Module Text Manipulation

How to Start Module:

- Before starting the module, Run the <module_script_name> to configure the environment and then <module_script_name> to verify you have done the work correctly
    1. Open a Terminal Window
    2. Clone the GitHub repo (If you have already downloaded Github Repo, skip Step2) (https://github.com/milodigwe/Linux_Essentials_m2itech)
        - From the command line type:
          git clone https://github.com/milodigwe/Linux_Essentials_m2itech
    3. Once repository is cloned, navigate to the Hands_On Folder and find the script named: **text_manipulation.sh**
    4. Run the text_manipulation.sh script: This will configure the environment for the hands-on module
        - **sh ./text_manipulation.sh**
        - The script will ask you for your public IP of your instance (which you can find in your aws console) and your key_pair (which you downloaded and assigned to instance during the ec2 creation process) to log into your instance.
    5. Once the script is finished it will provide you with an output on how to log into the system.
        - Should look like: ssh -i <path to key pair> ec2-user@<ip address>
    6. Once logged in to the instance, Perform the required tasks below.
    7. To verify that you have performed the task correctly. You will need to run the **text_manipulation_check.sh** script located in /home/ec2-user directory.

- **text_manipulation_check.sh** You must score a 100% to pass this module.

8. Please Note * Terminate or Stop your instance when not using it.

HAPPY LEARNING!!!

Questions:

Text and Manipulation + pipe and redirection.

Lab 1: Printing Text (echo)
Task: Use the echo command to print the message "Welcome to Linux!" to the file /home/ec2-user/motd

```
[ec2-user@ip-172-31-30-172 ~]$ echo "Welcome to Linux!" > ~/motd
[ec2-user@ip-172-31-30-172 ~]$ cat ~/motd
Welcome to Linux!
[ec2-user@ip-172-31-30-172 ~]$
```

Lab 2: Searching Text (grep)
Task: Search for the word "unfeminineness" in the file /home/ec2-user/random.txt file and redirect that word to a file called other_words in the same directory

```
[ec2-user@ip-172-31-30-172 ~]$ cat random.txt | grep unfeminineness
unfeminineness
[ec2-user@ip-172-31-30-172 ~]$ cat random.txt | grep unfeminineness > other_words
[ec2-user@ip-172-31-30-172 ~]$ cat other_words
unfeminineness
[ec2-user@ip-172-31-30-172 ~]$
```

Lab 3: Stream Editing (sed)
Task: Replace all occurrences of "old" with "new" in a sample file named sample.txt in /home/ec2-user/ directory.

```
[ec2-user@ip-172-31-30-172 ~]$ cat sample.txt
old1
old2
old3
new1
new2
new3
[ec2-user@ip-172-31-30-172 ~]$ sed -i 's/old/new/g' sample.txt
[ec2-user@ip-172-31-30-172 ~]$ cat sample.txt
new1
new2
new3
new1
new2
new3
[ec2-user@ip-172-31-30-172 ~]$
```

Lab 4: Text Processing (awk)
Task: Extract the first field from each line in the /etc/passwd file and redirect this output to a file called words_results in your home directory.

```
[ec2-user@ip-172-31-30-172 ~]$ awk -F: '{print $1}' /etc/passwd > words_results
[ec2-user@ip-172-31-30-172 ~]$ cat words_results
root
bin
daemon
adm
lp
sync
shutdown
halt
mail
operator
games
ftp
nobody
dbus
systemd-network
systemd-oom
systemd-resolve
sshd
rpc
libstoragemgmt
systemd-coredump
systemd-timesync
chrony
ec2-instance-connect
rpcuser
tcpdump
ec2-user
```

Lab 5: Word Count (wc)

Task: Count the number of lines, words, and characters in the words_results file.
First find the number of lines and redirect that number to line_results folder in home directory.

```
[ec2-user@ip-172-31-26-195 ~]$ cat words_results  | wc -l
27
[ec2-user@ip-172-31-26-195 ~]$ cat words_results  | wc -l > line_results
[ec2-user@ip-172-31-26-195 ~]$ cat line_results
27
[ec2-user@ip-172-31-26-195 ~]$
```

Lab 6: Word Count (wc)
Task: Find the total number of characters in the words_results file and redirect that number to a file called character results.

```
[ec2-user@ip-172-31-26-195 ~]$ cat words_results | wc -m
233
[ec2-user@ip-172-31-26-195 ~]$ cat words_results | wc -m > character_results
[ec2-user@ip-172-31-26-195 ~]$
```

