Day 5 – Phase 5 Scripting Automation, Redirection & FDs

Tasks:

1.

```
saleh@saleh-VirtualBox:/home$ export SENSOR_TYPE="Temperature"
saleh@saleh-VirtualBox:/home$ echo $SENSOR_TYPE
Temperature
```

2,3.

```
saleh@saleh-VirtualBox:/home$ cd ~/iot_logger/scripts/
saleh@saleh-VirtualBox:~/iot_logger/scripts$ nano sensor_script.py
saleh@saleh-VirtualBox:~/iot_logger/scripts$ python3 sensor_script.py >> ~/iot_logger/logs/temp
temp_Hard.log    temp_HL.log    tempreature.log
saleh@saleh-VirtualBox:~/iot_logger/scripts$ python3 sensor_script.py >> ~/iot_logger/logs/tempreature.log
^CTraceback (most recent call last):
  File "/home/saleh/iot_logger/scripts/sensor_script.py", line 8, in <module>
    time.sleep(5)
KeyboardInterrupt

saleh@saleh-VirtualBox:~/iot_logger/scripts$ python3 sensor_script.py >> ~/iot_logger/logs/tempreature.log
^CTraceback (most recent call last):
  File "/home/saleh/iot_logger/scripts/sensor_script.py", line 8, in <module>
    time.sleep(5)
KeyboardInterrupt

saleh@saleh-VirtualBox:~/iot_logger/scripts$ python3 sensor_script.py >> ~/iot_logger/logs/tempreature.log
^CTraceback (most recent call last):
  File "/home/saleh/iot_logger/scripts/sensor_script.py", line 8, in <module>
    time.sleep(5)
KeyboardInterrupt

saleh@saleh-VirtualBox:~/iot_logger/scripts$ python3 ~/iot_logger/scripts/sensor_script.py
Temp: 27.9
Temp: 24.11
Temp: 28.26
Temp: 24.97
Temp: 22.34
Temp: 28.37
Temp: 29.97
Temp: 23.48
Temp: 26.89
^CTraceback (most recent call last):
  File "/home/saleh/iot_logger/scripts/sensor_script.py", line 8, in <module>
    time.sleep(5)
KeyboardInterrupt

saleh@saleh-VirtualBox:~/iot_logger/scripts$ ls ~/iot_logger/logs
hard_link  symbolic_link  temp_Hard.log  temp_HL.log  tempreature.log
saleh@saleh-VirtualBox:~/iot_logger/scripts$ python3 sensor_script.py >> ~/iot_logger/logs/tempreature.log &
[1] 4642
saleh@saleh-VirtualBox:~/iot_logger/scripts$ tail -f ~/iot_logger/logs/tempreature.log
```

4.

```
saleh@saleh-VirtualBox:~/iot_logger/scripts$ man ps
saleh@saleh-VirtualBox:~/iot_logger/scripts$ ps aux |grep sensor_script.py
saleh      4642  0.0  0.3  18780  8960 pts/0    S    18:45   0:00 python3 sensor_script.py
saleh      4665  0.0  0.0   9228  2560 pts/0    S+   18:48   0:00 grep --color=auto sensor_script.py
saleh@saleh-VirtualBox:~/iot_logger/scripts$ ls -l /proc/4642/fd
total 0
lrwx------ 1 saleh saleh 64 0 18:50 3 سبت -> /dev/pts/0
l-wx------ 1 saleh saleh 64 1 18:50 3 سبت -> /home/saleh/iot_logger/logs/tempreature.log
lrwx------ 1 saleh saleh 64 2 18:50 3 سبت -> /dev/pts/0
saleh@saleh-VirtualBox:~/iot_logger/scripts$
```

5.

```
saleh@saleh-VirtualBox:~/iot_logger/logs$ grep "25" ~/iot_logger/logs/tempreature.log > ~/iot_logger/logs/high_temp.log
saleh@saleh-VirtualBox:~/iot_logger/logs$ cat high_temp.log
Temp: 25.15
Temp: 25.81
Temp: 25.52
Temp: 25.91
saleh@saleh-VirtualBox:~/iot_logger/logs$
```

6.

```
saleh@saleh-VirtualBox:~/iot_logger/logs$ cp ~/iot_logger/logs/*.log ~/iot_logger/data/
cp: cannot stat '/home/saleh/iot_logger/logs/temp_Hard.log': No such file or directory
saleh@saleh-VirtualBox:~/iot_logger/logs$ cd ..
saleh@saleh-VirtualBox:~/iot_logger$ cp ./logs/high_temp.log ./data/
saleh@saleh-VirtualBox:~/iot_logger$ cd data
saleh@saleh-VirtualBox:~/iot_logger/data$ ls
high_temp.log  services  temp_HL.log  tempreature.log
saleh@saleh-VirtualBox:~/iot_logger/data$ cat high_temp.log
Temp: 25.15
Temp: 25.81
Temp: 25.52
Temp: 25.91
saleh@saleh-VirtualBox:~/iot_logger/data$
```

