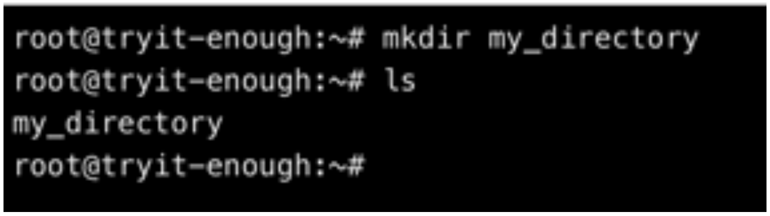
Assignment: Bash Shell Basics

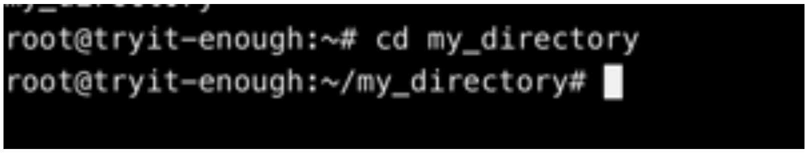
Task 1

1:Creating a directory called “my_directory”



```
root@tryit-enough:~# mkdir my_directory
root@tryit-enough:~# ls
my_directory
root@tryit-enough:~#
```

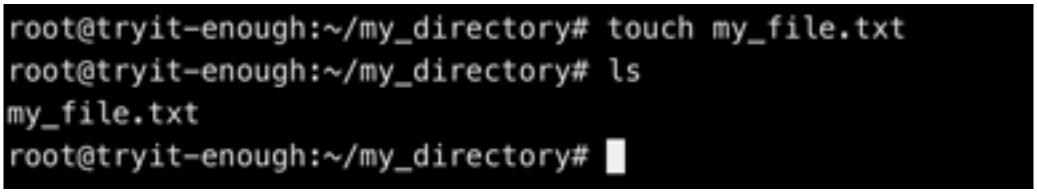
2:Navigating into “my_directory”



```
root@tryit-enough:~# cd my_directory
root@tryit-enough:~/my_directory#
```

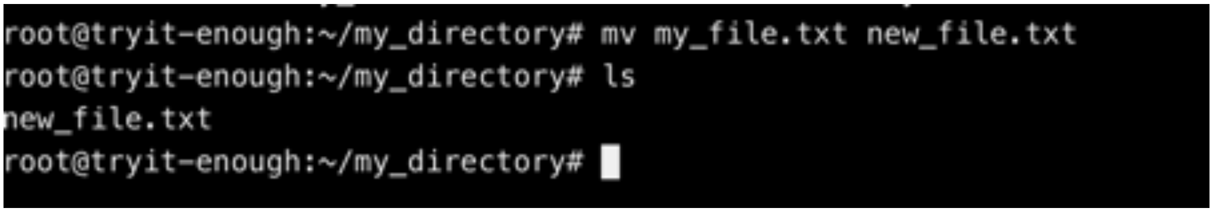
3: creating a empty file called “my_file.txt”

4:list all files and directoriey in current directory



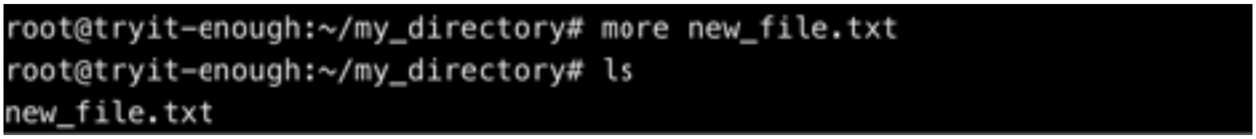
```
root@tryit-enough:~/my_directory# touch my_file.txt
root@tryit-enough:~/my_directory# ls
my_file.txt
root@tryit-enough:~/my_directory#
```

