# National Textile University

## Department of Computer Science

**Subject**

Operating System

**Submitted to:**

Sir Nasir Mehmood

**Submitted by:**

kashmir Jamshaid

**Registration Number**

23-NTU-CS-1167

**Home_Task**

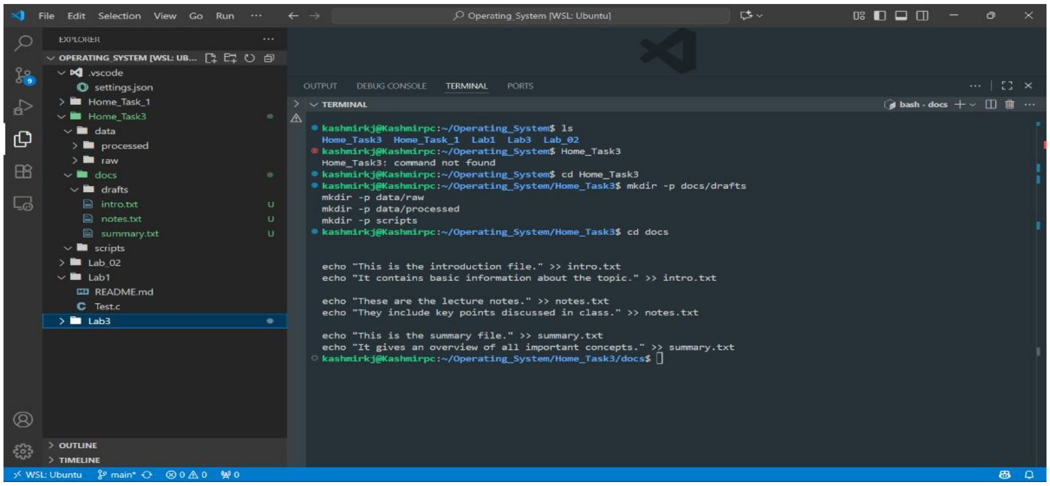
03

**Semester**

5th

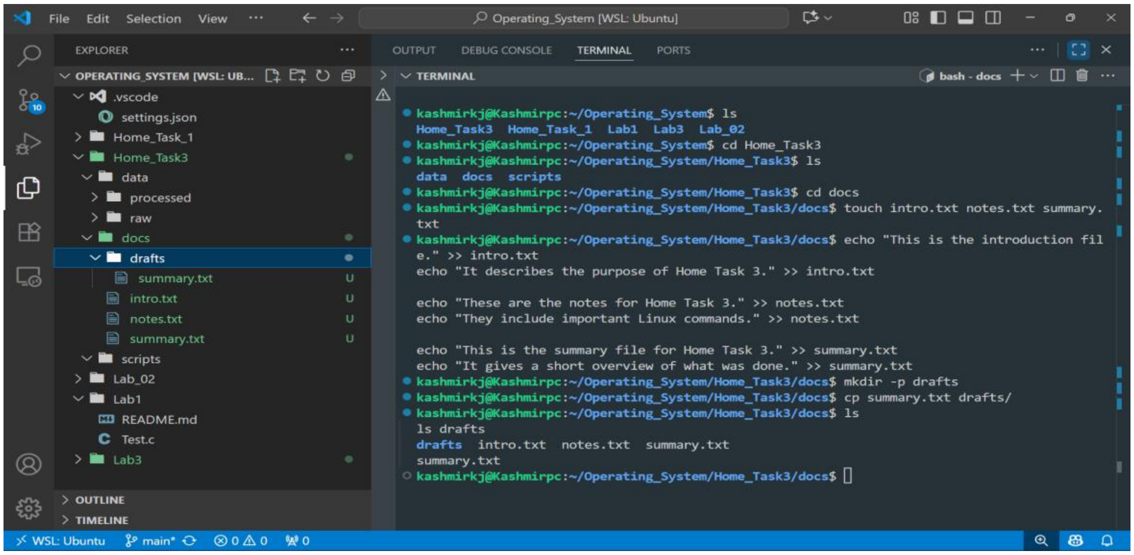
## Part 1: File and Directory Operation

1. Create the following directory structure in your home directory:



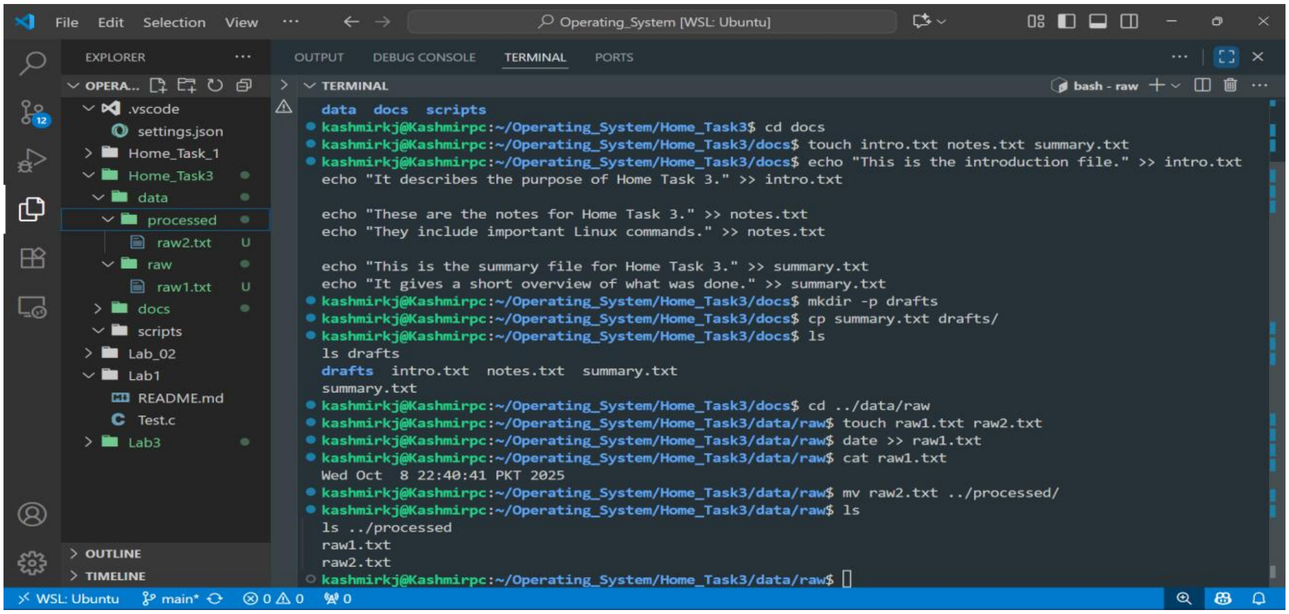
```
kashmirkj@Kashmirpc:~/Operating_System$ ls
Home_Task3  Home_Task_1  Lab1  Lab3  Lab_02
kashmirkj@Kashmirpc:~/Operating_System$ Home_Task3
Home_Task3: command not found
kashmirkj@Kashmirpc:~/Operating_System$ cd Home_Task3
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3$ mkdir -p docs/drafts
mkdir -p data/raw
mkdir -p data/processed
mkdir -p scripts
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3$ cd docs


echo "This is the introduction file." >> intro.txt
echo "It contains basic information about the topic." >> intro.txt

echo "These are the lecture notes." >> notes.txt
echo "They include key points discussed in class." >> notes.txt

echo "This is the summary file." >> summary.txt
echo "It gives an overview of all important concepts." >> summary.txt
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/docs$ 
```

2. Inside docs/ :
