Day 5 – Phase 5: Scripting Automation, Redirection & FDs

• Set an environment variable for sensor type.

```
salma2002@MSI:~$ export SENSOR_TYPE=temperature
salma2002@MSI:~$ echo SENSOR_TYPE
SENSOR_TYPE
salma2002@MSI:~$ echo $SENSOR_TYPE
temperature
salma2002@MSI:~$
```

• Write scripts/sensor_script.py to simulate data logging (timestamps + random values).

```
salma2002@MSI:~$ ls
Desktop  Documents  Downloads  iot_logger  new  sensor_poll.sh  yes  yes.pub
salma2002@MSI:~$ cd iot_logger/
salma2002@MSI:~/iot_logger$ cd scripts/
salma2002@MSI:~/iot_logger/scripts$ touch sensor_script.py
touch: cannot touch 'sensor_script.py': Permission denied
salma2002@MSI:~/iot_logger/scripts$ sudo touch sensor_script
.py
[sudo] password for salma2002:
salma2002@MSI:~/iot_logger/scripts$ nano sensor_script.py
```

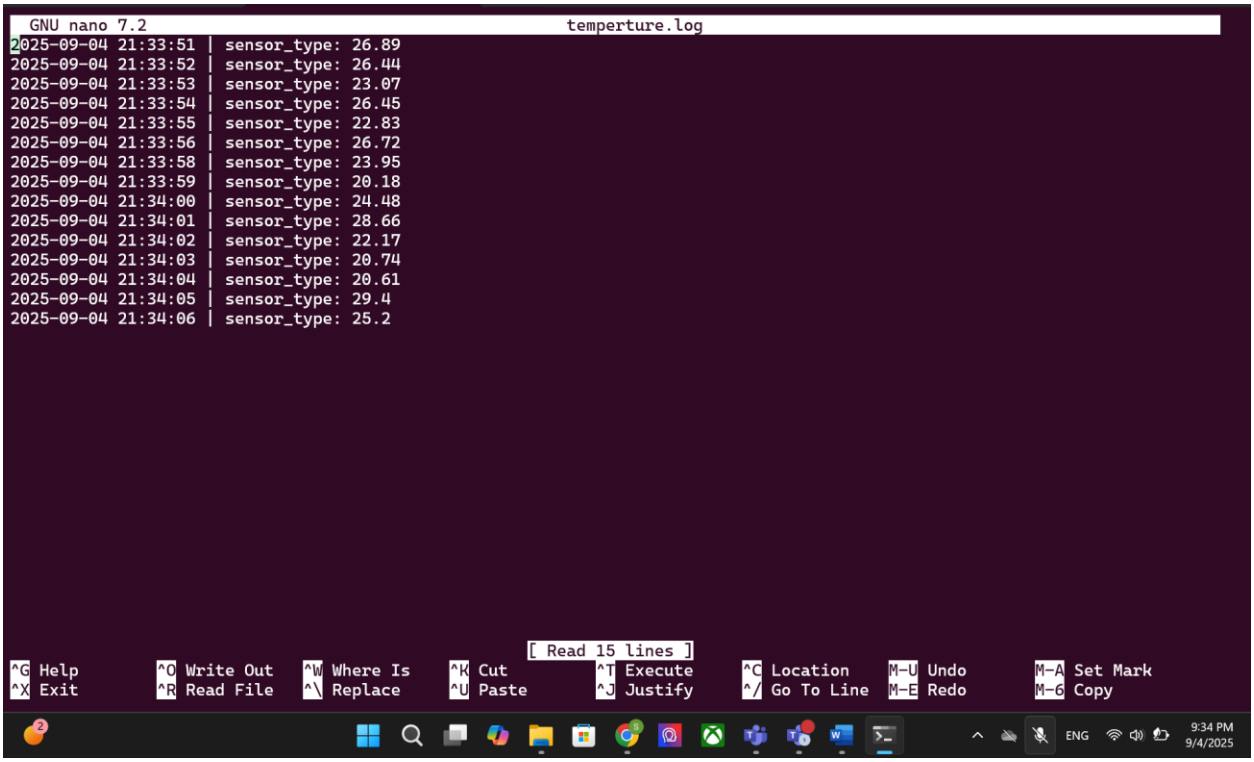
```
salma2002@MSI:~/iot_logger/scripts$ chmod +x scripts/sensor_script.py
chmod: cannot access 'scripts/sensor_script.py': No such file or directory
salma2002@MSI:~/iot_logger/scripts$ sudo chmod +x sensor_scr
ipt.py
salma2002@MSI:~/iot_logger/scripts$ nano sensor_script.py
salma2002@MSI:~/iot_logger/scripts$
```

```
  GNU nano 7.2                                    sensor_script.py
#!/usr/bin/env python3
import time, random, sys


while True:
    value = round(random.uniform(20.0, 30.0), 2)  # simulate sensor data
    timestamp = time.strftime("%Y-%m-%d %H:%M:%S")
    print(f"{timestamp} | {sensor_type}: {value}")
    sys.stdout.flush()
    time.sleep(1)
```

