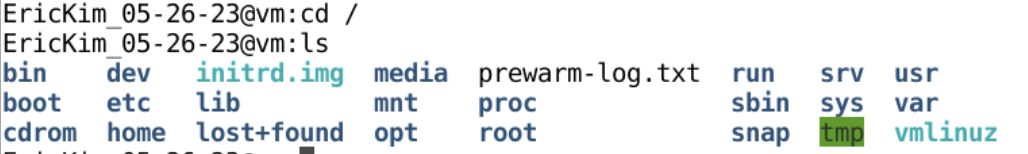
Observation: These commands are really useful, for

The next command I used was the tab function. This is a very useful command because it will auto complete history. For this case typing h and then tab in the command will show multiple commands that start with h. 

I find this command very useful as when writing code in the terminal, I have used the tab often.

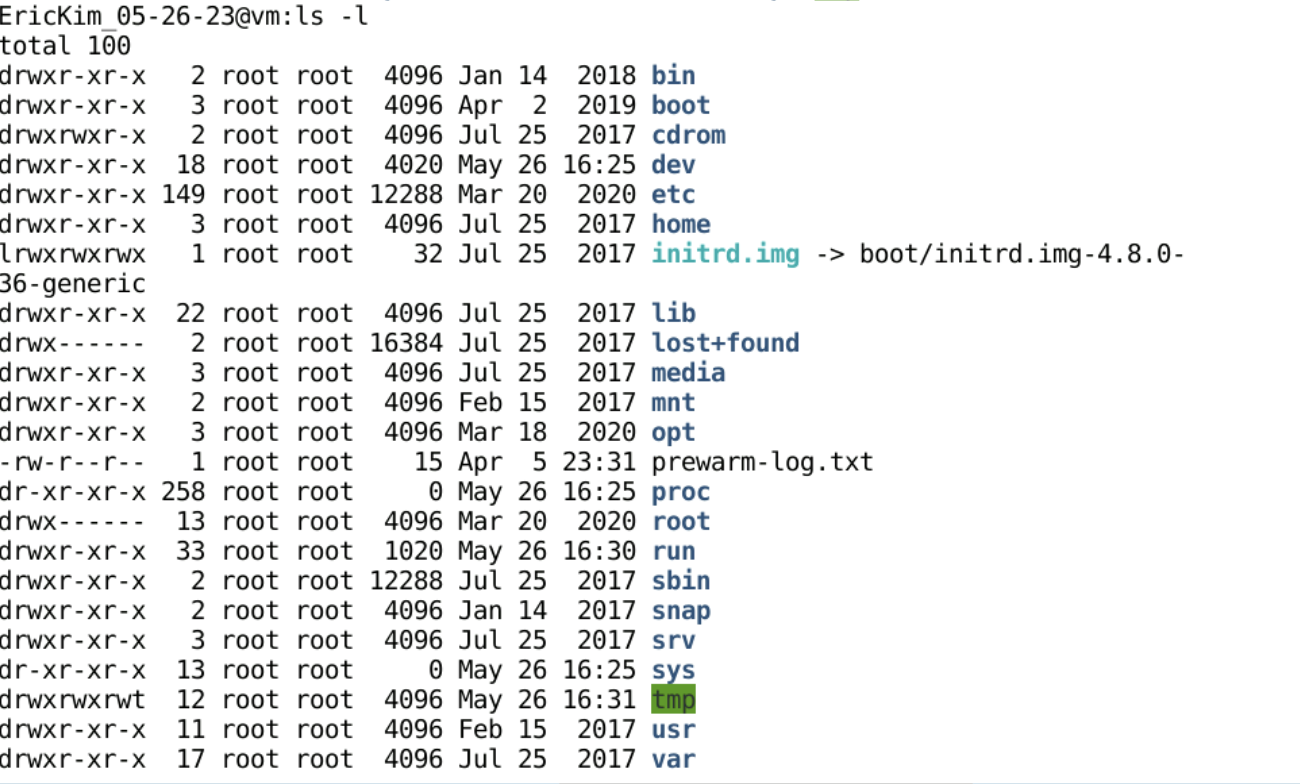
2-1 The Linux File Systems

In this lab, we will examine how to navigate inside of the linux file systems and learn different commands for the file system