5:Renaming “my file.txt”using pager tool of your choise



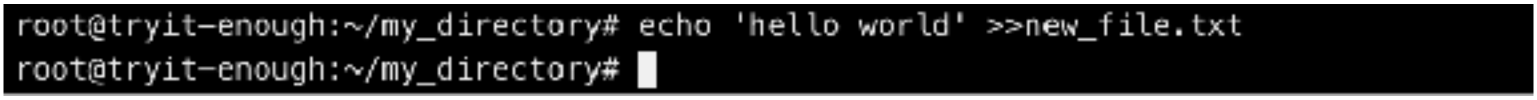
```
root@tryit-enough:~/my_directory# mv my_file.txt new_file.txt
root@tryit-enough:~/my_directory# ls
new_file.txt
root@tryit-enough:~/my_directory#
```

6:displaying the content of the “new_text file”



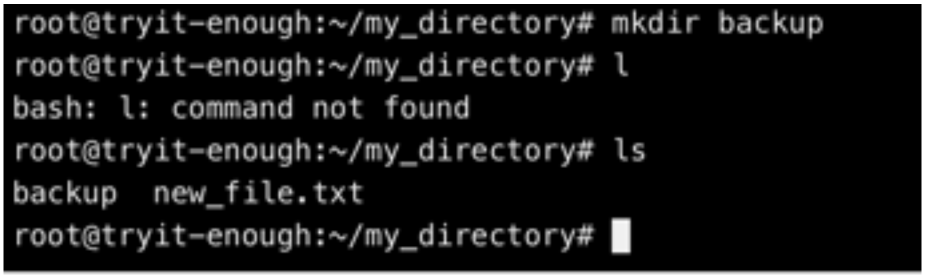
```
root@tryit-enough:~/my_directory# more new_file.txt
root@tryit-enough:~/my_directory# ls
new_file.txt
```

7:Append the text “hello world to new file”



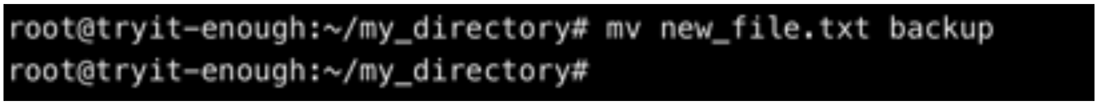
```
root@tryit-enough:~/my_directory# echo 'hello world' >>new_file.txt
root@tryit-enough:~/my_directory# █
```

8:Create a new directory called backup:



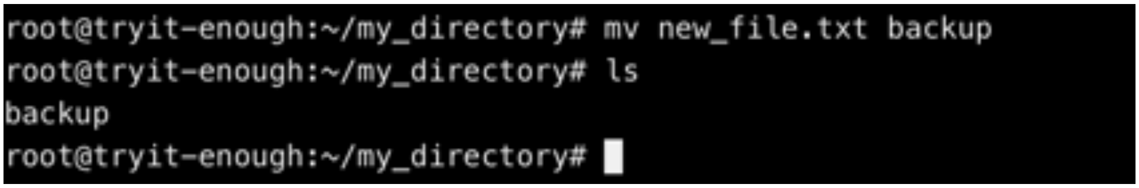
```
root@tryit-enough:~/my_directory# mkdir backup
root@tryit-enough:~/my_directory# l
bash: l: command not found
root@tryit-enough:~/my_directory# ls
backup  new_file.txt
root@tryit-enough:~/my_directory# █
```

9:Move new_file.txt to backup directory



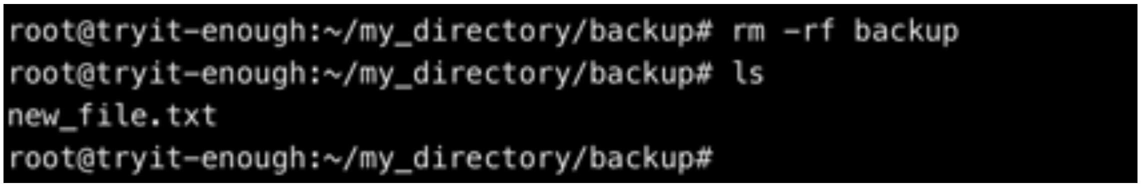
```
root@tryit-enough:~/my_directory# mv new_file.txt backup
root@tryit-enough:~/my_directory#
```