• Redirect script output to logs/temperature.log while running as a background process.

```
salma2002@MSI:~/iot_logger/scripts$ cd ../logs
salma2002@MSI:~/iot_logger/logs$ ../scripts/sensor_script.py >> temperture.log &
[1] 842
salma2002@MSI:~/iot_logger/logs$ ls
temperture.log  temperture_hard.log  temperture_soft.log
salma2002@MSI:~/iot_logger/logs$ nano temperture.log
salma2002@MSI:~/iot_logger/logs$
```



- **Find the PID of the process, inspect file descriptors in /proc//fd.**

```
salma2002@MSI:~/iot_logger/logs$ ../scripts/sensor_script.py >> temperture.log &
[1] 842
salma2002@MSI:~/iot_logger/logs$ ls
temperture.log  temperture_hard.log  temperture_soft.log
salma2002@MSI:~/iot_logger/logs$ nano temperture.log
salma2002@MSI:~/iot_logger/logs$ echo $!
842
salma2002@MSI:~/iot_logger/logs$
```

```
salma2002@MSI:~/iot_logger/logs$ echo $!
842
```

```
salma2002@MSI:~/iot_logger/logs$ ls -l /proc/842/fd
total 0
lrwx------ 1 salma2002 salma2002 64 Sep  4 21:52 0 -> /dev/pts/0
l-wx------ 1 salma2002 salma2002 64 Sep  4 21:52 1 -> /home/salma2002/iot_logger/logs/temperture.log
lrwx------ 1 salma2002 salma2002 64 Sep  4 21:52 2 -> /dev/pts/0
salma2002@MSI:~/iot_logger/logs$
```

- **Filter log data into another file.**

```
salma2002@MSI:~/iot_logger/logs$ touch high_temp.log
salma2002@MSI:~/iot_logger/logs$ ls
high_temp.log   temperture_hard.log
temperture.log  temperture_soft.log
salma2002@MSI:~/iot_logger/logs$ awk -F": " '$2 > 25' temperature.log > high_temp.log
awk: fatal: cannot open file `temperature.log' for reading: No such file or directory
salma2002@MSI:~/iot_logger/logs$ awk -F": " '$2 > 25' temperture.log > high_temp.log
salma2002@MSI:~/iot_logger/logs$ nano high_temp.log
```

```
 GNU nano 7.2                                                high_temp.log
2025-09-04 21:33:51 | sensor_type: 26.89
2025-09-04 21:33:52 | sensor_type: 26.44
2025-09-04 21:33:54 | sensor_type: 26.45
2025-09-04 21:33:56 | sensor_type: 26.72
2025-09-04 21:34:01 | sensor_type: 28.66
2025-09-04 21:34:05 | sensor_type: 29.4
2025-09-04 21:34:06 | sensor_type: 25.2
2025-09-04 21:34:30 | sensor_type: 29.17
2025-09-04 21:34:31 | sensor_type: 25.4
2025-09-04 21:34:34 | sensor_type: 29.18
2025-09-04 21:34:35 | sensor_type: 26.82
2025-09-04 21:34:37 | sensor_type: 29.94
2025-09-04 21:34:38 | sensor_type: 27.95
2025-09-04 21:34:42 | sensor_type: 28.19
2025-09-04 21:34:43 | sensor_type: 26.54
2025-09-04 21:34:44 | sensor_type: 25.69
2025-09-04 21:34:47 | sensor_type: 25.56
2025-09-04 21:34:49 | sensor_type: 28.98
2025-09-04 21:34:50 | sensor_type: 25.28
2025-09-04 21:34:52 | sensor_type: 26.35
2025-09-04 21:34:54 | sensor_type: 26.08
2025-09-04 21:34:55 | sensor_type: 27.71
2025-09-04 21:34:57 | sensor_type: 29.32
2025-09-04 21:35:02 | sensor_type: 29.42
2025-09-04 21:35:03 | sensor_type: 29.15
2025-09-04 21:35:11 | sensor_type: 28.13
2025-09-04 21:35:13 | sensor_type: 25.46
2025-09-04 21:35:14 | sensor_type: 29.24
2025-09-04 21:35:15 | sensor_type: 29.5
2025-09-04 21:35:16 | sensor_type: 26.21
2025-09-04 21:35:17 | sensor_type: 29.64
```

