

Name: Basant Tarik Salah

Instructor: Eng. Mohamed Abo-Khalil

SIC7_Task.Phase4

Part1: Tasks

1. Run a background task to simulate sensor polling.

```
basant@basant-VirtualBox:~$ while true
do
  echo "Checking sensor..."
  sleep 25
done &
[1] 2857
basant@basant-VirtualBox:~$ Checking sensor...
Checking sensor...
ls
Desktop Documents Downloads gitdemo lot_logger Music Pictures Public snap Templates Videos
basant@basant-VirtualBox:~$ date
02 2025 , السبت 08:02:52 م
basant@basant-VirtualBox:~$ Checking sensor...
```

I used some commands like ls and date to ensure the terminal works while the process is running in the background.

2. List processes and filter for the background task.

```
basant@basant-VirtualBox:~$ ps aux | grep sleep
basant      2872  0.0  0.0   9216  2560 pts/0    R+   20:04   0:00 grep --color=auto sleep
[1]+  Terminated                  while true; do
  echo "Checking sensor..."; sleep 25;
done
basant@basant-VirtualBox:~$
```

3. Check network states (established connections).

```
basant@basant-VirtualBox:~$ ss -tulnp
```

Netid	State	Recv-Q	Send-Q	Local Address:Port	Peer Address:Port	Process
udp	UNCONN	0	0	0.0.0.0:5353	0.0.0.0:*	
udp	UNCONN	0	0	0.0.0.0:39271	0.0.0.0:*	
udp	UNCONN	0	0	127.0.0.53%lo:53	0.0.0.0:*	
udp	UNCONN	0	0	:::36397	:::*	
udp	UNCONN	0	0	:::5353	:::*	
tcp	LISTEN	0	128	0.0.0.0:22	0.0.0.0:*	
tcp	LISTEN	0	128	127.0.0.1:631	0.0.0.0:*	
tcp	LISTEN	0	4096	127.0.0.53%lo:53	0.0.0.0:*	
tcp	LISTEN	0	128	:::22	:::*	
tcp	LISTEN	0	128	:::631	:::*	

```
basant@basant-VirtualBox:~$
```


Part 2: Open Ended Questions

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

- I write command and press Enter.
- Bash shell looks for the program in my path.
- The program runs.
- The kernel loads the program into the memory.
- The program executes and asks the kernel to read the directory contents.
- Kernel returns the data.
- The result shows on the screen.

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

- **Daemon:**

A process that runs in the background all the time to respond to requests from services.

They are detected by running **ps -ef** then they will be listed as names ending with d.

- **Zombie:**

When a child process has terminated execution but remains in the process table list. This process may appear as 'defunct Process' in the process list.

If the zombie process increases, the process table may run out of capacity. It can prevent regular processes from running.

They are detected by running **ps aux** and looking for processes with **STAT = Z** or the word **defunct**.

- **Orphan:**

A process whose parent died which means the parent ends first while the process still running, so it gets adopted by the init.

They are detected by running **ps -ef** and looking for the **parent PID = 1**.

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

To make two programs able to talk to each other and share data.

Examples:

- **Pipes:** One process output becomes another process input.
- **Sockets:** Used for communication between processes over networks.
- **Shared memory:** Many processes access the same memory area.
- **Message queues:** Send and receive messages between processes.