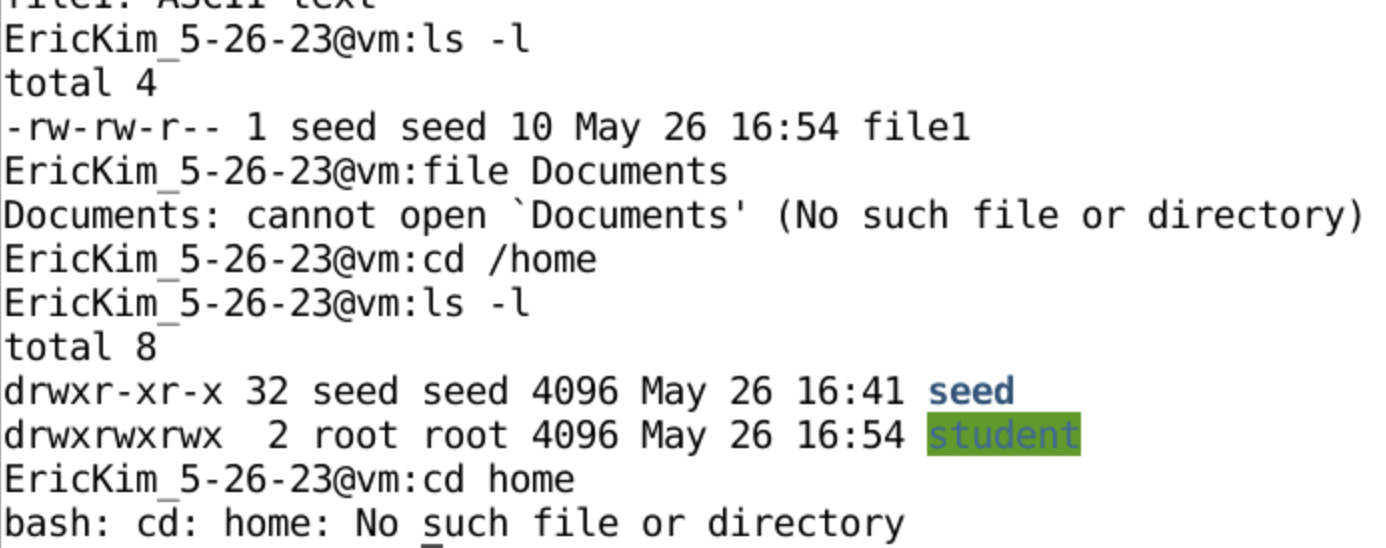
The cd changes the directory to the string that is inputted after the cd command which would be / in this case. The ls shows the list of directories.

Observations: These commands are frequently used in the linux systems. It shows information of privileges,size, date, and the file name

