



National Textile University

Department of Computer Science

Subject:

Operating Systems

Submitted To:

Sir Nasir Mehmood

Submitted By:

Alishba Riasat

Registration No:

23-NTU-CS-1135

Lab No:

3

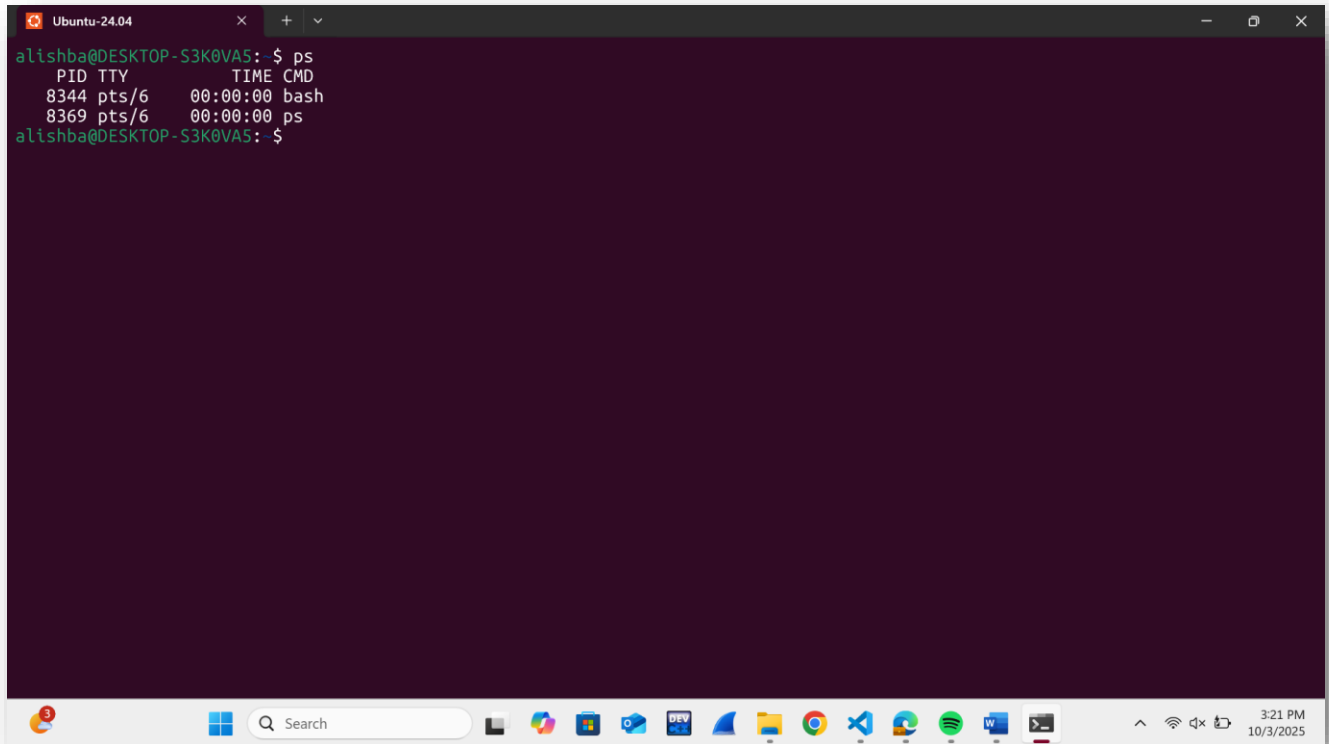
Semester:

5th

2. Linux Process Commands

2.1 Viewing Processes

ps → Process Status



A terminal window titled "Ubuntu-24.04" showing the output of the `ps` command. The prompt is `alishba@DESKTOP-S3K0VAS: $`. The output is a table with columns: PID, TTY, TIME, and CMD. It lists two processes: `bash` (PID 8344) and `ps` (PID 8369). The terminal is running on a desktop environment with a taskbar at the bottom showing various application icons and system status information (3:21 PM, 10/3/2025).

```
alishba@DESKTOP-S3K0VAS: $ ps
  PID TTY          TIME CMD
 8344 pts/6    00:00:00 bash
 8369 pts/6    00:00:00 ps
alishba@DESKTOP-S3K0VAS: $
```

ps-ef command

```
Ubuntu-24.04
alishba@DESKTOP-S3K0VAS:~$ ps -ef
UID          PID    PPID  C   TIME TTY          TIME CMD
root           1        0  0  14:27 ?        00:00:07 /sbin/init
root           2        1  0  14:27 ?        00:00:00 /init
root           7        2  0  14:27 ?        00:00:00 plan9 --control-socket 7 --log-level 4 --server-fd 8 --pipe-fd 10 --log-
root          52        1  0  14:27 ?        00:00:01 /usr/lib/systemd/systemd-journald
root         111        1  0  14:27 ?        00:00:01 /usr/lib/systemd/systemd-udev
systemd+     120        1  0  14:27 ?        00:00:00 /usr/lib/systemd/systemd-resolved
systemd+     124        1  0  14:27 ?        00:00:00 /usr/lib/systemd/systemd-timesyncd
root         185        1  0  14:27 ?        00:00:00 /usr/sbin/cron -f -P
message+     186        1  0  14:27 ?        00:00:00 @dbus-daemon --system --address=systemd: --nofork --nopidfile --systemd-
root         202        1  0  14:27 ?        00:00:00 /usr/lib/systemd/systemd-logind
root         204        1  0  14:27 ?        00:00:01 /usr/libexec/wsl-pro-service -vv
root         216        1  0  14:27 hvc0     00:00:00 /sbin/agetty -o -p -- \u --noclear --keep-baud - 115200,38400,9600 vt220
syslog       224        1  0  14:27 ?        00:00:01 /usr/sbin/rsyslogd -n -iNONE
root         230        1  0  14:27 tty1     00:00:00 /sbin/agetty -o -p -- \u --noclear - linux
root         244        1  0  14:27 ?        00:00:00 /usr/bin/python3 /usr/share/unattended-upgrades/unattended-upgrade-shutd
root         298        2  0  14:27 pts/1    00:00:00 /bin/login -f
alishba      382        1  0  14:27 ?        00:00:00 /usr/lib/systemd/systemd --user
alishba      385       382  0  14:27 ?        00:00:00 (sd-pam)
alishba      413       298  0  14:28 pts/1    00:00:00 -bash
root         859        2  0  14:37 ?        00:00:00 /init
root         860       859  0  14:37 ?        00:00:00 /init
alishba      861       860  0  14:37 pts/0    00:00:00 sh -c "$VSCODE_WSL_EXT_LOCATION/scripts/wslServer.sh" e3a5acfb517a443235
alishba      862       861  0  14:37 pts/0    00:00:00 sh /mnt/c/Users/HP/.vscode/extensions/ms-vscode-remote.remote-wsl-0.104.
alishba      868       862  0  14:37 pts/0    00:00:00 sh /home/alishba/.vscode-server/bin/e3a5acfb517a443235981655413d56653310
alishba      872       868  0  14:37 pts/0    00:00:15 /home/alishba/.vscode-server/bin/e3a5acfb517a443235981655413d566533107e9
root         883        2  0  14:37 ?        00:00:00 /init
root         884       883  0  14:37 ?        00:00:00 /init
alishba      885       884  0  14:37 pts/2    00:00:03 /home/alishba/.vscode-server/bin/e3a5acfb517a443235981655413d566533107e9
root         900        2  0  14:37 ?        00:00:00 /init
root         902       900  0  14:37 ?        00:00:00 /init
```