   Create three files: intro.txt , notes.txt , summary.txt . Add at least two lines of text into each using echo >> . Copy summary.txt into the drafts/ folder using



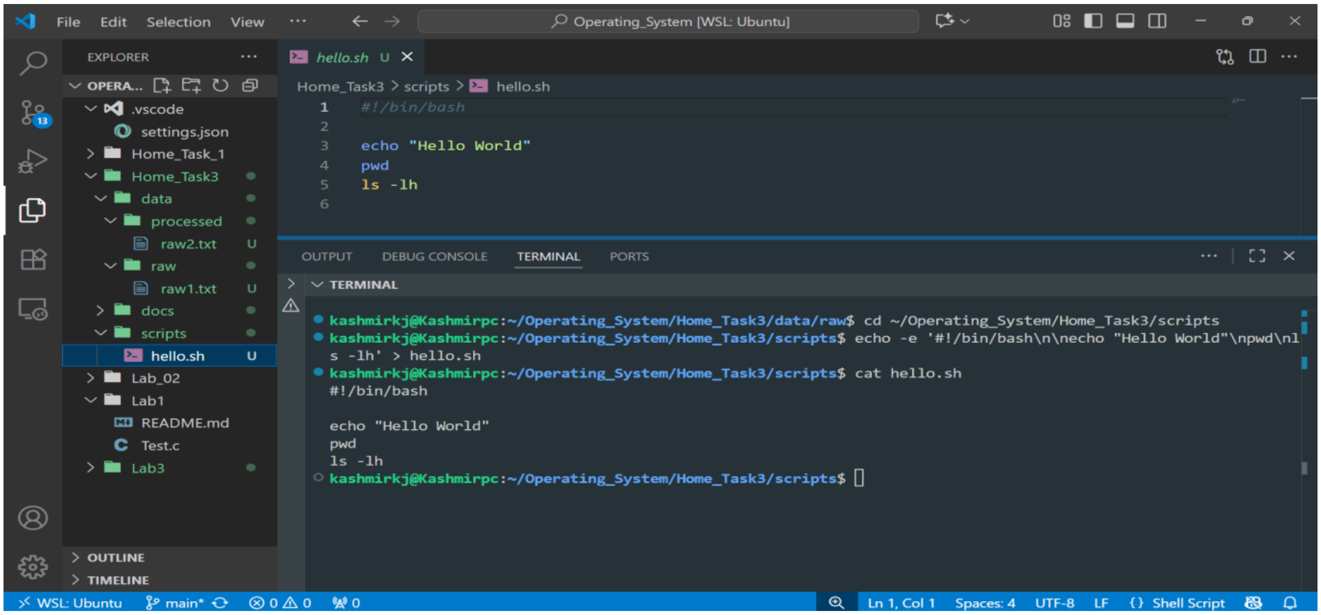
```
kashmirkj@Kashmirpc:~/Operating_System$ ls
Home_Task3  Home_Task_1  Lab1  Lab3  Lab_02
kashmirkj@Kashmirpc:~/Operating_System$ cd Home_Task3
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3$ ls
data  docs  scripts
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3$ cd docs
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/docs$ touch intro.txt notes.txt summary.
txt
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/docs$ echo "This is the introduction fil
e." >> intro.txt
echo "It describes the purpose of Home Task 3." >> intro.txt

echo "These are the notes for Home Task 3." >> notes.txt
echo "They include important Linux commands." >> notes.txt

echo "This is the summary file for Home Task 3." >> summary.txt
echo "It gives a short overview of what was done." >> summary.txt
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/docs$ mkdir -p drafts
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/docs$ cp summary.txt drafts/
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/docs$ ls
ls drafts
drafts  intro.txt  notes.txt  summary.txt
summary.txt
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/docs$ 
```