10: verify that new file is now located I backup



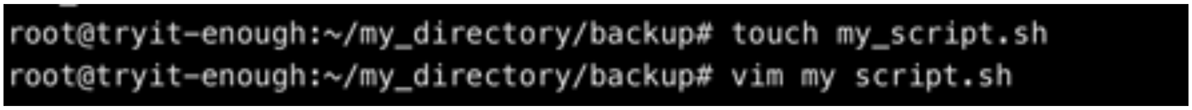
```
root@tryit-enough:~/my_directory# mv new_file.txt backup
root@tryit-enough:~/my_directory# ls
backup
root@tryit-enough:~/my_directory# █
```

11:delete the backup directory and all its contents:



```
root@tryit-enough:~/my_directory/backup# rm -rf backup
root@tryit-enough:~/my_directory/backup# ls
new_file.txt
root@tryit-enough:~/my_directory/backup#
```

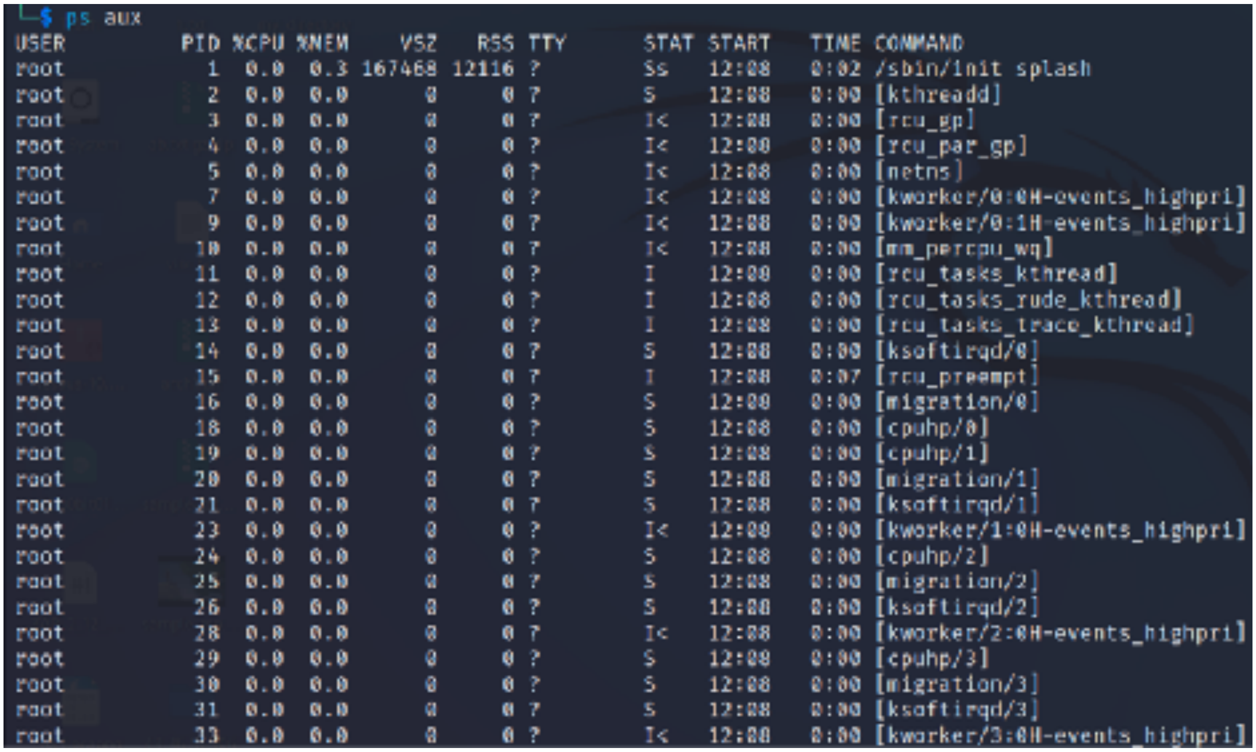
Task 2: Permission and scripting :
1:Creating a new file called “my script.sh”
2:Editing my script sh using text editor of your choise



```
root@tryit-enough:~/my_directory/backup# touch my_script.sh
root@tryit-enough:~/my_directory/backup# vim my script.sh
```

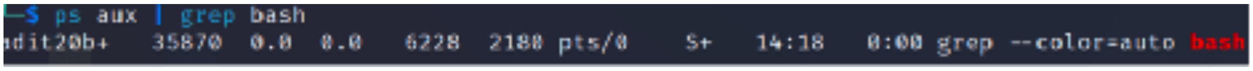
Task 3: Command Execution and Pipelines

● List all the processes running on your system using the "ps" command.