ps-ef | grep bash

```
Ubuntu-24.04
alishba 382 1 0 14:27 ? 00:00:00 /usr/lib/systemd/systemd --user
alishba 385 382 0 14:27 ? 00:00:00 (sd-pam)
alishba 413 298 0 14:28 pts/1 00:00:00 -bash
root 859 2 0 14:37 ? 00:00:00 /init
root 860 859 0 14:37 ? 00:00:00 /init
alishba 861 860 0 14:37 pts/0 00:00:00 sh -c "$VS_CODE_WSL_EXT_LOCATION/scripts/wslServer.sh" e3a5acfb517a443235
alishba 862 861 0 14:37 pts/0 00:00:00 sh /mnt/c/Users/HP/.vscode/extensions/ms-vscode-remote.remote-wsl-0.104.
alishba 868 862 0 14:37 pts/0 00:00:00 sh /home/alishba/.vscode-server/bin/e3a5acfb517a443235981655413d56653310
alishba 872 868 0 14:37 pts/0 00:00:15 /home/alishba/.vscode-server/bin/e3a5acfb517a443235981655413d566533107e9
root 883 2 0 14:37 ? 00:00:00 /init
root 884 883 0 14:37 ? 00:00:00 /init
alishba 885 884 0 14:37 pts/2 00:00:03 /home/alishba/.vscode-server/bin/e3a5acfb517a443235981655413d566533107e9
root 900 2 0 14:37 ? 00:00:00 /init
root 902 900 0 14:37 ? 00:00:00 /init
alishba 907 902 0 14:37 pts/3 00:00:03 /home/alishba/.vscode-server/bin/e3a5acfb517a443235981655413d566533107e9
alishba 918 872 0 14:37 pts/0 00:00:01 /home/alishba/.vscode-server/bin/e3a5acfb517a443235981655413d566533107e9
alishba 934 872 3 14:37 pts/0 00:01:39 /home/alishba/.vscode-server/bin/e3a5acfb517a443235981655413d566533107e9
alishba 1115 872 0 14:38 pts/0 00:00:10 /home/alishba/.vscode-server/bin/e3a5acfb517a443235981655413d566533107e9
alishba 6583 1115 0 15:07 pts/4 00:00:00 /bin/bash --init-file /home/alishba/.vscode-server/bin/e3a5acfb517a44323
root 8340 2 0 15:20 ? 00:00:00 /init
root 8341 8340 0 15:20 ? 00:00:00 /init
alishba 8344 8341 0 15:20 pts/6 00:00:00 -bash
alishba 8625 8344 0 15:22 pts/6 00:00:00 ps -ef
alishba@DESKTOP-S3K0VAS: $ ps-ef | grep bash
ps-ef: command not found
alishba@DESKTOP-S3K0VAS: $ ps -ef | grep bash
alishba 413 298 0 14:28 pts/1 00:00:00 -bash
alishba 6583 1115 0 15:07 pts/4 00:00:00 /bin/bash --init-file /home/alishba/.vscode-server/bin/e3a5acfb517a44323
5981655413d566533107e92/out/vs/workbench/contrib/terminal/common/scripts/shellIntegration-bash.sh
alishba 8344 8341 0 15:20 pts/6 00:00:00 -bash
alishba 8886 8344 0 15:24 pts/6 00:00:00 grep --color=auto bash
alishba@DESKTOP-S3K0VAS: $
```

ps-ef | grep alishba

```
Ubuntu-24.04
alishba@DESKTOP-S3K0VAS: $ ps -ef | grep alishba
alishba      382      1  0 14:27 ?        00:00:00 /usr/lib/systemd/systemd --user
alishba      385     382  0 14:27 ?        00:00:00 (sd-pam)
alishba      413     298  0 14:28 pts/1      00:00:00 -bash
alishba      861     860  0 14:37 pts/0      00:00:00 sh -c "SVSCODE_WSL_EXT_LOCATION/scripts/wslServer.sh" e3a5acfb517a443235
981655413d566533107e92 stable code-server .vscode-server --host=127.0.0.1 --port=0 --connection-token=22657172-276610082-116
4544043-666262966 --use-host-proxy --without-browser-env-var --disable-websocket-compression --accept-server-license-terms --
-telemetry-level=all
alishba      862     861  0 14:37 pts/0      00:00:00 sh /mnt/c/Users/HP/.vscode/extensions/ms-vscode-remote.remote-wsl-0.104.
2/scripts/wslServer.sh e3a5acfb517a443235981655413d566533107e92 stable code-server .vscode-server --host=127.0.0.1 --port=0
--connection-token=22657172-276610082-1164544043-666262966 --use-host-proxy --without-browser-env-var --disable-websocket-co
mpression --accept-server-license-terms --telemetry-level=all
alishba      868     862  0 14:37 pts/0      00:00:00 sh /home/alishba/.vscode-server/bin/e3a5acfb517a443235981655413d56653310
7e92/bin/code-server --host=127.0.0.1 --port=0 --connection-token=22657172-276610082-1164544043-666262966 --use-host-proxy --
without-browser-env-var --disable-websocket-compression --accept-server-license-terms --telemetry-level=all
alishba      872     868  0 14:37 pts/0      00:00:16 /home/alishba/.vscode-server/bin/e3a5acfb517a443235981655413d566533107e9
2/node /home/alishba/.vscode-server/bin/e3a5acfb517a443235981655413d566533107e92/out/server-main.js --host=127.0.0.1 --port=
0 --connection-token=22657172-276610082-1164544043-666262966 --use-host-proxy --without-browser-env-var --disable-websocket-
compression --accept-server-license-terms --telemetry-level=all
alishba      885     884  0 14:37 pts/2      00:00:03 /home/alishba/.vscode-server/bin/e3a5acfb517a443235981655413d566533107e9
2/node -e const net = require('net'); process.stdin.pause(); const client = net.createConnection({ host: '127.0.0.1', port:
46023 }, () => { client.pipe(process.stdout); process.stdin.pipe(client); }); client.on('close', function (hadError) { conso
le.error(hadError ? 'Remote close with error' : 'Remote close'); process.exit(hadError ? 1 : 0); }); client.on('error', func
tion (err) { process.stderr.write(err && (err.stack || err.message) || String(err)); });
alishba      907     902  0 14:37 pts/3      00:00:03 /home/alishba/.vscode-server/bin/e3a5acfb517a443235981655413d566533107e9
2/node -e const net = require('net'); process.stdin.pause(); const client = net.createConnection({ host: '127.0.0.1', port:
46023 }, () => { client.pipe(process.stdout); process.stdin.pipe(client); }); client.on('close', function (hadError) { conso
le.error(hadError ? 'Remote close with error' : 'Remote close'); process.exit(hadError ? 1 : 0); }); client.on('error', func
tion (err) { process.stderr.write(err && (err.stack || err.message) || String(err)); });
alishba      918     872  0 14:37 pts/0      00:00:01 /home/alishba/.vscode-server/bin/e3a5acfb517a443235981655413d566533107e9
2/node /home/alishba/.vscode-server/bin/e3a5acfb517a443235981655413d566533107e92/out/bootstrap-fork --type=fileWatcher
alishba      934     872  3 14:37 pts/0      00:01:41 /home/alishba/.vscode-server/bin/e3a5acfb517a443235981655413d566533107e9
```