3. Inside data/raw/ :
   Create two files: raw1.txt , raw2.txt . Append the current date into cp command.



```
data  docs  scripts
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3$ cd docs
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/docs$ touch intro.txt notes.txt summary.txt
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/docs$ echo "This is the introduction file." >> intro.txt
echo "It describes the purpose of Home Task 3." >> intro.txt

echo "These are the notes for Home Task 3." >> notes.txt
echo "They include important Linux commands." >> notes.txt

echo "This is the summary file for Home Task 3." >> summary.txt
echo "It gives a short overview of what was done." >> summary.txt
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/docs$ mkdir -p drafts
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/docs$ cp summary.txt drafts/
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/docs$ ls
ls drafts
drafts  intro.txt  notes.txt  summary.txt
summary.txt
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/docs$ cd ../data/raw
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/data/raw$ touch raw1.txt raw2.txt
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/data/raw$ date >> raw1.txt
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/data/raw$ cat raw1.txt
Wed Oct  8 22:40:41 PKT 2025
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/data/raw$ mv raw2.txt ../processed/
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/data/raw$ ls
ls ../processed
raw1.txt
raw2.txt
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/data/raw$ 
```

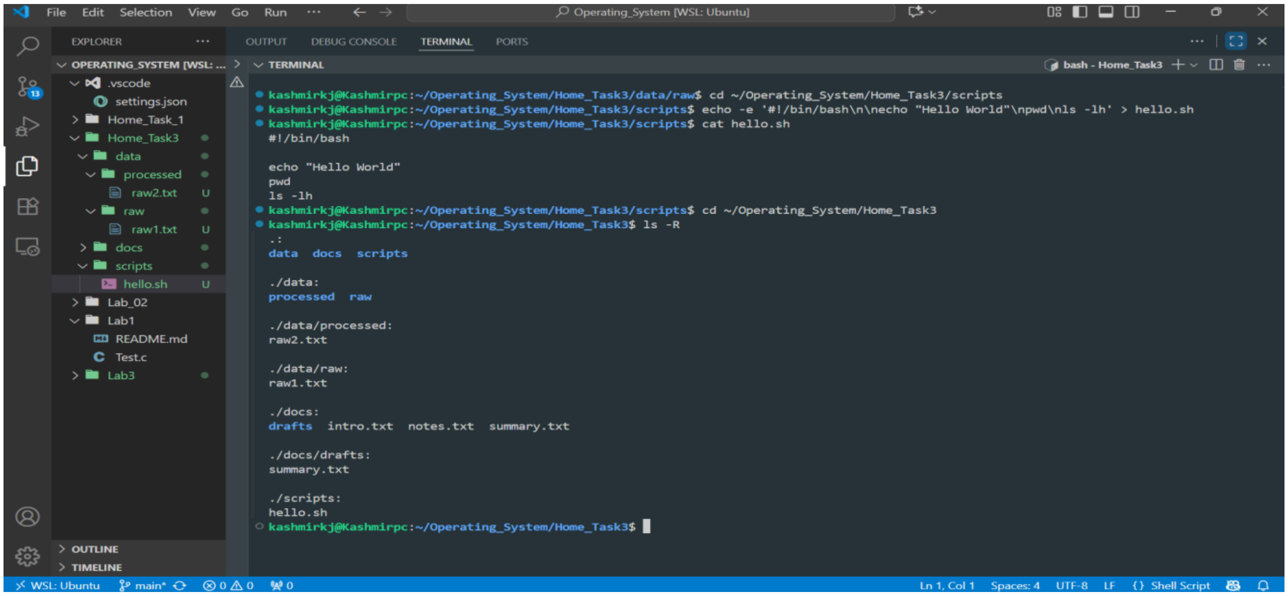
4. Inside scripts/ :
   Create a script named echo "Hello World" pwd ls-lh hello.sh with the following content:



```
#!/bin/bash

echo "Hello World"
pwd
ls -lh