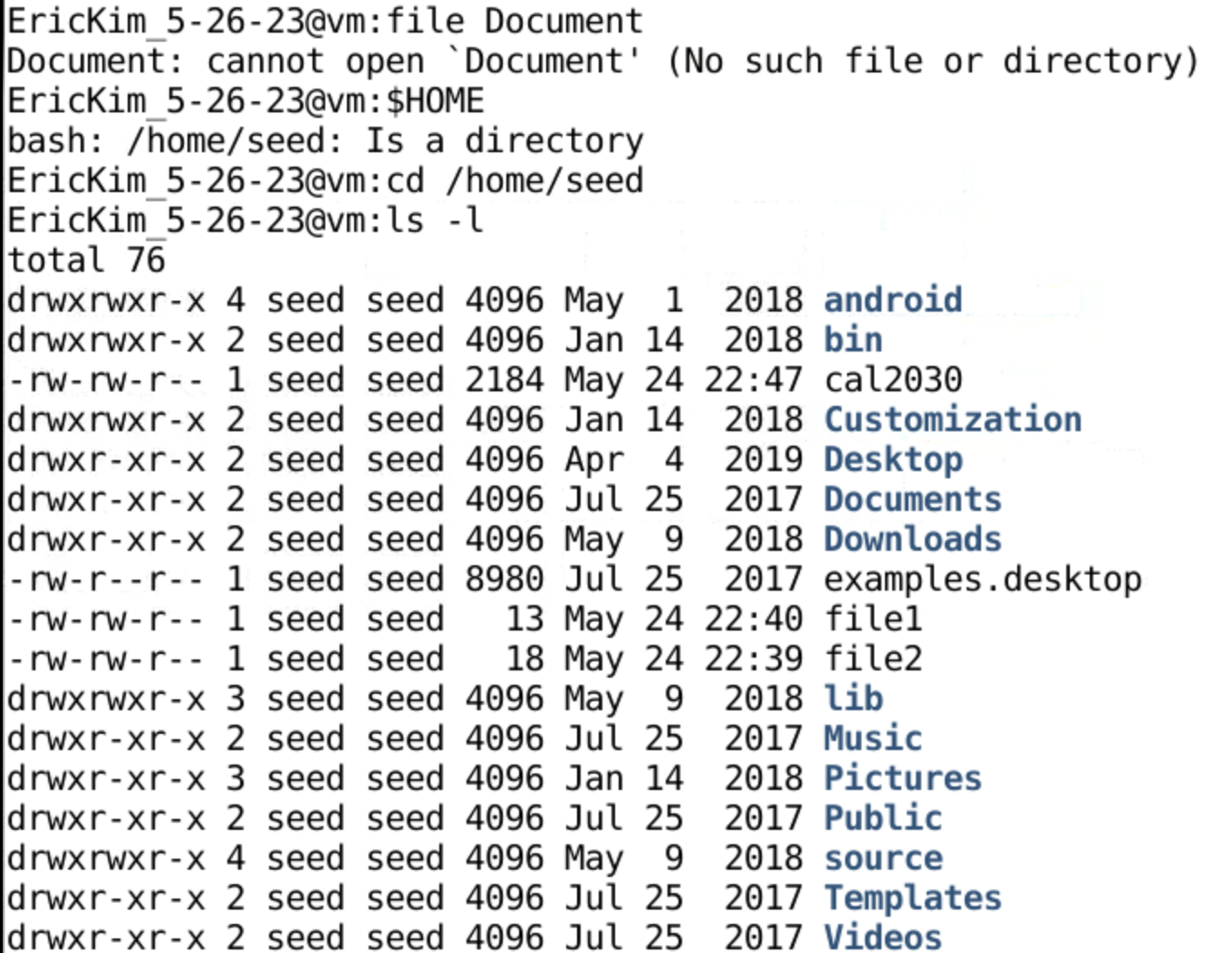


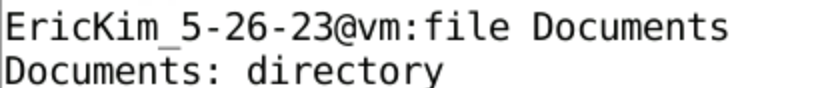
The ls -l command shows the long list of directories. 