2.2 Monitoring Processes Interactively

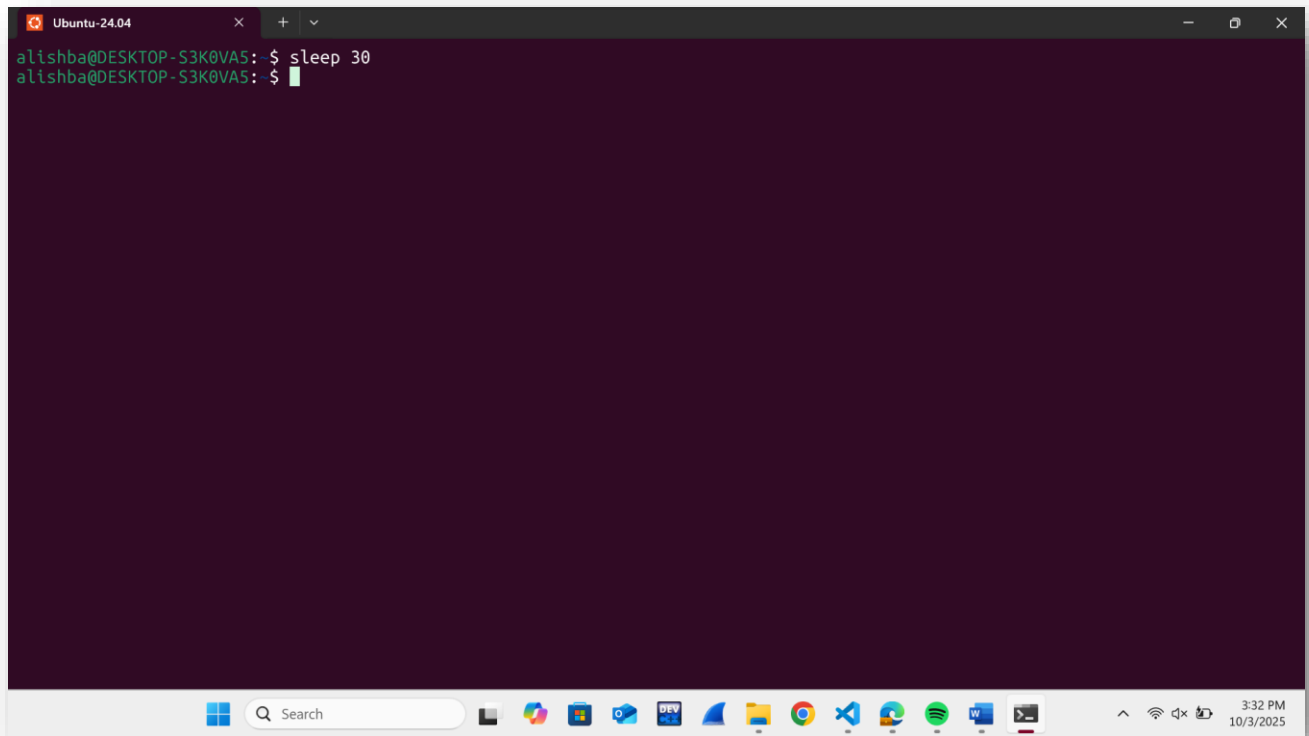
top command

```
alishba@DESKTOP-S3K0VA5: $ top
top - 15:29:24 up 1:02, 1 user, load average: 0.00, 0.00, 0.00
Tasks: 39 total, 1 running, 38 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.1 us, 0.1 sy, 0.0 ni, 99.5 id, 0.1 wa, 0.0 hi, 0.1 si, 0.0 st
MiB Mem : 3836.2 total, 3057.2 free, 727.1 used, 191.3 buff/cache
MiB Swap: 1024.0 total, 1024.0 free, 0.0 used, 3109.1 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
934	alishba	20	0	31.9g	136240	53504	S	1.0	3.5	1:44.57	node
1115	alishba	20	0	1134680	78136	48256	S	0.3	2.0	0:10.73	node
9542	alishba	20	0	9276	5376	3328	R	0.3	0.1	0:00.01	top
1	root	20	0	21752	12252	9180	S	0.0	0.3	0:07.46	systemd
2	root	20	0	3072	1664	1664	S	0.0	0.0	0:00.08	init-systemd(Ub
7	root	20	0	3120	1936	1920	S	0.0	0.0	0:00.48	init
52	root	19	-1	66812	16716	15820	S	0.0	0.4	0:01.68	systemd-journal
111	root	20	0	25404	6400	4864	S	0.0	0.2	0:01.67	systemd-udev
120	systemd+	20	0	21456	12544	10496	S	0.0	0.3	0:00.56	systemd-resolve
124	systemd+	20	0	91024	7680	6784	S	0.0	0.2	0:00.64	systemd-timesyn
185	root	20	0	4236	2432	2304	S	0.0	0.1	0:00.04	cron
186	message+	20	0	9632	4992	4480	S	0.0	0.1	0:00.38	dbus-daemon
202	root	20	0	17960	7936	7168	S	0.0	0.2	0:00.81	systemd-logind
204	root	20	0	1756096	13184	11008	S	0.0	0.3	0:01.62	wsl-pro-service
216	root	20	0	3160	1920	1792	S	0.0	0.0	0:00.06	agetty
224	syslog	20	0	222508	5760	4480	S	0.0	0.1	0:01.42	rsyslogd
230	root	20	0	3116	1792	1664	S	0.0	0.0	0:00.05	agetty
244	root	20	0	107032	22144	13056	S	0.0	0.6	0:00.59	unattended-upgr
298	root	20	0	6820	4352	3712	S	0.0	0.1	0:00.05	login
382	alishba	20	0	20312	11008	9088	S	0.0	0.3	0:00.67	systemd
385	alishba	20	0	21144	3512	1792	S	0.0	0.1	0:00.05	(sd-pam)
413	alishba	20	0	6072	4992	3456	S	0.0	0.1	0:00.11	bash
859	root	20	0	3088	1032	896	S	0.0	0.0	0:00.00	SessionLeader
860	root	20	0	3088	1168	1024	S	0.0	0.0	0:00.00	Relay(861)

2.3 Foreground and Background Job

Foreground:



The screenshot shows a Windows 11 desktop environment. A terminal window titled 'Ubuntu-24.04' is open, displaying the following text:

```
aishba@DESKTOP-S3K0VA5: $ sleep 30  
aishba@DESKTOP-S3K0VA5: $
```

The terminal window has a dark purple background. The Windows taskbar is visible at the bottom, featuring the Start button, a search bar, and several application icons including File Explorer, Microsoft Edge, and Spotify. The system tray on the right shows the time as 3:32 PM and the date as 10/3/2025.

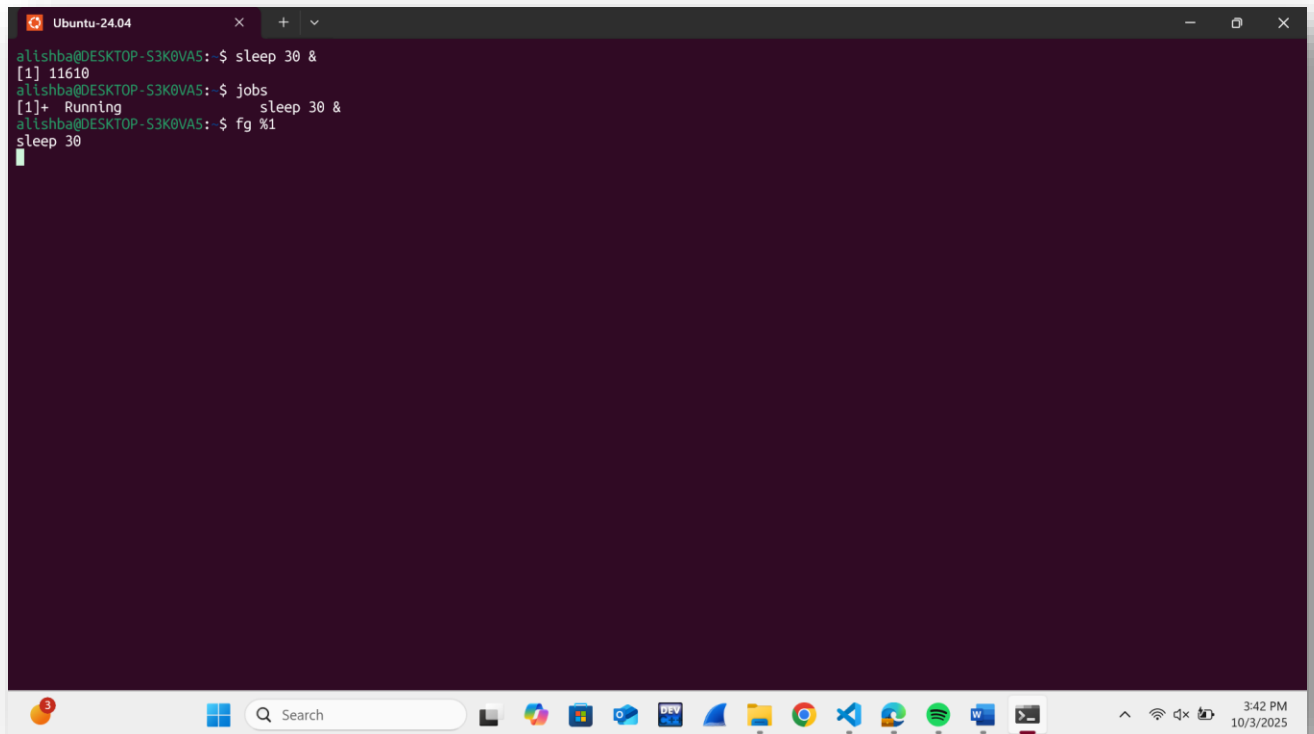
Background:

```
Ubuntu-24.04
alishba@DESKTOP-S3K0VA5: $ sleep 30
alishba@DESKTOP-S3K0VA5: $ sleep 30 &
[1] 10125
alishba@DESKTOP-S3K0VA5: $ top
top - 15:38:56 up 1:11, 1 user, load average: 0.01, 0.02, 0.00
Tasks: 39 total, 1 running, 38 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.1 us, 0.1 sy, 0.0 ni, 99.6 id, 0.0 wa, 0.0 hi, 0.1 si, 0.0 st
MiB Mem : 3836.2 total, 3048.5 free, 734.8 used, 192.4 buff/cache
MiB Swap: 1024.0 total, 1024.0 free, 0.0 used, 3101.4 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
934	alishba	20	0	31.9g	136240	53504	S	1.0	3.5	1:52.36	node
1115	alishba	20	0	1134680	78136	48256	S	0.3	2.0	0:11.24	node
10184	alishba	20	0	9276	5504	3456	R	0.3	0.1	0:00.16	top
1	root	20	0	21752	12252	9180	S	0.0	0.3	0:07.47	systemd
2	root	20	0	3072	1664	1664	S	0.0	0.0	0:00.08	init-systemd(Ub
7	root	20	0	3120	1936	1920	S	0.0	0.0	0:00.48	init
52	root	19	-1	66812	16716	15820	S	0.0	0.4	0:01.69	systemd-journal
111	root	20	0	25404	6400	4864	S	0.0	0.2	0:01.79	systemd-udev
120	systemd+	20	0	21456	12544	10496	S	0.0	0.3	0:00.56	systemd-resolve
124	systemd+	20	0	91024	7680	6784	S	0.0	0.2	0:00.65	systemd-timesyn
185	root	20	0	4236	2432	2304	S	0.0	0.1	0:00.04	cron
186	message+	20	0	9632	4992	4480	S	0.0	0.1	0:00.38	dbus-daemon
202	root	20	0	17960	7936	7168	S	0.0	0.2	0:00.81	systemd-logind
204	root	20	0	1756096	13184	11008	S	0.0	0.3	0:01.63	wsl-pro-service
216	root	20	0	3160	1920	1792	S	0.0	0.0	0:00.06	agetty
224	syslog	20	0	222508	5760	4480	S	0.0	0.1	0:01.42	rsyslogd
230	root	20	0	3116	1792	1664	S	0.0	0.0	0:00.05	agetty
244	root	20	0	107032	22144	13056	S	0.0	0.6	0:00.59	unattended-upgr
298	root	20	0	6820	4352	3712	S	0.0	0.1	0:00.05	login
382	alishba	20	0	20312	11008	9088	S	0.0	0.3	0:00.67	systemd
385	alishba	20	0	21144	3512	1792	S	0.0	0.1	0:00.05	(sd-pam)
413	alishba	20	0	6072	4992	3456	S	0.0	0.1	0:00.11	bash
859	root	20	0	3088	1032	896	S	0.0	0.0	0:00.00	SessionLeader
860	root	20	0	3088	1168	1024	S	0.0	0.0	0:00.00	Relay(861)
861	alishba	20	0	2800	1664	1664	S	0.0	0.0	0:00.02	sh

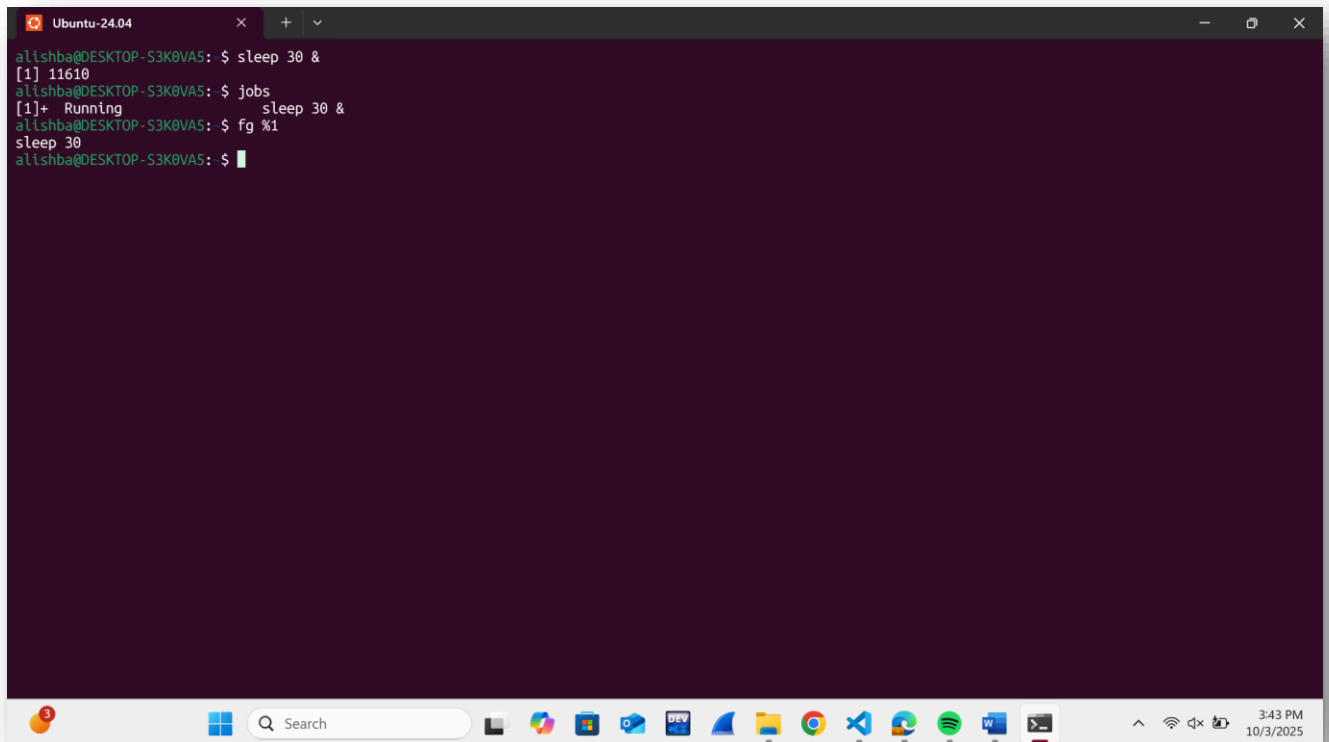
```
[1]+ Done
sleep 30
```


Check Background Jobs:

A terminal window titled 'Ubuntu-24.04' with a dark purple background. It shows a series of commands and their outputs. The user runs 'sleep 30 &', which returns '[1] 11610'. Then they run 'jobs', which shows '[1]+ Running sleep 30 &'. Next, they run 'fg %1', which returns 'sleep 30'. The terminal is part of a desktop environment with a taskbar at the bottom containing various application icons and system status indicators.

```
alishba@DESKTOP-S3K0VAS: $ sleep 30 &
[1] 11610
alishba@DESKTOP-S3K0VAS: $ jobs
[1]+  Running      sleep 30 &
alishba@DESKTOP-S3K0VAS: $ fg %1
sleep 30
```