```

```
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/data/raw$ cd ~/Operating_System/Home_Task3/scripts
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/scripts$ echo -e '#!/bin/bash\n\necho "Hello World"\npwd\nl
s -lh' > hello.sh
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/scripts$ cat hello.sh
#!/bin/bash

echo "Hello World"
pwd
ls -lh
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/scripts$ 
```

5. Display the directory structure recursively and take a screenshot:



```
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/data/raw$ cd ~/Operating_System/Home_Task3/scripts
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/scripts$ echo -e '#!/bin/bash\n\necho "Hello World"\npwd\nls -lh' > hello.sh
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/scripts$ cat hello.sh
#!/bin/bash

echo "Hello World"
pwd
ls -lh
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/scripts$ cd ~/Operating_System/Home_Task3
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3$ ls -R
.:
data  docs  scripts

./data:
processed  raw

./data/processed:
raw2.txt

./data/raw:
raw1.txt

./docs:
drafts  intro.txt  notes.txt  summary.txt

./docs/drafts:
summary.txt

./scripts:
hello.sh
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3$ 
```

## Part 2:

## Practice with Basic Linux Commands Run the following commands

inside Lab_3/ and note their outputs:

pwd → Show current working directory.

whoami → Display the current logged-in user.

touch extra.txt → Create an empty file.

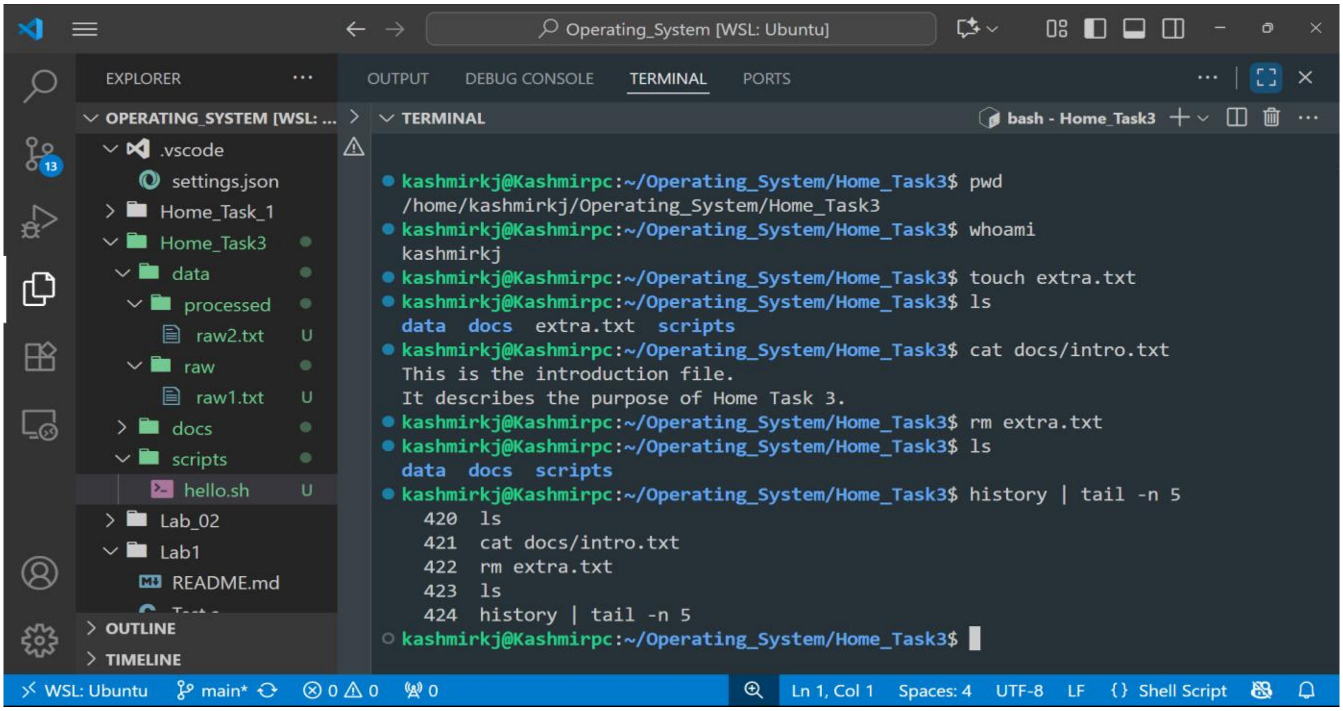
cat intro.txt → Display file contents.

rm extra.txt → Delete a file.

history | tail -n 5 → Show your last 5 executed commands.

clear → Clear the terminal.