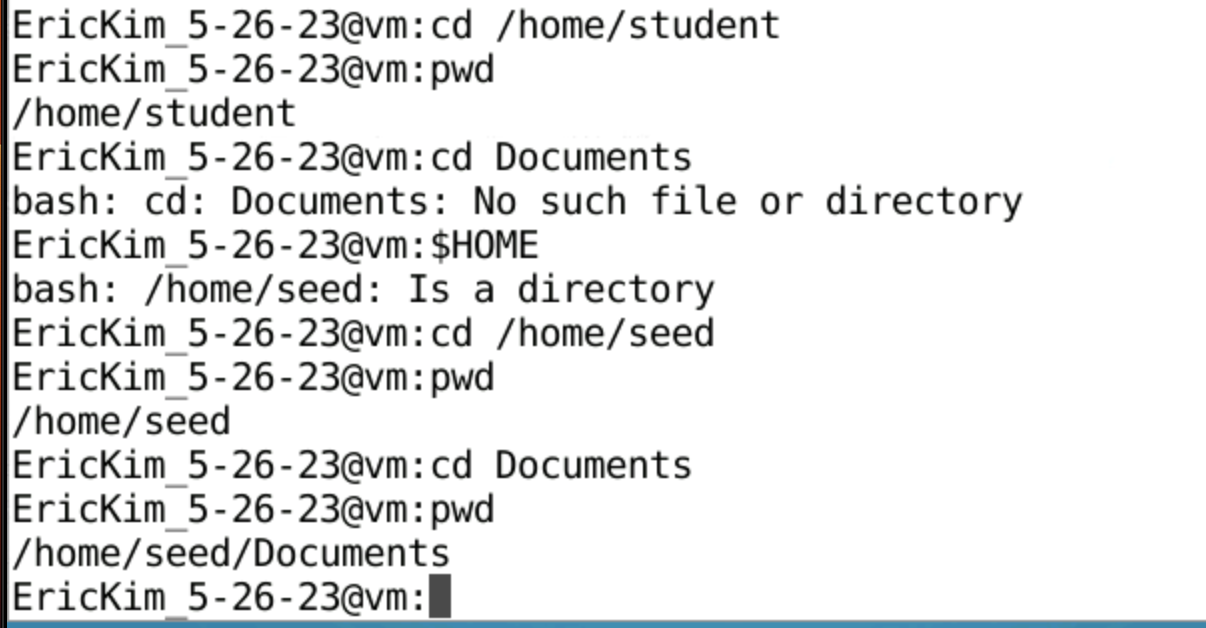
Observations: At first, the cd / home/student command did not work. Because there was no directory under student, I had to create a student directory first. However, when trying the echo Some text > file command to redirect the file into file1 it did not allow permission. Therefore, I did sudo as a root user command chmod 777 student to give all permission to everybody. Afterwards, I was able examine the file1 and saw that it was a basic text file because it said ASCII text.

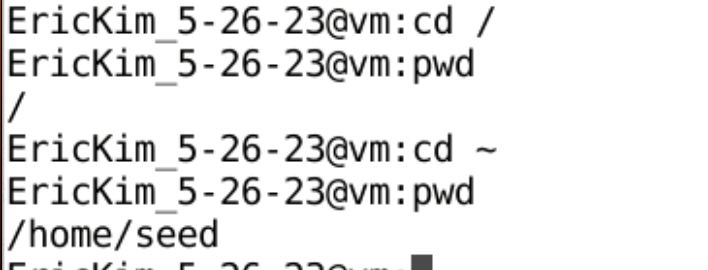


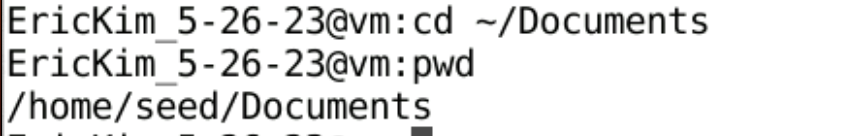
Afterwards, I tried viewing the home directory by ls -l, but I was unable to find the Documents directory under home.



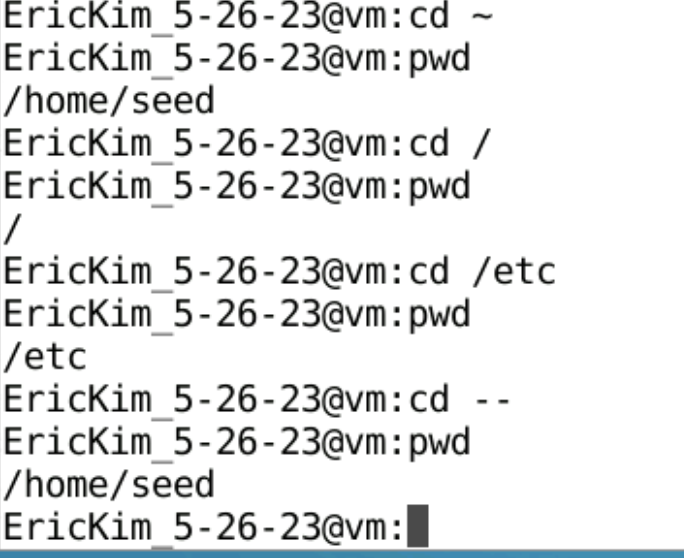
After googling, I was able to find the home directory by typing in $HOME using the environment’s variable to find the home directory. After reaching out to the correct directory, I was able to locate the Documents. 