Bring a job to foreground



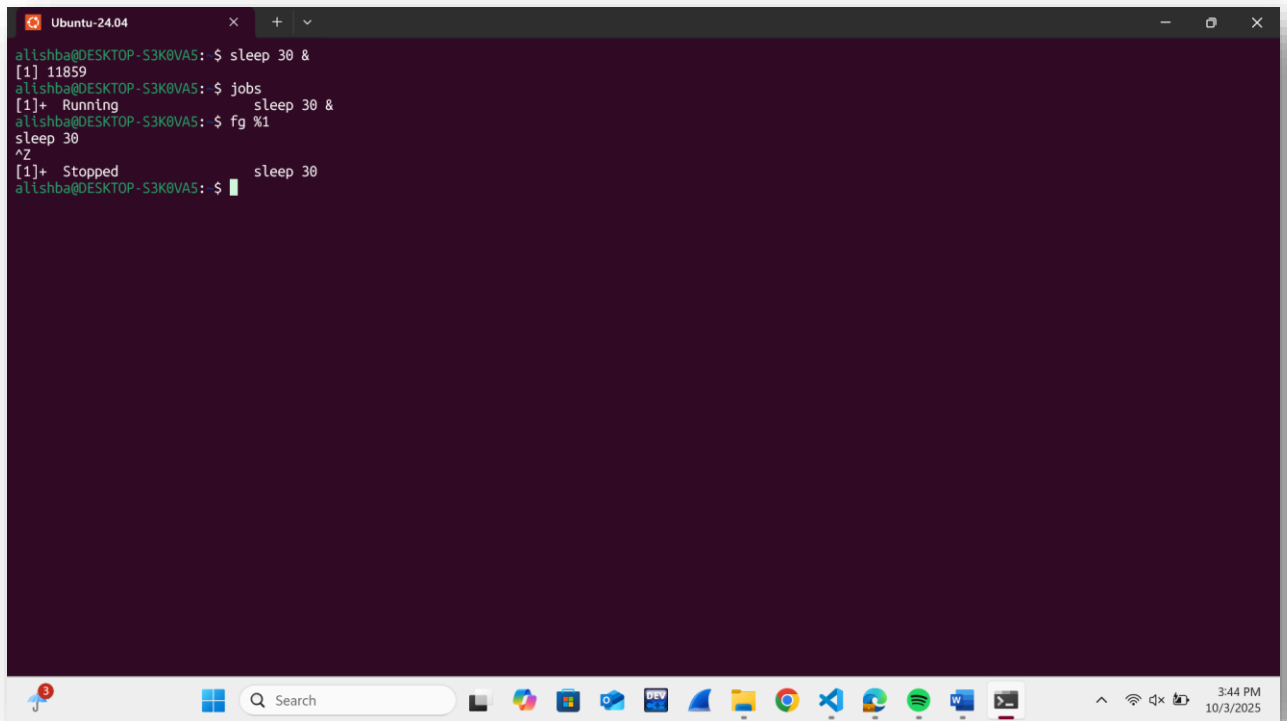
The image shows a terminal window titled "Ubuntu-24.04" with a dark purple background. The user, "alishba@DESKTOP-S3K0VAS", has executed the following commands:

```
alishba@DESKTOP-S3K0VAS: $ sleep 30 &  
[1] 11610  
alishba@DESKTOP-S3K0VAS: $ jobs  
[1]+  Running                  sleep 30 &  
alishba@DESKTOP-S3K0VAS: $ fg %1  
sleep 30  
alishba@DESKTOP-S3K0VAS: $
```

The terminal output shows that a background job (PID 11610) was started with the command `sleep 30 &`. The `jobs` command lists it as a running background job. The `fg %1` command successfully brought the job to the foreground, and the prompt now shows `sleep 30` running.

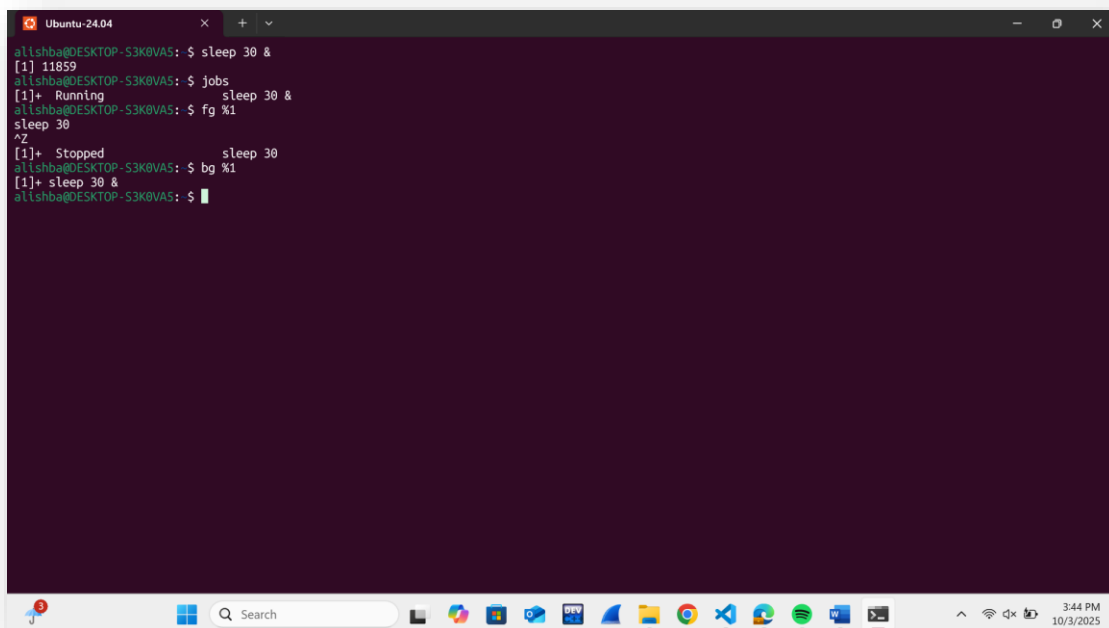
The terminal window is part of a desktop environment. The taskbar at the bottom includes the Windows logo, a search bar, and various application icons. The system tray on the right shows the time as 3:43 PM on 10/3/2025.

Suspend a job:

A terminal window titled 'Ubuntu-24.04' with a dark purple background. The user 'alishba@DESKTOP-S3K0VAS' runs the command 'sleep 30 &'. The prompt changes to '[1] 11859'. Then, the user runs 'jobs', showing '[1]+ Running sleep 30 &'. Next, the user runs 'fg %1', which brings the job to the foreground and shows 'sleep 30'. The user then presses Ctrl-Z (represented as '^Z'), and the job is stopped, showing '[1]+ Stopped sleep 30'. Finally, the user presses Enter, returning to the prompt 'alishba@DESKTOP-S3K0VAS: \$'. The window has a standard Ubuntu desktop environment at the bottom with a taskbar and system tray.

```
alishba@DESKTOP-S3K0VAS: $ sleep 30 &
[1] 11859
alishba@DESKTOP-S3K0VAS: $ jobs
[1]+  Running                  sleep 30 &
alishba@DESKTOP-S3K0VAS: $ fg %1
sleep 30
^Z
[1]+  Stopped                  sleep 30
alishba@DESKTOP-S3K0VAS: $
```

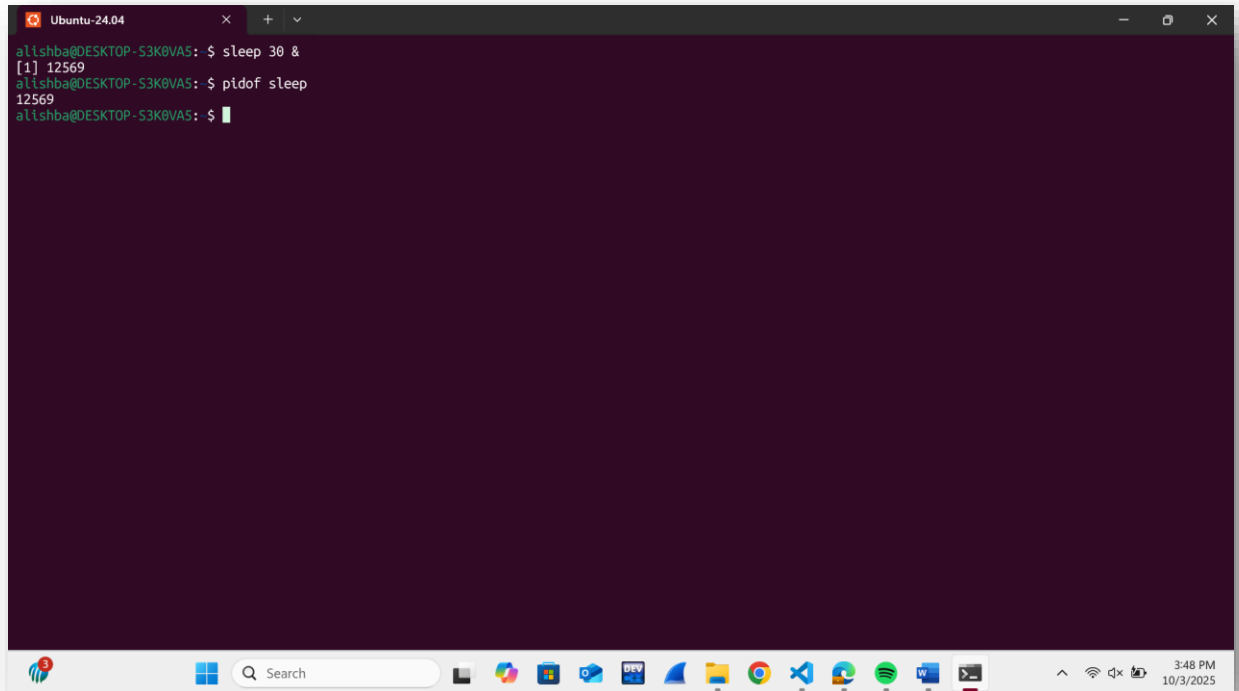
Resume suspended job in background:

A terminal window titled 'Ubuntu-24.04' with a dark purple background. The user 'alishba@DESKTOP-S3K0VAS' runs the command 'sleep 30 &'. The prompt changes to '[1] 11859'. Then, the user runs 'jobs', showing '[1]+ Running sleep 30 &'. Next, the user runs 'fg %1', which brings the job to the foreground and shows 'sleep 30'. The user then presses Ctrl-Z (represented as '^Z'), and the job is stopped, showing '[1]+ Stopped sleep 30'. Finally, the user runs 'bg %1', which resumes the job in the background, showing '[1]+ sleep 30 &'. The prompt returns to 'alishba@DESKTOP-S3K0VAS: \$'. The window has a standard Ubuntu desktop environment at the bottom with a taskbar and system tray.

```
alishba@DESKTOP-S3K0VAS: $ sleep 30 &
[1] 11859
alishba@DESKTOP-S3K0VAS: $ jobs
[1]+  Running                  sleep 30 &
alishba@DESKTOP-S3K0VAS: $ fg %1
sleep 30
^Z
[1]+  Stopped                  sleep 30
alishba@DESKTOP-S3K0VAS: $ bg %1
[1]+  sleep 30 &
alishba@DESKTOP-S3K0VAS: $
```

2.4 Process Identification

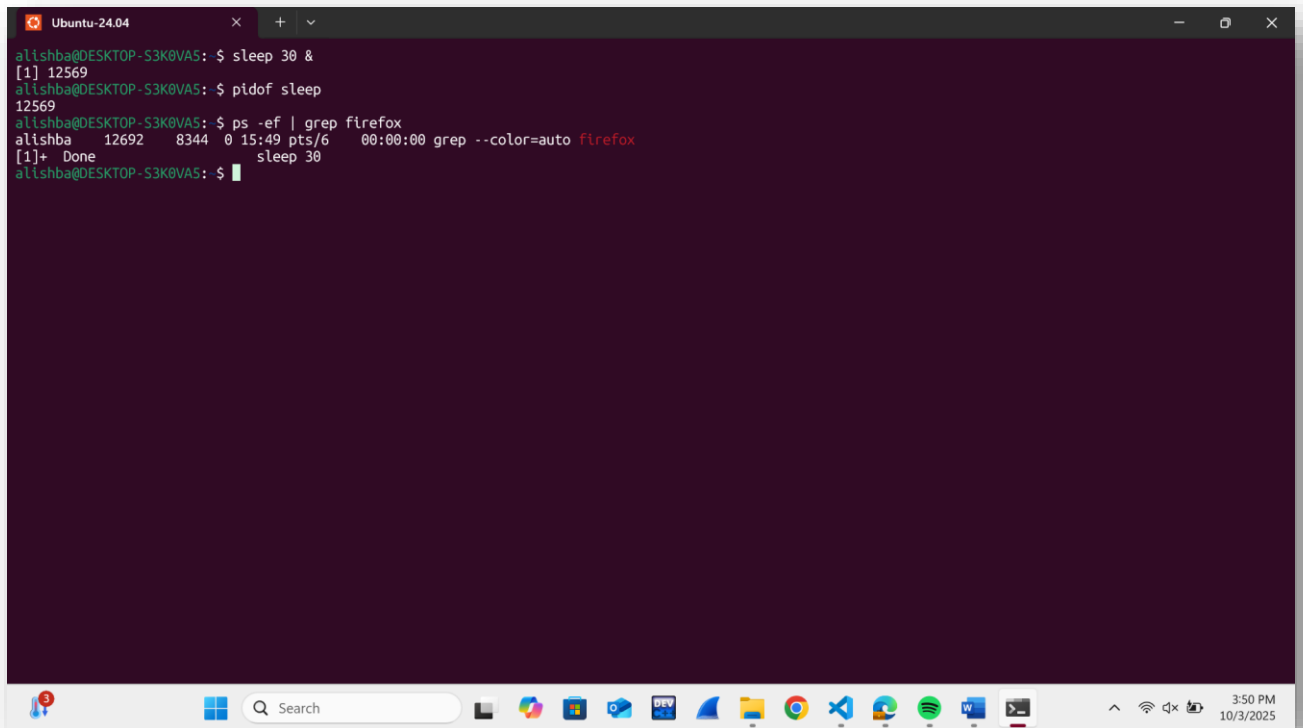
Get PID of a process by name:



```
Ubuntu-24.04
a1ishba@DESKTOP-S3K0VAS: $ sleep 30 &
[1] 12569
a1ishba@DESKTOP-S3K0VAS: $ pidof sleep
12569
a1ishba@DESKTOP-S3K0VAS: $
```

The image shows a terminal window titled 'Ubuntu-24.04' with a dark purple background. The user 'a1ishba' is at the 'DESKTOP-S3K0VAS' machine. They run the command 'sleep 30 &', which returns '[1] 12569'. Then they run 'pidof sleep', which returns '12569'. The terminal window is open on a Windows desktop, with the taskbar visible at the bottom showing various application icons and the system clock indicating 3:48 PM on 10/3/2025.

Search using ps and grep :

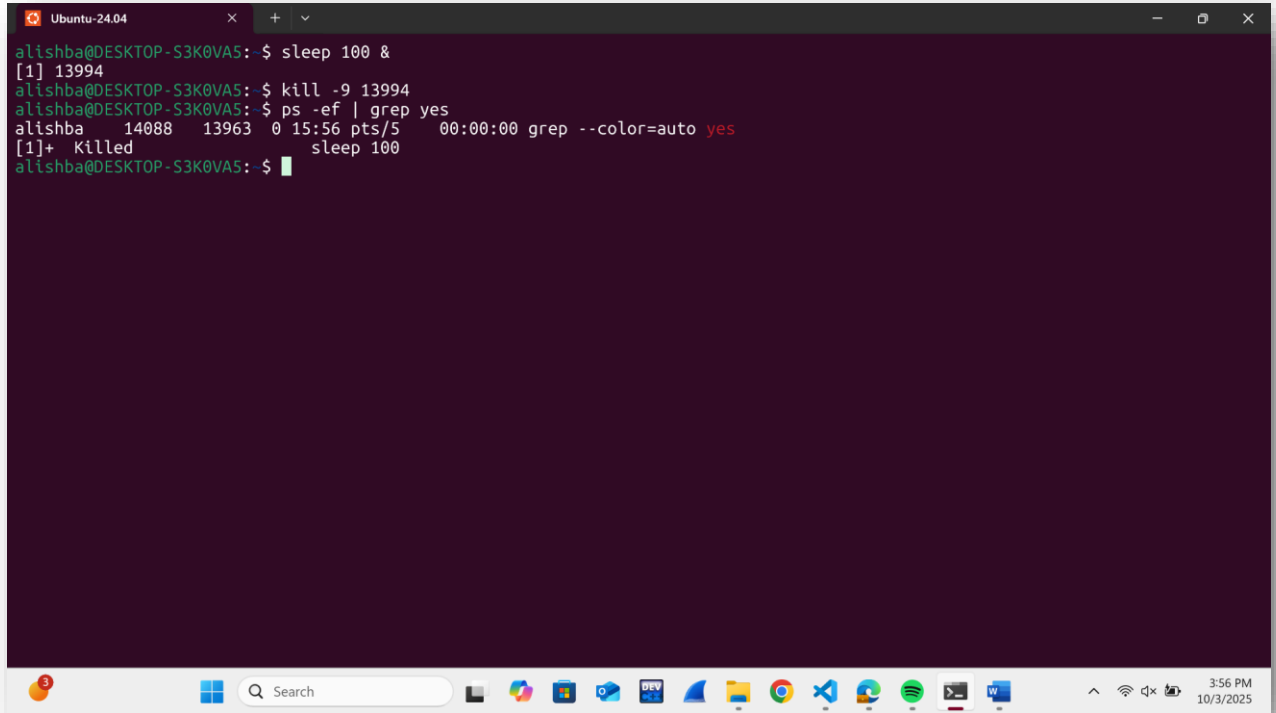


```
alishba@DESKTOP-S3K0VAS: ~$ sleep 30 &
[1] 12569
alishba@DESKTOP-S3K0VAS: ~$ pidof sleep
12569
alishba@DESKTOP-S3K0VAS: ~$ ps -ef | grep firefox
alishba  12692   8344  0 15:49 pts/6    00:00:00 grep --color=auto firefox
[1]+  Done                  sleep 30
alishba@DESKTOP-S3K0VAS: ~$
```