Take screenshots of commands and outputs.

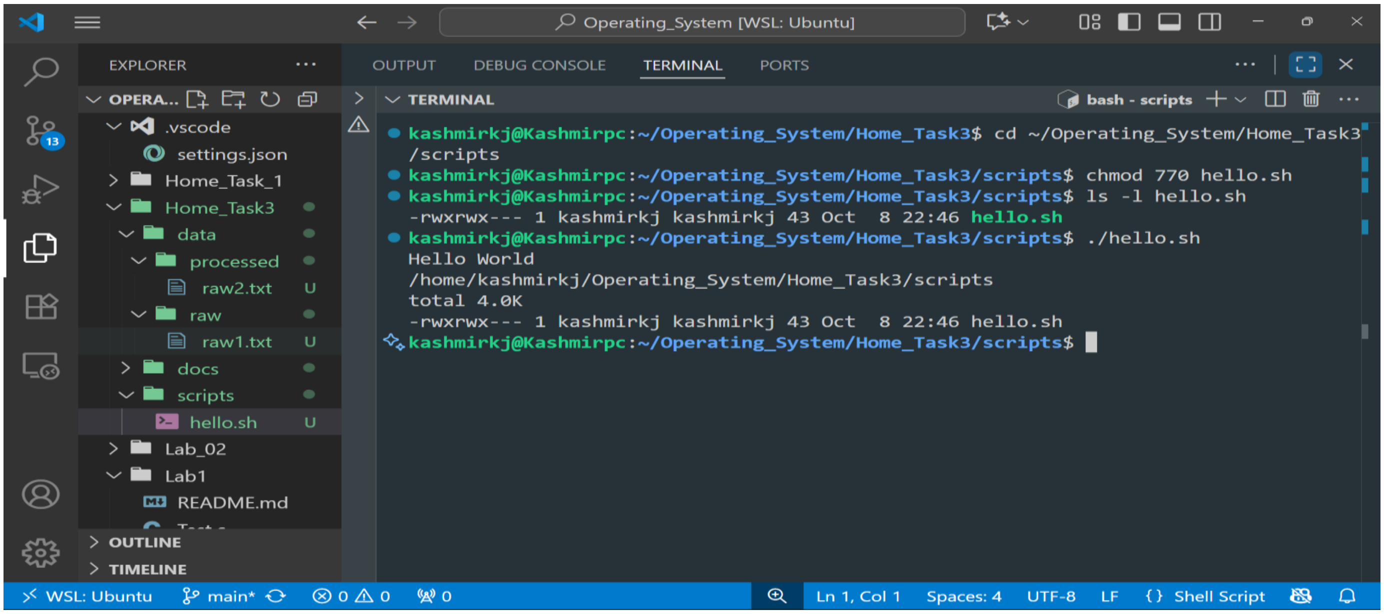


```
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3$ pwd
/home/kashmirkj/Operating_System/Home_Task3
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3$ whoami
kashmirkj
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3$ touch extra.txt
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3$ ls
data  docs  extra.txt  scripts
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3$ cat docs/intro.txt
This is the introduction file.
It describes the purpose of Home Task 3.
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3$ rm extra.txt
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3$ ls
data  docs  scripts
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3$ history | tail -n 5
  420  ls
  421  cat docs/intro.txt
  422  rm extra.txt
  423  ls
  424  history | tail -n 5
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3$
```

## Part 3:

## File Permissions and Ownership

- Change the permissions of hello.sh so that:
- Owner → Read,
- Write & Execute Group → Read,
- Write & Execute Others → No permissions Run the script using:
  ./hello.sh