Then I examined the file type of Documents, which was a directory. 

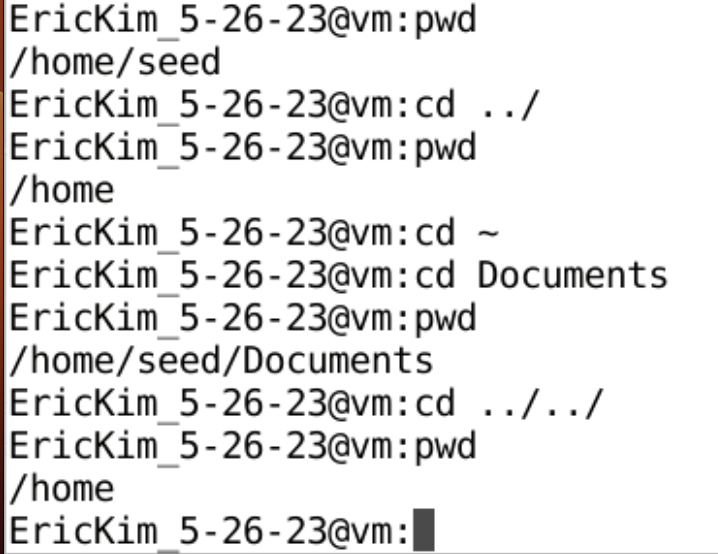
We started with the absolute path which would be /. We have changed to view the seed directory in the home directory that is in the root directory. We later moved into another Document directory that is in the seed directory. Again I noticed that, pwd states where I was located. I have noticed that the cd ~ which is the root directory would also have shown me to /home/seed typing pwd afterwards instead of having to type $HOME. 



This command has taken us to the Document directory right from the home directory.



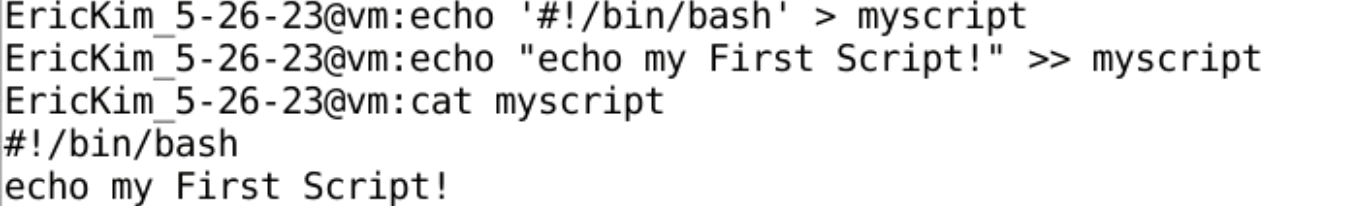
As mentioned before, we have navigated into different parts of the directory in the file systems. First we went to the seed home directory and checked the location. Afterwards, we went to the root directory. Then, we went to the /etc directory. Lastly, we went back to the seed home directory once again by using the cd – command. The cd – command takes us back to the two locations.



The cd ../ moves back one directory. I’ve noticed that from the previous directory it did move back to the home directory. After changing into the seed directory, I went to change into the document directory once again. Therefore I was able to use the cd ../../ command to move back two directory, which would resort back into the home directory.

