Essential Skills – Lesson 23 and 24

Today we will finish our lessons on new Bash commands. Look at the following videos on useful commands - Grep, Sort, And Cut, Pipes and Pipelines, and IO Redirection. NOTE that we do not have the file he is using but concentrate on the commands, what they do, and how they can be useful to us. NOTE: These videos get a bit advanced – for now we want you to understand the purpose of the commands presented.

- https://learning.oreilly.com/videos/linux-commandline/9781771374446/9781771374446-video224196 (Grep, Sort and Cut)
- https://learning.oreilly.com/videos/linux-commandline/9781771374446/9781771374446-video224197 (Pipes and Pipelines)
- https://learning.oreilly.com/videos/linux-commandline/9781771374446/9781771374446-video224198 (I/O Redirection)

Here is a video from O'Reilly that you can watch to summarize Bash and Linux commands. This are not required but can be used if you want a bit of a review.

- https://learning.oreilly.com/videos/linux-fundamentals/9780135560396)

Finally, here is a site that provides you with reference to 101 Bash Commands – like your reference document. You are probably asking why I asked you to create the reference document and just provided this one – well, now I know that you looked at the commands at least once – and your reference document is probably better with more information.

https://dev.to/awwsmm/101-bash-commands-and-tips-for-beginners-to-experts-30je#pwd-ls-cd

Exercise for Practice.

Following is an exercise to help with Bash Commands. Open a Terminal and complete the following exercise using Bash commands only – don't right click and select an option from the menu – that is using the GUI.

Create a document with each command(s) used for each step. Make a note of any options you are not sure of so we can discuss in class.

a. Create 3 files – one called **AddMe.py**, one called **TooMuchFun.txt**, and a third called **Stuff.py**. In the AddMe file write a small program that will add two numbers together. In the TooMuchFun file write a sentence or two about the best part of programming – remember this is just a text file and not a python program. In the

Stuff file write a small program that does some stuff. Don't forget to use Ctrl + S to save each of the files – not necessary with PyCharm but a good habit.

After you create the files, how can you run **AddMe.py** and **Stuff.py**?

b. What is the current directory? List the contents of the current directory. List the contents of the current directory with any hidden files. List the files with the permissions displayed.

c. Display the full contents of each of the files created above. Display only the first 3 lines of AddMe.py and the last 3 lines of Stuff.py. Display the last 3 lines of all files. If you have a file that is large, what options are available to have it appear one screen at a time?

d. Create a directory called **PythonCurrent**, one called **PythonDone** and one called **TextFiles**. In the **TextFiles** directory create the folders **PythonText** and **OtherText**.

e. Change the current directory to **PythonDone – notice how the prompt changes to show the current directory**. Use pwd to confirm you are in the PythonDone directory. List the files. Create some files in this directory – use both .py and .txt extensions. Add about 15-20 lines of code in the .txt file and include the words BugFest and ValText somewhere. List the files again. Go back to the **TooMuchFun.txt** file and add the word BugFest somewhere in the document.

f. Move up a directory level and list the files. You should now be back in the current working directory.

g. Copy the **Stuff.py** folder to the **PythonCurrent** folder and give it the same name. Check and make sure the file has been copied.

h. Move the file **AddMe.py** from its current location to the **PythonDone** directory and give it the name **AddMeDone.py**. Check and make sure the file has been moved and renamed.

i. Display the contents of the **AddMeDone.py** file from the **PythonDone** directory. Use two methods – from current position and changing to the folder. After you are finished return to the working folder.

j. Find the file called **Stuff.py**. Find all files that start with **S**. Why is this an issue?

k. Run the **tree** command. If you do not have it run the command **sudo apt install tree.** Run the tree command again. What information does it provide? How can it be useful to you? Direct the output to a file called **MyTree.Txt**.

l. Find the text **BugFest** in a file and indicate which file(s) it is located – research how you can search subdirectories. Do the same with the word **ValText** – in this case allow the search by ignoring case. Perform one of these again and direct the output to a file called **FindResults.txt**.

m. Change the permission on the **TooMuchFun.txt** file so that the other group has only read access by removing write and execute. Change permissions to the **Stuff.py** file using – **chmod 400 Stuff.py**. What happened and why?

n. Remove each file in the current directory except main.py – Just the files and not the directories.

o. Remove the **PythonCurrent** directory using the rmdir command. What did you need to do first? Do the same with the **PythonDone** directory – prompt the user to confirm the deletion.

p. Remove the **TextFiles** directory using the most efficient method keeping in mind that it contains several other directories and files. Prompt the user to confirm.

If you want to learn a bit about creating bash scripts (files that are set up with Bash commands that can be executed at any time), you can look at the following website that has some videos.

https://linuxconfig.org/bash-scripting-tutorial-for-beginners

See you at 10.