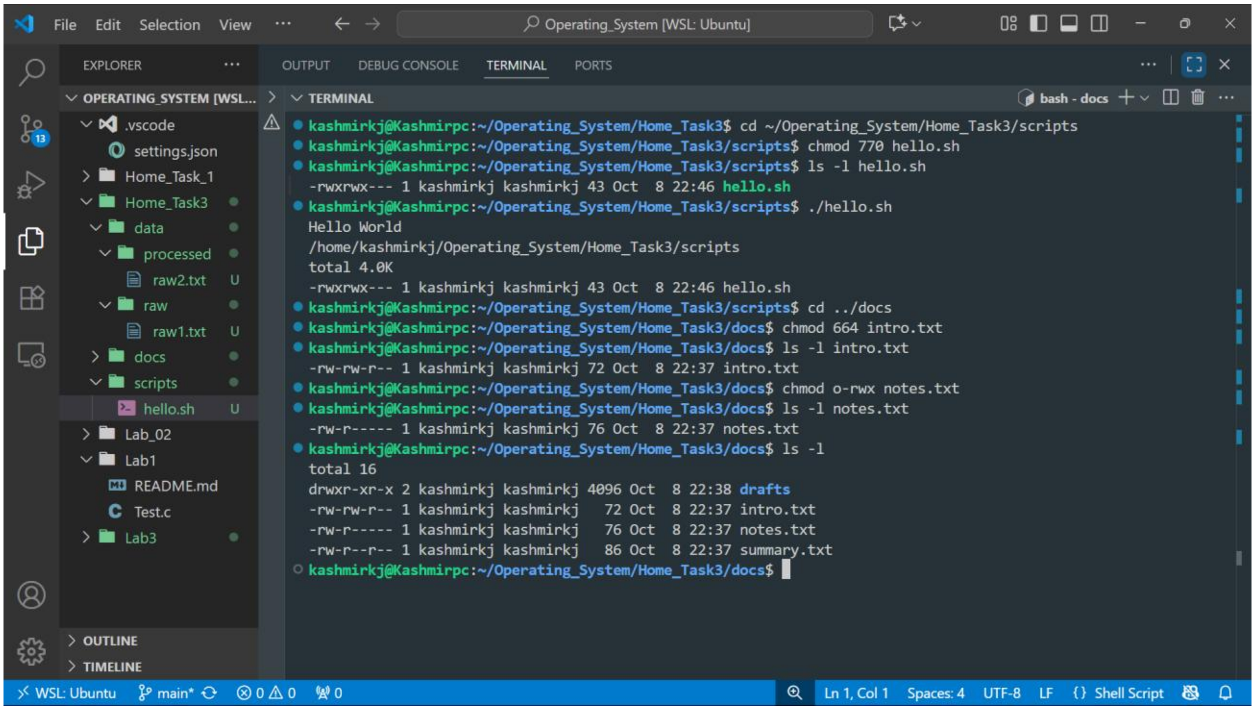
```
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3$ cd ~/Operating_System/Home_Task3
/scripts
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/scripts$ chmod 770 hello.sh
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/scripts$ ls -l hello.sh
-rwxrwx--- 1 kashmirkj kashmirkj 43 Oct  8 22:46 hello.sh
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/scripts$ ./hello.sh
Hello World
/home/kashmirkj/Operating_System/Home_Task3/scripts
total 4.0K
-rwxrwx--- 1 kashmirkj kashmirkj 43 Oct  8 22:46 hello.sh
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/scripts$
```

- Change the permissions of intro.txt using numeric notation so that:
- Owner → Read & Write
- Group → Read & Write
- Others → Read only

3. Change the permissions of have any permission on it.

4. Verify all changes with:

ls-l notes.txt using symbolic notation so that others don't

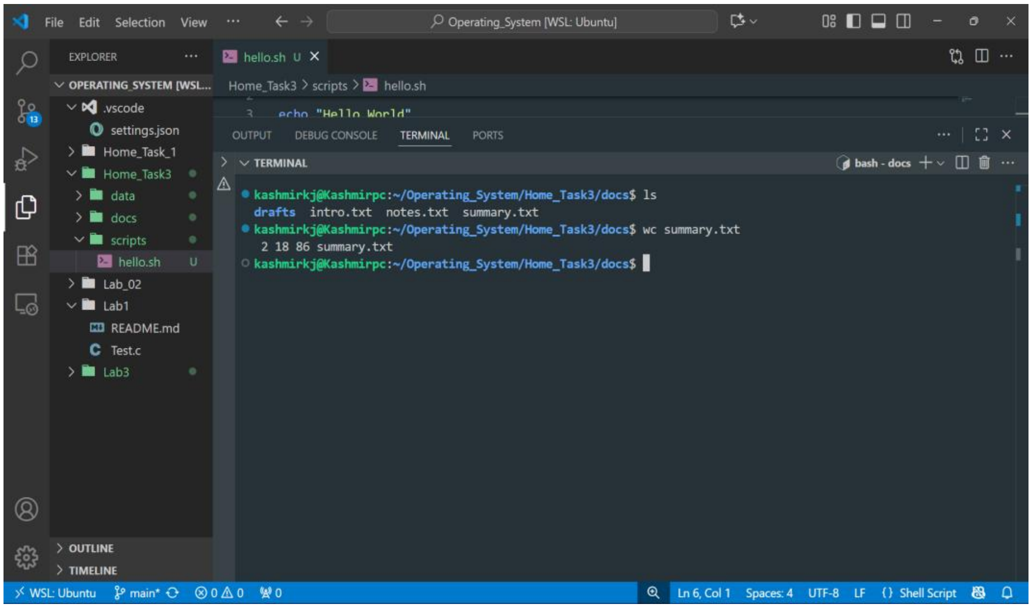


```
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3$ cd ~/Operating_System/Home_Task3/scripts
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/scripts$ chmod 770 hello.sh
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/scripts$ ls -l hello.sh
-rwxrwx--- 1 kashmirkj kashmirkj 43 Oct  8 22:46 hello.sh
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/scripts$ ./hello.sh
Hello World
/home/kashmirkj/Operating_System/Home_Task3/scripts
total 4.0K
-rwxrwx--- 1 kashmirkj kashmirkj 43 Oct  8 22:46 hello.sh
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/scripts$ cd ../docs
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/docs$ chmod 664 intro.txt
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/docs$ ls -l intro.txt
-rw-rw-r-- 1 kashmirkj kashmirkj 72 Oct  8 22:37 intro.txt
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/docs$ chmod o-rwx notes.txt
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/docs$ ls -l notes.txt
-rw-r----- 1 kashmirkj kashmirkj 76 Oct  8 22:37 notes.txt
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/docs$ ls -l
total 16
drwxr-xr-x 2 kashmirkj kashmirkj 4096 Oct  8 22:38 drafts
-rw-rw-r-- 1 kashmirkj kashmirkj   72 Oct  8 22:37 intro.txt
-rw-r----- 1 kashmirkj kashmirkj   76 Oct  8 22:37 notes.txt
-rw-r--r-- 1 kashmirkj kashmirkj   86 Oct  8 22:37 summary.txt
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/docs$
```