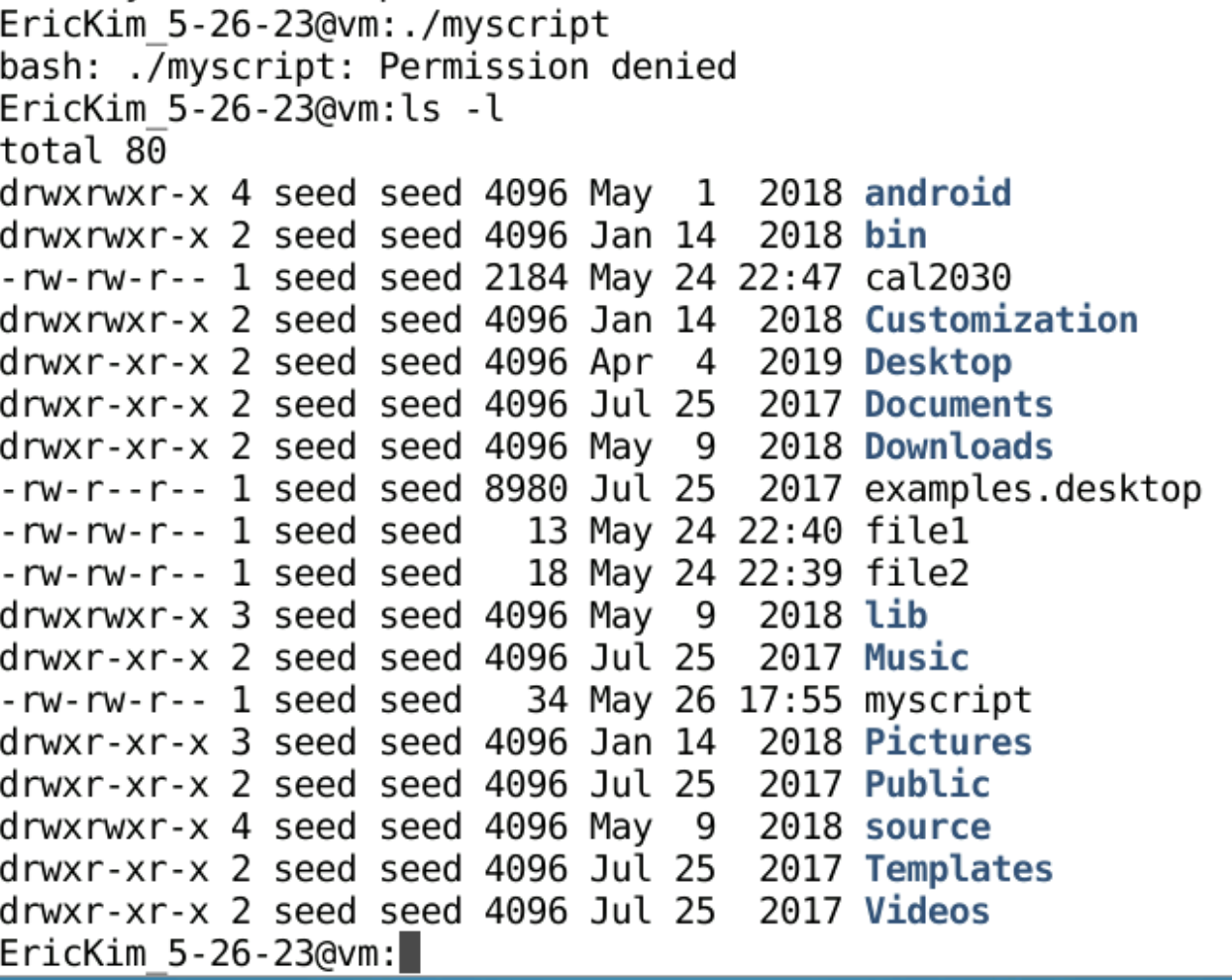
Observations: as many ../ there will resort back to the directory depending on how many times I have stated that many times.



./ means the same directory. Therefore, when concatenating the contents of file1 using ./file1 and cat file1. The output was the same “Hello World!”. 