• Use wildcards to copy logs to data/.

```
salma2002@MSI:~/iot_logger/logs$ cp *.log data/
cp: target 'data/': No such file or directory
salma2002@MSI:~/iot_logger/logs$ cp *.log ../data/
salma2002@MSI:~/iot_logger/logs$ cd ../data
salma2002@MSI:~/iot_logger/data$ ls
high_temp.log  services  temperture.log  temperture_hard.log  temperture_soft.log
salma2002@MSI:~/iot_logger/data$ cd ../logs
salma2002@MSI:~/iot_logger/logs$ ls
high_temp.log  temperture.log  temperture_hard.log  temperture_soft.log
salma2002@MSI:~/iot_logger/logs$
```

• Clear variable when done.

```
salma2002@MSI:~/iot_logger/logs$ unset SENSOR_TYPE
salma2002@MSI:~/iot_logger/logs$
```

Challenge – Pipes & FD inspection :

• Run a pipeline (e.g., ls -l | grep .py).

```
salma2002@MSI:~/iot_logger/logs$ (ls -l; sleep 30) | grep .py
salma2002@MSI:~/iot_logger/logs$
```

• While it's running, inspect the FDs in /proc//fd.

```
salma2002@MSI:~$ ps -fC ls
UID          PID    PPID  C STIME TTY          TIME CMD
salma2002@MSI:~$ ps -fC grep
UID          PID    PPID  C STIME TTY          TIME CMD
salma20+     930     539  0 22:10 pts/0    00:00:00 grep --c
salma2002@MSI:~$ ls -l /proc/930/fd
total 0
lr-x------ 1 salma2002 salma2002 64 Sep  4 22:10 0 -> 'pipe:[13140]'
lrwx------ 1 salma2002 salma2002 64 Sep  4 22:10 1 -> /dev/pts/0
lrwx------ 1 salma2002 salma2002 64 Sep  4 22:10 2 -> /dev/pts/0
salma2002@MSI:~$
```

• Hint: To give yourself time, put a sleep in one command of the pipeline so the process stays alive long enough for inspection.

Open-Ended Questions:

• What's the difference between ' ' and " " in shell?

- ' ' → literal, no variable/command expansion.

  echo 'Hello $USER' → Hello $USER

- " " → expands variables/commands.

  echo "Hello $USER" → Hello salma2002

• Explain [-f filename] vs [-d dirname].

[-f file] → true if file exists and is a regular file.

[-d dir] → true if path exists and is a directory.

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

What are stdout and stderr?

- In Linux, everything is a file, even input/output.
- A process has 3 default file descriptors (FDs):
  - 0 = stdin (keyboard input)

- 1 = stdout (normal output, like results of ls)
- 2 = stderr (error messages, like “No such file or directory”)

Redirection

- > → send stream to a file (overwrite).
- >> → send stream to a file (append).
- By default, > means **stdout (1)**.