# Part 4:

# Reading & Searching Files Inside docs/ :

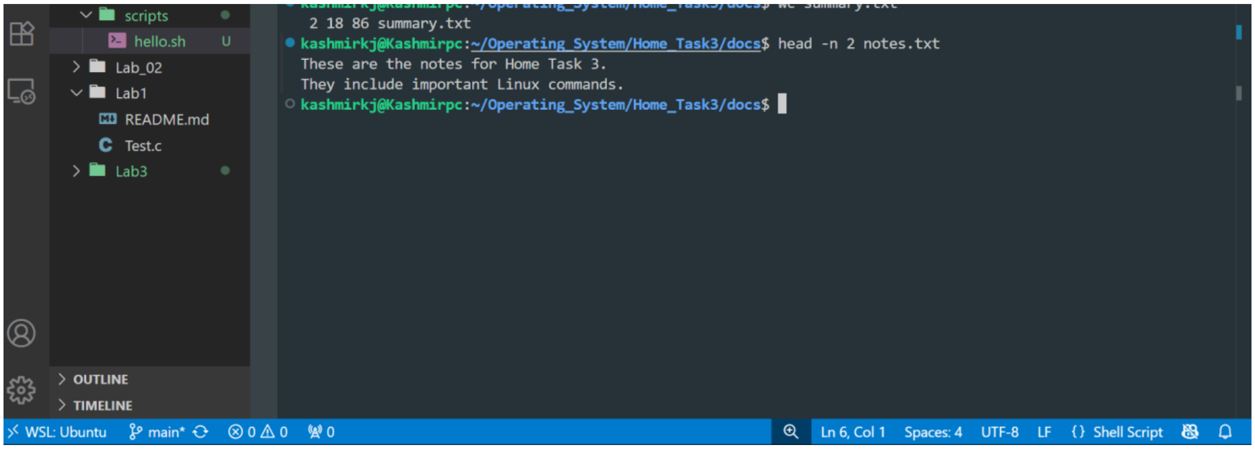
1. Count the number of lines, words, and characters in



```
3   echo "Hello World"
```

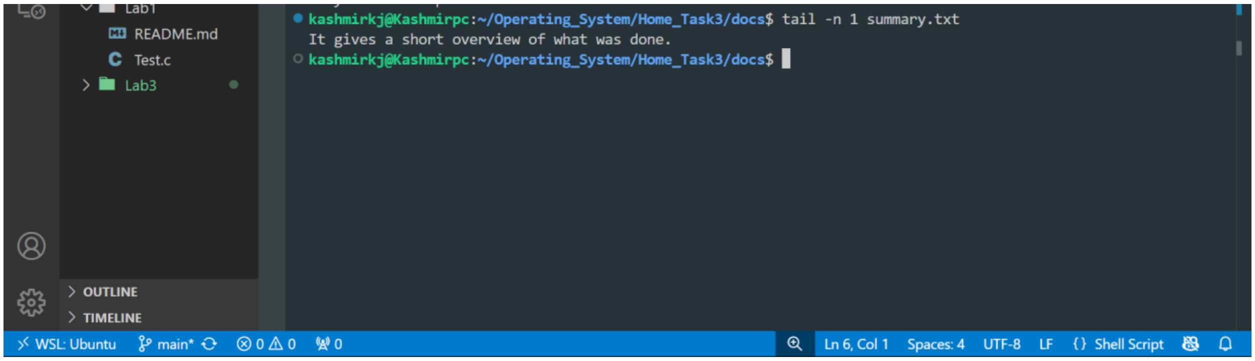
```
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/docs$ ls
drafts  intro.txt  notes.txt  summary.txt
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/docs$ wc summary.txt
 2 18 86 summary.txt
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/docs$ 
```