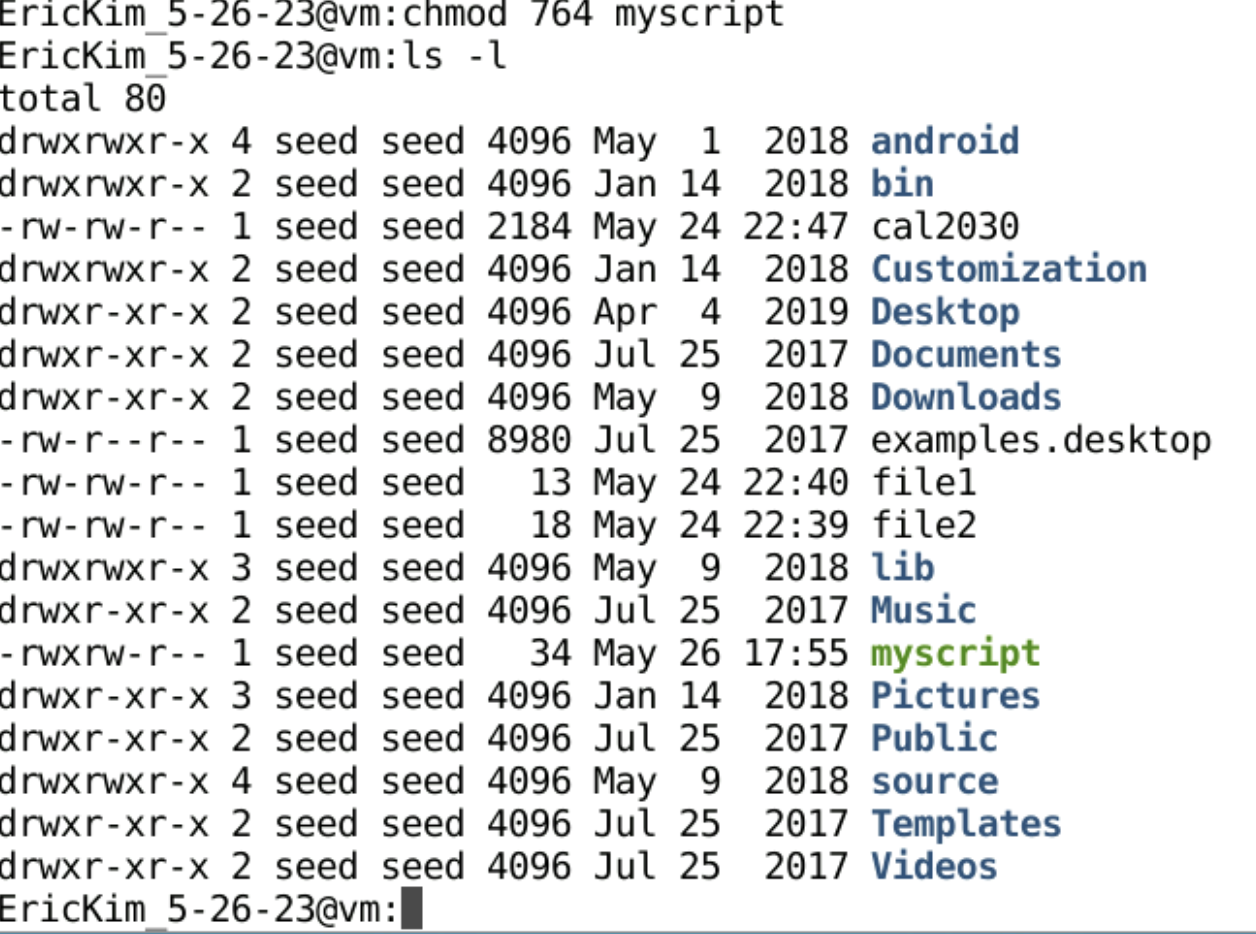
Here, I am writing my first script. I have noticed that for special characters I needed to use the single quotes when writing a string into the file. Then we appended more text into the myscript file. We then checked the content inside of the file.

Observations: The first part is it tells the system which programming language I will be using for the script. In this case it would be the bash.



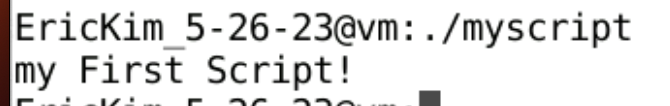
Then we tried running an executable file command ./myscript.

Observations: However, this did not work because of the permission issue. I later checked the permissions from ls -l command and saw that it was only for read and write permissions.



Then we changed the permissions by chmod 764 myscript command. Checking the permissions once again, I was able to see that the user was able to execute files.

Observation: 777 which I am familiar with gives permissions to everyone, but the 764 only changes the permissions for the reader to execute the file.



And finally the script was executed!