

This exercise has 25 questions on module 2 Working in the shell.

First without answers, the second set is with answers.

Establish a login to your VM.

1. What is your user id? (Use multiple commands)
2. What is your process id? (Use multiple commands)
3. Create a variable 'NUM' with value '33'
4. Start a new bash shell.
5. What is the content of the 'NUM' variable?
6. Exit the shell and check the content of the variable NUM.
7. Export the variable NUM, start a new shell and check the content of the NUM variable
8. List all variable of your current shell.
9. List all exported variables of your current shell.
10. Store the output of the env command in a file called 'ENV\_OUT' in your login directory. (use output redirection)
11. Store the output of the set command in a file called 'SET\_OUT' in your login directory. (use output redirection)
12. What happens if you run the following command:  

```
echo hello1 > ~/SET_OUT
```

What happens if you run the following command:

```
echo hello2 >> ~/SET_OUT
```

13. Change your user id to root using the su command.

14. Change your user id to root using the sudo command.

15. What does the 'source' command do?

16. What is another way to source a script?

17. You are in your login directory. (cd)  
Change to the /etc directory using an absolute pathname.

Change to the var directory using a relative pathname.

18. Use the history command to view the commands you entered.

19. Use <CTRL+R> to search for the sudo bash command and rerun that.

20. What is the functionality of an unnamed pipe?

21. Use the man command to see how the 'wc' command works.

22. Use an unnamed pipe to count all the processes on your system.

23. Use the man command to see how the 'sort' command works.

24. Use an unnamed pipe and output redirection to generate an sorted list of all processes on your system and store that list in a file called 'PROC\_LIST'.

25. When you look at the contents of PROC\_LIST, is this list really sorted on the first field? Find out how you can sort the list on numerical values.

Generated a sorted list of all processes based on numerical values and store that in PROC\_LIST\_NUM

Check how PROC\_LIST differs from PROC\_LIST\_NUM

---

Establish a login to your VM.

1. What is your user id? (Use multiple commands)

```
whoami
who am i
id
```

2. What is your process id? (Use multiple commands)

```
ps
echo $$
```

3. Create a variable 'NUM' with value '33'

```
NUM=33
```

4. Start a new bash shell.

```
bash
```

5. What is the content of the 'NUM' variable?

```
echo $NUM
```

6. Exit the shell and check the content of the variable NUM.

```
exit
echo $NUM
```

7. Export the variable NUM, start a new shell and check the content of the NUM variable

```
export NUM
bash echo $NUM
```

8. List all variable of your current shell.

```
set
```

9. List all exported variables of your current shell.

```
env
```

10. Store the output of the env command in a file called 'ENV\_OUT' in your login directory. (use output redirection)

```
env > ~/ENV_OUT
```

11. Store the output of the set command in a file called 'SET\_OUT' in your login directory. (use output redirection)

```
set > ~/SET_OUT
```

12. What happens if you run the following command:

```
echo hello1 > ~/SET_OUT
```

What happens if you run the following command:

```
echo hello2 >> ~/SET_OUT
```

13. Change your user id to root using the su command.

```
su
```

```
id
exit
```

14. Change your user id to root using the sudo command.

```
sudo bash
exit
```

15. What does the 'source' command do?

**The source command runs a script in the current shell and does not fork.**

16. What is another way to source a script?

```
. scriptname
```

17. You are in your login directory. (cd)  
Change to the /etc directory using an absolute pathname.

```
cd /etc
```

Change to the var directory using a relative pathname.

```
cd ../var
```

18. Use the history command to view the commands you entered.

```
history
```

19. Use <CTRL+R> to search for the sudo bash command and rerun that.

```
CTRL+R sudo
```

20. What is the functionality of an unnamed pipe?

**The unnamed pipe will pass the output of one command through to another command.**

21. Use the man command to see how the 'wc' command works.

```
man wc
```

22. Use an unnamed pipe to count all the processes on your system.

```
ps -ef | wc -l
```

23. Use the man command to see how the 'sort' command works.

```
man sort
```

24. Use an unnamed pipe and output redirection to generate an sorted list of all processes on your system and store that list in a file called 'PROC\_LIST'.

```
ps -e | sort > PROC_LIST
```

25. When you look at the contents of PROC\_LIST, is this list really sorted on the first field? Find out how you can sort the list on numerical values.

```
man sort
```

Generated a sorted list of all processes based on numerical values and store that in PROC\_LIST\_NUM

```
ps -e | sort -n > PROC_LIST_NUM
```

Check how PROC\_LIST differs from PROC\_LIST\_NUM

```
cat PROC_LIST PROC_LIST_NUM | more
```