Lab 7: Word Count (wc)
Task: Find the total number of words in words_results and redirect that number to a file called total_words_results

```
[ec2-user@ip-172-31-30-172 ~]$ cat /home/ec2-user/words_results | wc -w
28
[ec2-user@ip-172-31-30-172 ~]$ cat /home/ec2-user/words_results | wc -w > total_words_results
```

Lab 8: Viewing the Beginning of a File (head)
Task: Display the first 5 lines of the /home/ec2-user/random.txt redirect this output to head_random.txt

```
[ec2-user@ip-172-31-30-172 ~]$ head -n5 random.txt
four-ply
FOS
incinerations
four-stranded
onomatopoetically
[ec2-user@ip-172-31-30-172 ~]$ head -n5 random.txt > head_random.txt
[ec2-user@ip-172-31-30-172 ~]$ cat head_random.txt
four-ply
FOS
incinerations
four-stranded
onomatopoetically
[ec2-user@ip-172-31-30-172 ~]$
```

Lab 9: Viewing the End of a File (tail)
Task: Display the last 5 lines of the /home/ec2-user/random.txt redirect this output to tail_random.txt

```
[ec2-user@ip-172-31-30-172 ~]$ tail -n5 random.txt
caddying
unproportionable
pimbina
currawong
seroperitoneum
[ec2-user@ip-172-31-30-172 ~]$ tail -n5 random.txt  > tail_random.txt
[ec2-user@ip-172-31-30-172 ~]$ cat tail_random.txt
caddying
unproportionable
pimbina
currawong
seroperitoneum
[ec2-user@ip-172-31-30-172 ~]$
```

Lab 10: Extracting Fields from Existing File (cut)

Task: Use the cut command to extract the username and home directory of the systemd-timesync user from the /etc/passwd file. First extract the username of "systemd-timesync" and redirect that name to cut_results.

```
[ec2-user@ip-172-31-30-172 ~]$ cat /etc/passwd | grep systemd-timesync | cut -d ":" -f1
systemd-timesync
[ec2-user@ip-172-31-30-172 ~]$ cat /etc/passwd | grep systemd-timesync | cut -d ":" -f1 > cut_results
[ec2-user@ip-172-31-30-172 ~]$ cat cut_results
systemd-timesync
[ec2-user@ip-172-31-30-172 ~]$
```

Lab 11: Extracting Fields from Existing File (cut)

Task: Extract the home directory name for the user systemd-timesync to the cut_home_dir_results. ** Note this can be achieved using other commands used above such as cat and grep.

```
[ec2-user@ip-172-31-30-172 ~]$ cat /etc/passwd | grep systemd-timesync | cut -d ":" -f7 > cut_home_dir_results
[ec2-user@ip-172-31-30-172 ~]$ cat cut_home_dir_results
/usr/sbin/nologin
[ec2-user@ip-172-31-30-172 ~]$
```

Once Complete, Run the check script

```
[ec2-user@ip-172-31-30-172 ~]$ sh ./text_manipulation_check.sh
1. Checking if Motd file exist and expected text is in the file. PASS
PASS

2. Checking if other_words file exists and the words unfemininess exist inside the file. PASS
PASS

3. Checking if sample.txt file exists and the word 'old' exists inside the file. PASS
PASS

4. Checking if words_results file exist and the first fields has been exported from the /etc/passed column. PASS
PASS

5. Checking if line_results file exists and you have exported the correct number of lines. PASS
PASS

6. Checking if character_results file exists and you have exported the correct number of characters. PASS
PASS

7. Checking if words_results file exists and you have exported the correct words to the file. PASS
PASS

8. Checking if head_random.txt file exist and you have exported the first 5 lines from the random_txt folder. PASS
PASS

9. Checking if tail_random.txt file exist and you have exported the last 5 lines from the random_txt folder. PASS
PASS

10. Checking if cut_results file exist and the correct username is here. PASS
PASS

11. Checking if cut_home_dir file exist and the correct home directory is present. PASS
PASS

Score: 11 / 11
Your score is 100%, You have passed this module!!

Number of Correct : 11 / Number of Fail : 0 PASS
[ec2-user@ip-172-31-30-172 ~]$
```