7.

```
saleh@saleh-VirtualBox:~/iot_logger/data$ unset SENSOR_TYPE
saleh@saleh-VirtualBox:~/iot_logger/data$ unset Temperature
saleh@saleh-VirtualBox:~/iot_logger/data$ echo $Temperature

saleh@saleh-VirtualBox:~/iot_logger/data$
```

Challenge – Pipes & FD inspection:

```
saleh@saleh-VirtualBox:~/iot_logger/data$ (ls -l; sleep 30) | grep .py
^C
saleh@saleh-VirtualBox:~/iot_logger/data$ ls -l | grep .py & sleep 30
[2] 4813
^C
[2]+  Exit 1                ls --color=auto -l | grep --color=auto .py
saleh@saleh-VirtualBox:~/iot_logger/data$ cd ~/iot_logger/
saleh@saleh-VirtualBox:~/iot_logger$ ps aux | grep sleep
saleh      4818  0.0  0.0   9216  2560 pts/0    S+   19:14   0:00 grep --color=auto sleep
saleh@saleh-VirtualBox:~/iot_logger$ ps aux | grep grep
saleh      4820  0.0  0.0   9216  2560 pts/0    S+   19:14   0:00 grep --color=auto grep
saleh@saleh-VirtualBox:~/iot_logger$ ls -l /proc/4818/fd
ls: cannot access '/proc/4818/fd': No such file or directory
saleh@saleh-VirtualBox:~/iot_logger$ ls  /proc/4818/fd
ls: cannot access '/proc/4818/fd': No such file or directory
saleh@saleh-VirtualBox:~/iot_logger$ ls  /proc/4820/fd
ls: cannot access '/proc/4820/fd': No such file or directory
saleh@saleh-VirtualBox:~/iot_logger$ ls /proc/4820/fd
ls: cannot access '/proc/4820/fd': No such file or directory
saleh@saleh-VirtualBox:~/iot_logger$ ps aux | grep sleep
saleh      4830  0.0  0.0   9216  2560 pts/0    S+   19:17   0:00 grep --color=auto sleep
saleh@saleh-VirtualBox:~/iot_logger$ ls /proc/4830/fd
ls: cannot access '/proc/4830/fd': No such file or directory
saleh@saleh-VirtualBox:~/iot_logger$ ps aux | grep grep
saleh      4835  0.0  0.0   9216  2560 pts/0    S+   19:18   0:00 grep --color=auto grep
saleh@saleh-VirtualBox:~/iot_logger$ ls -l /proc/4853/fd
ls: cannot access '/proc/4853/fd': No such file or directory
saleh@saleh-VirtualBox:~/iot_logger$
```

Open-Ended Questions:

1. What's the difference between ' ' and " " in shell?

single quotes (' ') mean *take everything inside exactly as it is*.
Variables or special characters will not change. For example:
echo '$USER'

Double quotes (" ") allow the shell to expand variables and some
special characters. For example:echo "$USER"

2. Explain [-f filename] vs [-d dirname]?

[-f filename]: checks if a file exists and it's a normal file (not a folder).

[-d dirname]: checks if something exists and it's a directory (folder).

Example:

[-f test.txt] && echo "File exists"

[-d myfolder] && echo "Directory exists"

3. Explain stdout/stderr redirection, appending vs overwrite. How can you confirm redirection using file descriptors?

Every program has 3 streams:

- 0 → stdin (input)

- 1 → stdout (normal output)

- 2 → stderr (error messages)

Redirection changes where output goes:

- > = overwrite file with new output

- >> = append output at the end of file (keep old data)

- 2> = redirect error messages to a file

- &> = redirect both stdout + stderr

Example:

ls > files.txt   # overwrite files.txt

ls >> files.txt   # add to files.txt

ls 2> errors.txt   # only errors go to errors.txt
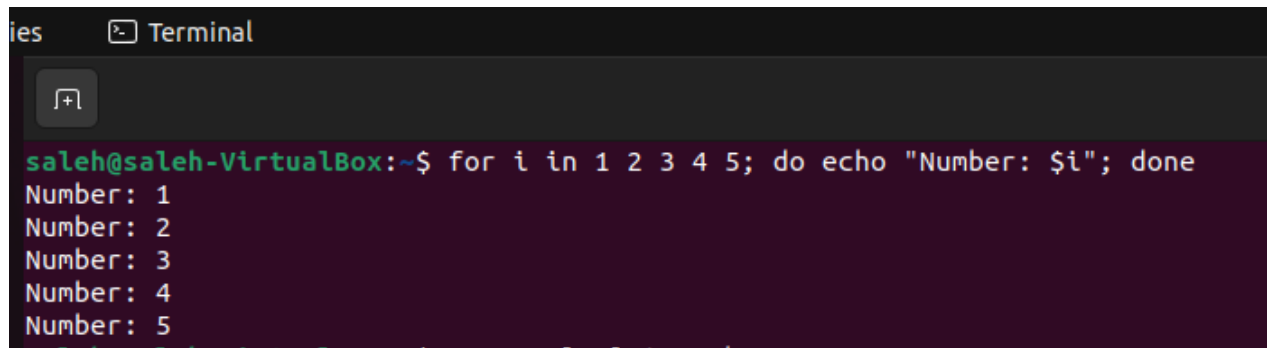
To confirm redirection, you can check the process's file descriptors:

ls -l /proc/<pid>/fd

There you'll see where 1 (stdout) and 2 (stderr) are pointing.
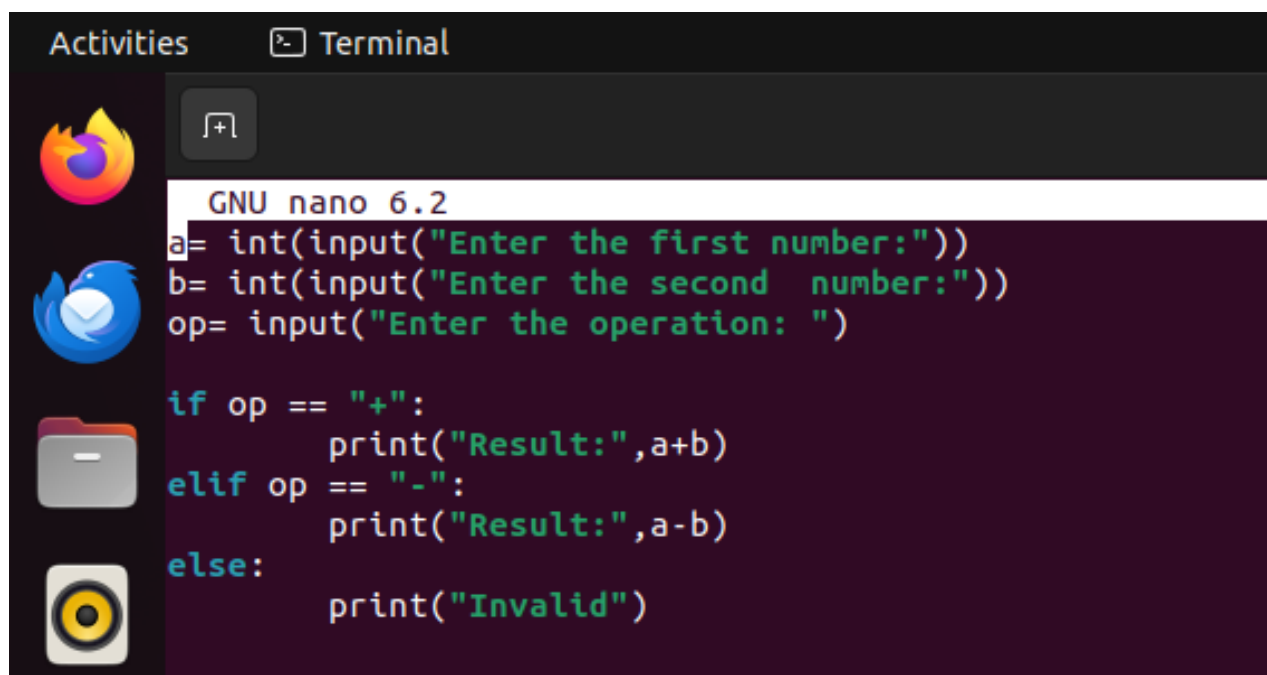
4. Show an example of a for loop in bash. Then, write a simple bash calculator that does.

For loop:

Calculator:

```
saleh@saleh-VirtualBox:~$ nano calc.py
saleh@saleh-VirtualBox:~$ python3 calc.py
Enter the first number:5
Enter the second  number:2
Enter the operation: +
Result: 7
saleh@saleh-VirtualBox:~$ python3 calc.py
Enter the first number:45
Enter the second  number:6
Enter the operation: -
Result: 39
saleh@saleh-VirtualBox:~$ 4554
4554: command not found
saleh@saleh-VirtualBox:~$ python3 calc.py
Enter the first number:55
Enter the second  number:99
Enter the operation: *
Invalid
saleh@saleh-VirtualBox:~$
```