

```
Activities Terminal 19:06 2 سبت
noor@NOOR: ~/iot_logger/scripts
noor@NOOR:~/iot_logger/scripts$ nano sensor.sh
noor@NOOR:~/iot_logger/scripts$ chmod +x sensor.sh
noor@NOOR:~/iot_logger/scripts$ ./sensor.sh &
[1] 4598
noor@NOOR:~/iot_logger/scripts$ ps aux | grep sensor.sh
noor      4598  0.0  0.0 10108 3840 pts/0    S   18:48   0:00 /bin/bash ./sensor.sh
noor      4611  0.0  0.0  9224 2560 pts/0    S+  18:48   0:00 grep --color=auto sensor.sh
noor@NOOR:~/iot_logger/scripts$ tail -f sensor.sh
#!/bin/bash
log="$HOME/iot_logger/logs/temperature.log"
while true; do
    echo "$(date +%F\ %T) - Fake temp:$((RANDOM % 20 +20)) c" >> "$log"
    sleep 5
done
^C
noor@NOOR:~/iot_logger/scripts$ tail -f ~/iot_logger/logs/temperature.log
2025-09-02 18:57:02 - Fake temp:21 c
2025-09-02 18:57:07 - Fake temp:36 c
2025-09-02 18:57:12 - Fake temp:31 c
2025-09-02 18:57:17 - Fake temp:28 c
2025-09-02 18:57:22 - Fake temp:33 c
2025-09-02 18:57:27 - Fake temp:34 c
2025-09-02 18:57:32 - Fake temp:28 c
```

```
2025-09-02 18:58:07 - Fake temp:31 c
^C
noor@NOOR:~/iot_logger/scripts$ ps aux | grep ~/iot_logger/logs/temperature.log
noor      4932  0.0  0.0  9248 2560 pts/0    S+  19:01   0:00 grep --color=auto /home/noor
/iot_logger/logs/temperature.log
noor@NOOR:~/iot_logger/scripts$ jobs
[1]+  Running                  ./sensor.sh &
noor@NOOR:~/iot_logger/scripts$ kill -9 ^C
noor@NOOR:~/iot_logger/scripts$ kill -9 4598
noor@NOOR:~/iot_logger/scripts$ ps aux | grep ~/iot_logger/logs/temperature.log
noor      5032  0.0  0.0  9248 2560 pts/0    S+  19:05   0:00 grep --color=auto /home/noor
/iot_logger/logs/temperature.log
[1]+  Killed                  ./sensor.sh
noor@NOOR:~/iot_logger/scripts$
```

```
Activities Terminal 21:33 سبت 21
noor@NOOR: ~/iot_logger/scripts
noor@NOOR:~/iot_logger/scripts$ ./sensor.sh &
[1] 5294
noor@NOOR:~/iot_logger/scripts$ ps aux | grep sensor.sh
noor      5294  0.0  0.0 10108 3840 pts/0    S   21:23   0:00 /bin/bash ./sensor.sh
noor      5306  0.0  0.0  9224 2560 pts/0    S+  21:23   0:00 grep --color=auto sensor.sh
noor@NOOR:~/iot_logger/scripts$ netstat | grep ESTABLISHED
udp      0      0 NOOR:bootpc          _gateway:bootps      ESTABLISHED
noor@NOOR:~/iot_logger/scripts$ jobs
[1]+  Running                  ./sensor.sh &
noor@NOOR:~/iot_logger/scripts$ fg %1
Rhythmbox
^Z
[1]+  Stopped                  ./sensor.sh
noor@NOOR:~/iot_logger/scripts$ bg %1
[1]+ ./sensor.sh &
noor@NOOR:~/iot_logger/scripts$ fg %1
./sensor.sh
^Z
[1]+  Stopped                  ./sensor.sh
noor@NOOR:~/iot_logger/scripts$ kill -9 5294
noor@NOOR:~/iot_logger/scripts$ ps aux | grep sensor.sh
noor      5502  0.0  0.0  9224 2560 pts/0    S+  21:32   0:00 grep --color=auto sensor.sh
[1]+  Killed                   ./sensor.sh
noor@NOOR:~/iot_logger/scripts$
```

**What happens step by step when you type a command in bash (e.g., ls) until you see the output?**

When I type for ex: ls

1. The bash read the input
2. check the syntax and look up for it
3. bash searches for it in the directories if it's not a built in in the shell
4. bash create a new process using fork () it makes a child process
5. child process execute ls
6. kernel runs ls
7. the results appear in the terminal

**Explain the types of processes in Linux: daemon, zombie, orphan. How can you detect them?**

Daemon process: long running process in the background to respond to request from the services

Orphan process: when the parent process end first while the process is still running  
And it gets adopted by the system

Zombie process: when the child process has terminated execution but remain in the process table list and its parent hasn't read its exit status

**Why do we need Inter-Process Communication (IPC)? List some IPC mechanisms and real-life examples.**

IPC (inter process communication): allows processes to exchange data and it help processes synchronize their activities

Types of IPC:

sockets: network communication method and it helps connect the processes

Pipes: one process writes the other read ex: `ls | grep txt`

Message queues: processes send or receive structured messages

Signal: control processes ex: `kill -9 DIP` send a SIGKILL

Ex: web browser each tab runs as a separate process but when the processes need to communicate with each other they use IPC (sockets ) to exchange data and send messages , we also use it with databases using the shared memory