The image shows a terminal window titled 'Ubuntu-24.04' with a dark purple background. The user 'alishba' is at the prompt. They run 'sleep 30 &' which starts a background process with PID 12569. Then they run 'pidof sleep' which returns '12569'. Finally, they run 'ps -ef | grep firefox'. The output shows a line for the user 'alishba' with PID 12692, PPID 8344, and the command 'grep --color=auto firefox', where 'firefox' is highlighted in red. The prompt returns to '\$' after the command completes. The terminal window is overlaid on a Windows desktop environment, with the taskbar visible at the bottom showing various application icons and the system clock indicating 3:50 PM on 10/3/2025.

2.5 Killing Processes

Kill by PID:



```
alishba@DESKTOP-S3K0VA5:~$ sleep 100 &
[1] 13994
alishba@DESKTOP-S3K0VA5:~$ kill -9 13994
alishba@DESKTOP-S3K0VA5:~$ ps -ef | grep yes
alishba  14088  13963  0 15:56 pts/5    00:00:00 grep --color=auto yes
[1]+  Killed                  sleep 100
alishba@DESKTOP-S3K0VA5:~$
```

The image shows a terminal window titled 'Ubuntu-24.04' with a dark purple background. The user 'alishba' is at the prompt 'alishba@DESKTOP-S3K0VA5:~\$'. They execute 'sleep 100 &', which starts a background process with PID 13994. Then they execute 'kill -9 13994'. Next, they run 'ps -ef | grep yes', which shows the 'sleep 100' process with PID 13963. Finally, they press Enter, and the terminal shows '[1]+ Killed sleep 100', indicating the process has been successfully terminated. The terminal window is part of a desktop environment with a taskbar at the bottom showing various application icons and the system clock at 3:56 PM on 10/3/2025.

Kill all processes by name:

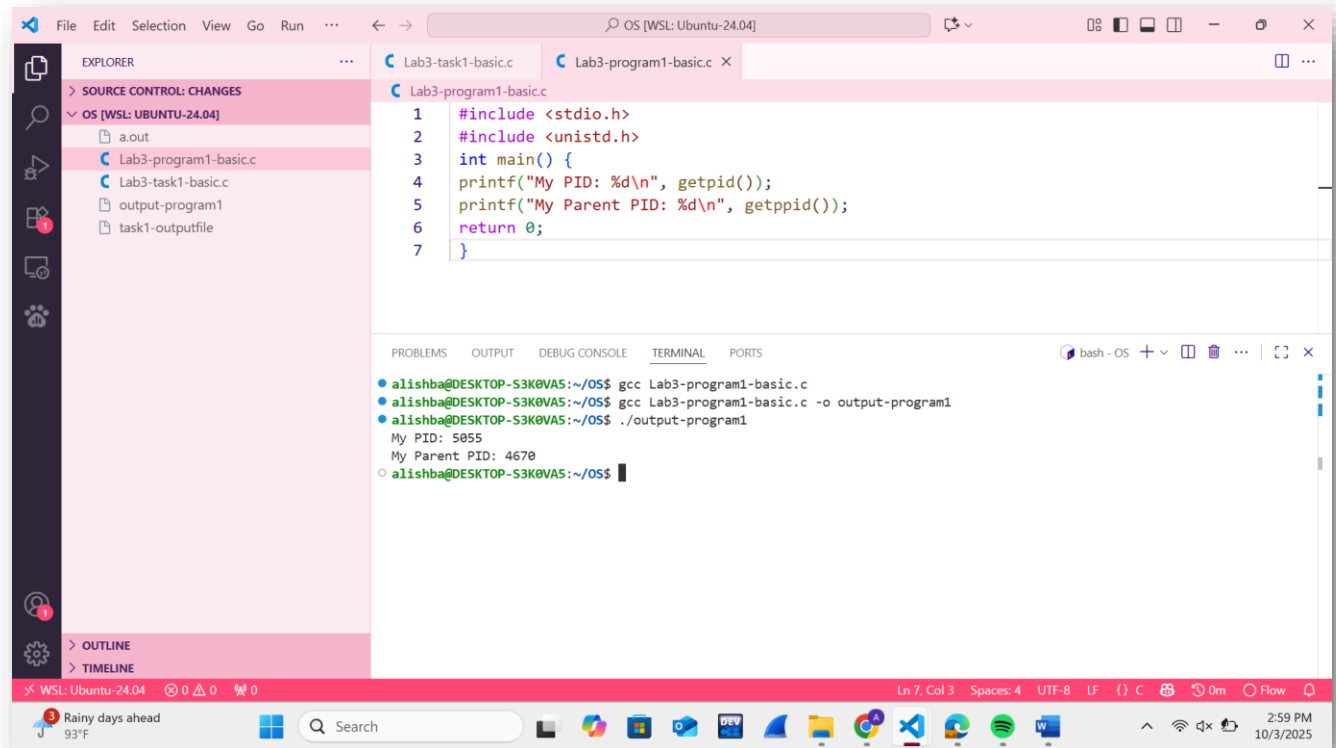
```
Ubuntu-24.04
alishba@DESKTOP-S3K0VAS: $ sleep 1000 &
[1] 14190
alishba@DESKTOP-S3K0VAS: $ ps -ef | grep sleep
alishba  14190  13963  0 15:57 pts/5    00:00:00 sleep 1000
alishba  14218  13963  0 15:57 pts/5    00:00:00 grep --color=auto sleep
alishba@DESKTOP-S3K0VAS: $ killall sleep
[1]+  Terminated                  sleep 1000
alishba@DESKTOP-S3K0VAS: $
```

Practice Task

```
Ubuntu-24.04
alishba@DESKTOP-S3K0VAS: $ yes > /dev/null &
[2] 14578
alishba@DESKTOP-S3K0VAS: $ ps -ef | grep yes
alishba  14354  13963  99 15:58 pts/5    00:01:25 yes
alishba  14578  13963  99 15:59 pts/5    00:00:15 yes
alishba  14626  13963  0 15:59 pts/5    00:00:00 grep --color=auto yes
alishba@DESKTOP-S3K0VAS: $ kill -9 14578
alishba@DESKTOP-S3K0VAS: $ ps -ef | grep yes
alishba  14354  13963  99 15:58 pts/5    00:01:48 yes
alishba  14696  13963  0 16:00 pts/5    00:00:00 grep --color=auto yes
[2]+  Killed                        yes > /dev/null
alishba@DESKTOP-S3K0VAS: $
```

3. C Programs on Processes

Program 1: Print PID and PPID



```
File Edit Selection View Go Run ... OS [WSL: Ubuntu-24.04]
```

EXPLORER

- SOURCE CONTROL: CHANGES
- OS [WSL: UBUNTU-24.04]
 - a.out
 - Lab3-program1-basic.c
 - Lab3-task1-basic.c
 - output-program1
 - task1-outputfile
- OUTLINE
- TIMELINE