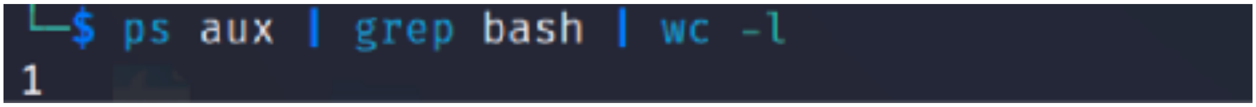
```
└─$ ps aux
USER         PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root           1  0.0  0.3 167468 12116 ?        Ss   12:08   0:02 /sbin/init splash
root           2  0.0  0.0      0     0 ?        S    12:08   0:00 [kthreadd]
root           3  0.0  0.0      0     0 ?        I<   12:08   0:00 [rcu_gp]
root           4  0.0  0.0      0     0 ?        I<   12:08   0:00 [rcu_par_gp]
root           5  0.0  0.0      0     0 ?        I<   12:08   0:00 [netns]
root           7  0.0  0.0      0     0 ?        I<   12:08   0:00 [kworker/0:0H-events_highpri]
root           9  0.0  0.0      0     0 ?        I<   12:08   0:00 [kworker/0:1H-events_highpri]
root          10  0.0  0.0      0     0 ?        I<   12:08   0:00 [mm_percpu_wq]
root          11  0.0  0.0      0     0 ?        I    12:08   0:00 [rcu_tasks_kthread]
root          12  0.0  0.0      0     0 ?        I    12:08   0:00 [rcu_tasks_rude_kthread]
root          13  0.0  0.0      0     0 ?        I    12:08   0:00 [rcu_tasks_trace_kthread]
root          14  0.0  0.0      0     0 ?        S    12:08   0:00 [ksoftirqd/0]
root          15  0.0  0.0      0     0 ?        I    12:08   0:07 [rcu_preempt]
root          16  0.0  0.0      0     0 ?        S    12:08   0:00 [migration/0]
root          18  0.0  0.0      0     0 ?        S    12:08   0:00 [cpuhp/0]
root          19  0.0  0.0      0     0 ?        S    12:08   0:00 [cpuhp/1]
root          20  0.0  0.0      0     0 ?        S    12:08   0:00 [migration/1]
root          21  0.0  0.0      0     0 ?        S    12:08   0:00 [ksoftirqd/1]
root          23  0.0  0.0      0     0 ?        I<   12:08   0:00 [kworker/1:0H-events_highpri]
root          24  0.0  0.0      0     0 ?        S    12:08   0:00 [cpuhp/2]
root          25  0.0  0.0      0     0 ?        S    12:08   0:00 [migration/2]
root          26  0.0  0.0      0     0 ?        S    12:08   0:00 [ksoftirqd/2]
root          28  0.0  0.0      0     0 ?        I<   12:08   0:00 [kworker/2:0H-events_highpri]
root          29  0.0  0.0      0     0 ?        S    12:08   0:00 [cpuhp/3]
root          30  0.0  0.0      0     0 ?        S    12:08   0:00 [migration/3]
root          31  0.0  0.0      0     0 ?        S    12:08   0:00 [ksoftirqd/3]
root          33  0.0  0.0      0     0 ?        I<   12:08   0:00 [kworker/3:0H-events_highpri]
```

- ● Use the "grep" command to filter the processes list and display only the processes with "bash" in their name.



```
└─$ ps aux | grep bash
adit20b+    35870  0.0  0.0   6228  2180 pts/0    S+   14:18   0:00 grep --color=auto bash
```

- Use the "wc" command to count the number of lines in the filtered output.



```
└─$ ps aux | grep bash | wc -l
1
```

- **Submission:**
  Provide a document or text file containing the commands used to complete the tasks above, along with any relevant output or screenshots. Include your explanations or observations where necessary.