

# Capstone Task Day 02

Name: Ibrahim Shnouda

Group: IoT701-A

Facilitator: Eng.Ehab Elsayed

---

## Task:

1.\$ echo "polling sensor" sleep 200 &

2.\$ ps aux | grep <PID>

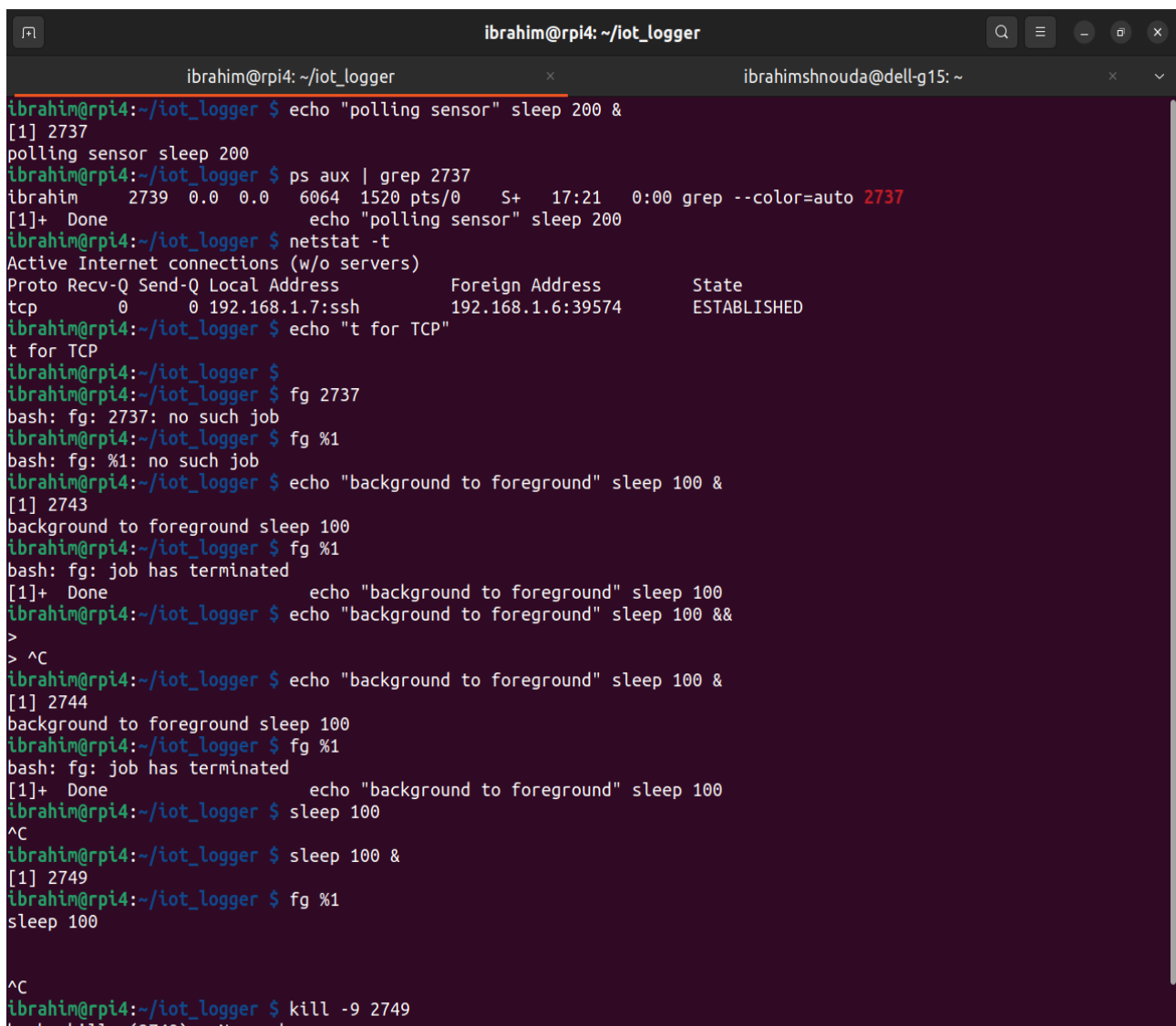
3.\$ netstat -t

4.\$ sleep 100

\$ fg <%job number>

5.\$ kill -9 <PID>

## Screenshot:



```
ibrahim@rpi4: ~/iot_logger
ibrahim@rpi4:~/iot_logger $ echo "polling sensor" sleep 200 &
[1] 2737
polling sensor sleep 200
ibrahim@rpi4:~/iot_logger $ ps aux | grep 2737
ibrahim 2739 0.0 0.0 6064 1520 pts/0 S+ 17:21 0:00 grep --color=auto 2737
[1]+ Done echo "polling sensor" sleep 200
ibrahim@rpi4:~/iot_logger $ netstat -t
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address Foreign Address State
tcp 0 0 192.168.1.7:ssh 192.168.1.6:39574 ESTABLISHED
ibrahim@rpi4:~/iot_logger $ echo "t for TCP"
t for TCP
ibrahim@rpi4:~/iot_logger $
ibrahim@rpi4:~/iot_logger $ fg 2737
bash: fg: 2737: no such job
ibrahim@rpi4:~/iot_logger $ fg %1
bash: fg: %1: no such job
ibrahim@rpi4:~/iot_logger $ echo "background to foreground" sleep 100 &
[1] 2743
background to foreground sleep 100
ibrahim@rpi4:~/iot_logger $ fg %1
bash: fg: job has terminated
[1]+ Done echo "background to foreground" sleep 100
ibrahim@rpi4:~/iot_logger $ echo "background to foreground" sleep 100 &&
>
> ^C
ibrahim@rpi4:~/iot_logger $ echo "background to foreground" sleep 100 &
[1] 2744
background to foreground sleep 100
ibrahim@rpi4:~/iot_logger $ fg %1
bash: fg: job has terminated
[1]+ Done echo "background to foreground" sleep 100
ibrahim@rpi4:~/iot_logger $ sleep 100
^C
ibrahim@rpi4:~/iot_logger $ sleep 100 &
[1] 2749
ibrahim@rpi4:~/iot_logger $ fg %1
sleep 100
^C
ibrahim@rpi4:~/iot_logger $ kill -9 2749
bash: kill: (2749): No such process
```

## **Open-Ended Questions:**

**1.shell parsing your prompt and the first argument (program) execute as child process to bash process using (fork()) System call and if this program need to options uses options the print output to terminal or file dependent on file descriptor.**

### **2.Daemon, Orphan, and Zombie**

**2.1.Daemon Process: Process running in the background without any GUI**  
**ps -ef | grep daemon**

**2.2.Orphan Process: is a process their parent dead or terminated and become with no parent, new parent will be systemd or init process**  
**ps -ef | grep <PID> if PPID = 1 : this orphan process**

**2.3.Zombie process: is a process their parent don't use wait() syscall to take child process status then terminate after done, process Is not killed after finishing**  
**ps aux | grep 'Z'**

**3.IPC is the way to allow processes to communicate to each other while running like pipes() to share data and memory sharing and sockets .**