Lab3-program1-basic.c

```
1 #include <stdio.h>
2 #include <unistd.h>
3 int main() {
4     printf("My PID: %d\n", getpid());
5     printf("My Parent PID: %d\n", getppid());
6     return 0;
7 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

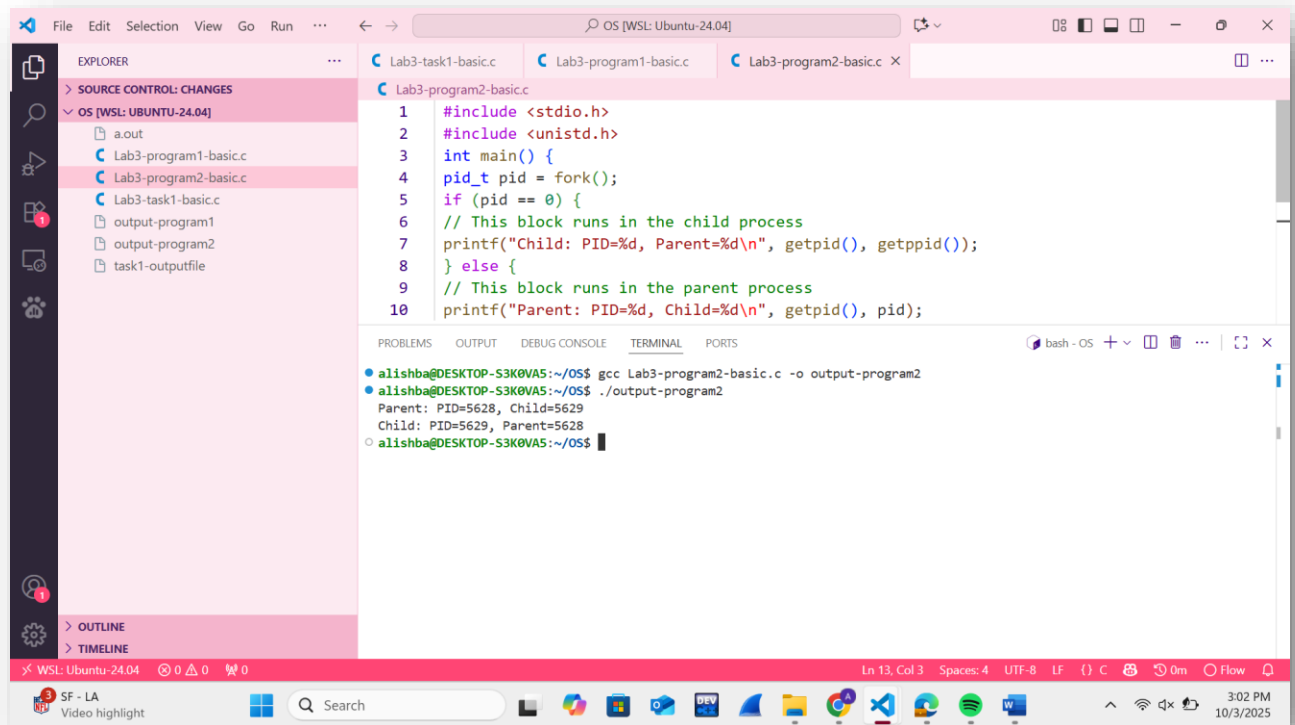
bash - OS

```
alishba@DESKTOP-S3K0VA5:~/OS$ gcc Lab3-program1-basic.c
alishba@DESKTOP-S3K0VA5:~/OS$ gcc Lab3-program1-basic.c -o output-program1
alishba@DESKTOP-S3K0VA5:~/OS$ ./output-program1
My PID: 5055
My Parent PID: 4670
alishba@DESKTOP-S3K0VA5:~/OS$
```

WSL: Ubuntu-24.04 Rainy days ahead 93°F Search

Ln 7, Col 3 Spaces: 4 UTF-8 LF C 0m Flow 2:59 PM 10/3/2025

Program 2: Fork – Creating Child Process



The screenshot displays the Visual Studio Code interface within a WSL (Ubuntu-24.04) environment. The Explorer sidebar on the left shows the file structure under the project name 'OS [WSL: UBUNTU-24.04]'. The Editor pane shows the code for 'Lab3-program2-basic.c'. The Terminal pane at the bottom shows the execution of the program, which successfully creates a child process using the fork() function.

```
1 #include <stdio.h>
2 #include <unistd.h>
3 int main() {
4     pid_t pid = fork();
5     if (pid == 0) {
6         // This block runs in the child process
7         printf("Child: PID=%d, Parent=%d\n", getpid(), getppid());
8     } else {
9         // This block runs in the parent process
10        printf("Parent: PID=%d, Child=%d\n", getpid(), pid);
11    }
```

Terminal Output:

```
alishba@DESKTOP-S3K0VA5:~/OS$ gcc Lab3-program2-basic.c -o output-program2
alishba@DESKTOP-S3K0VA5:~/OS$ ./output-program2
Parent: PID=5628, Child=5629
Child: PID=5629, Parent=5628
alishba@DESKTOP-S3K0VA5:~/OS$
```

Program 3: Exec1 – Replacing a Process

The screenshot shows the Visual Studio Code editor interface with the following components:

- EXPLORER:** Displays the file structure for the project, including source files and output files.
- EDITOR:** Shows the code for `Lab3-program3-basic.c`, which uses `fork()` and `execvp()` to replace the parent process with `ls`.
- TERMINAL:** Shows the compilation and execution of the program, followed by a `ps` command output.

```
1 #include <stdio.h>
2 #include <unistd.h>
3 int main() {
4     pid_t pid = fork();
5     if (pid == 0) {
6         execvp("ls", "ls", "-l", NULL);
7         printf("This will not print if exec succeeds.\n");
8     } else {
9         printf("Parent still running...\n");
10    }
```

```
alishba@DESKTOP-S3K0VAS:~/OS$ gcc Lab3-program3-basic.c -o output-program3
alishba@DESKTOP-S3K0VAS:~/OS$ ./output-program3
Parent still running...
total 96
-rw-r--r-- 1 alishba alishba 146 Oct 3 14:57 Lab3-program1-basic.c
-rw-r--r-- 1 alishba alishba 315 Oct 3 15:01 Lab3-program2-basic.c
-rw-r--r-- 1 alishba alishba 241 Oct 3 15:05 Lab3-program3-basic.c
-rw-r--r-- 1 alishba alishba 591 Oct 3 14:43 Lab3-task1-basic.c
-rwxr-xr-x 1 alishba alishba 16064 Oct 3 14:59 a.out
-rwxr-xr-x 1 alishba alishba 16064 Oct 3 14:59 output-program1
-rwxr-xr-x 1 alishba alishba 16104 Oct 3 15:02 output-program2
-rwxr-xr-x 1 alishba alishba 16056 Oct 3 15:06 output-program3
-rwxr-xr-x 1 alishba alishba 16184 Oct 3 14:47 task1-outputfile
alishba@DESKTOP-S3K0VAS:~/OS$
```

Program 4: Wait – Synchronization

The screenshot shows the Visual Studio Code editor with the file `Lab3-program4-basic.c` open. The code implements a program that forks a child process and then waits for it to finish using `waitpid`.

```
1 #include <stdio.h>
2 #include <unistd.h>
3 #include <sys/wait.h>
4 int main() {
5     pid_t pid = fork();
6     if (pid == 0) {
7         execlp("ls", "ls", "-l", NULL);
8         printf("This will not print if exec succeeds.\n");
9     } else {
10        waitpid(pid, NULL, 0); // Wait for the child process to finish
11    }
```

The terminal output shows the compilation and execution of the program:

```
alishba@DESKTOP-S3K0VA5:~/OS$ gcc Lab3-program4-basic.c -o output-program4
alishba@DESKTOP-S3K0VA5:~/OS$ ./output-program4
total 116
-rw-r--r-- 1 alishba alishba 146 Oct 3 14:57 Lab3-program1-basic.c
-rw-r--r-- 1 alishba alishba 315 Oct 3 15:01 Lab3-program2-basic.c
-rw-r--r-- 1 alishba alishba 241 Oct 3 15:05 Lab3-program3-basic.c
-rw-r--r-- 1 alishba alishba 328 Oct 3 15:07 Lab3-program4-basic.c
-rw-r--r-- 1 alishba alishba 591 Oct 3 14:43 Lab3-task1-basic.c
-rwxr-xr-x 1 alishba alishba 16064 Oct 3 14:59 a.out
-rwxr-xr-x 1 alishba alishba 16064 Oct 3 14:59 output-program1
-rwxr-xr-x 1 alishba alishba 16104 Oct 3 15:02 output-program2
-rwxr-xr-x 1 alishba alishba 16056 Oct 3 15:06 output-program3
-rwxr-xr-x 1 alishba alishba 16104 Oct 3 15:07 output-program4
-rwxr-xr-x 1 alishba alishba 16184 Oct 3 14:47 task1-outputfile
Parent still running...
```