2. Show only the first 2 lines of notes.txt using summary.txt using wc . head -n 2 .



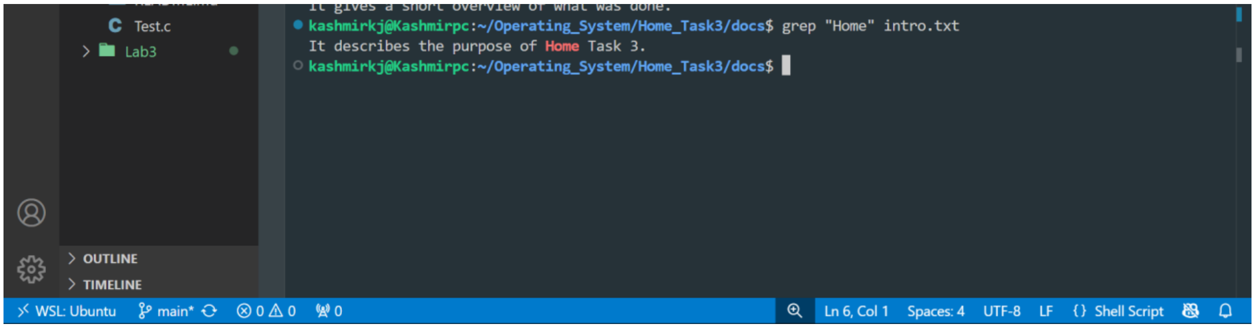
```
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/docs$ wc summary.txt
 2 18 86 summary.txt
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/docs$ head -n 2 notes.txt
These are the notes for Home Task 3.
They include important Linux commands.
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/docs$ 
```

3. Show the last line of summary.txt using tail -n 1 .



```
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/docs$ tail -n 1 summary.txt
It gives a short overview of what was done.
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/docs$ 
```

4. Search for a keyword (of your choice) in intro.txt using

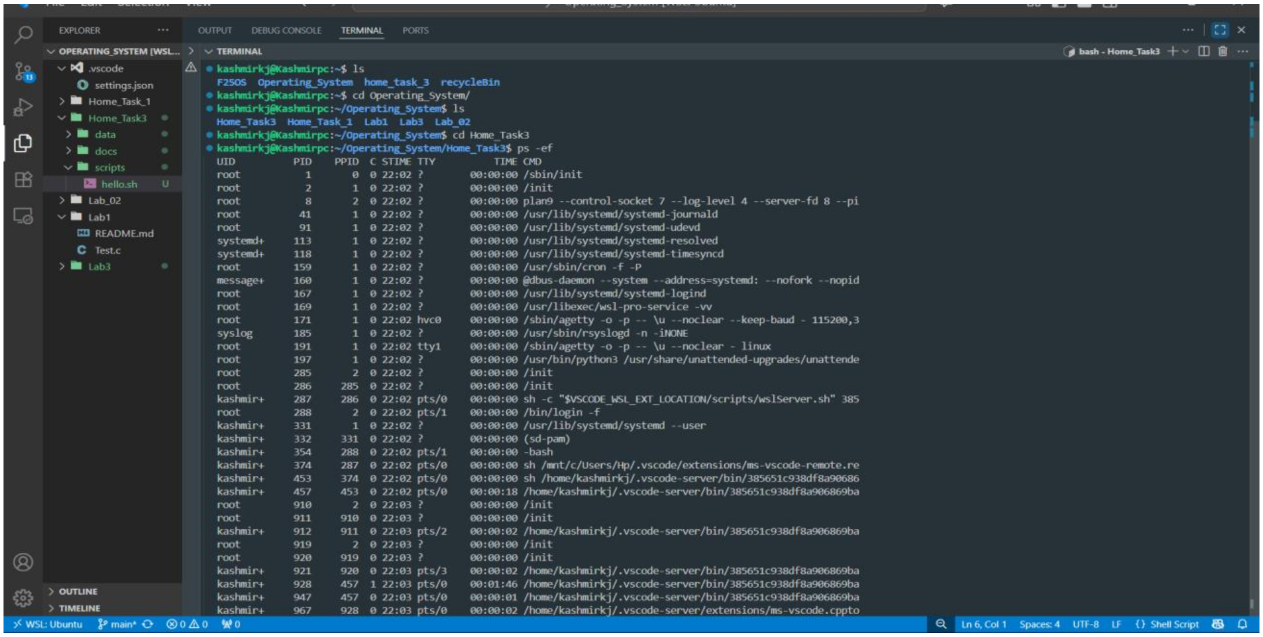


```
It gives a short overview of what was done.
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/docs$ grep "Home" intro.txt
It describes the purpose of Home Task 3.
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/docs$
```

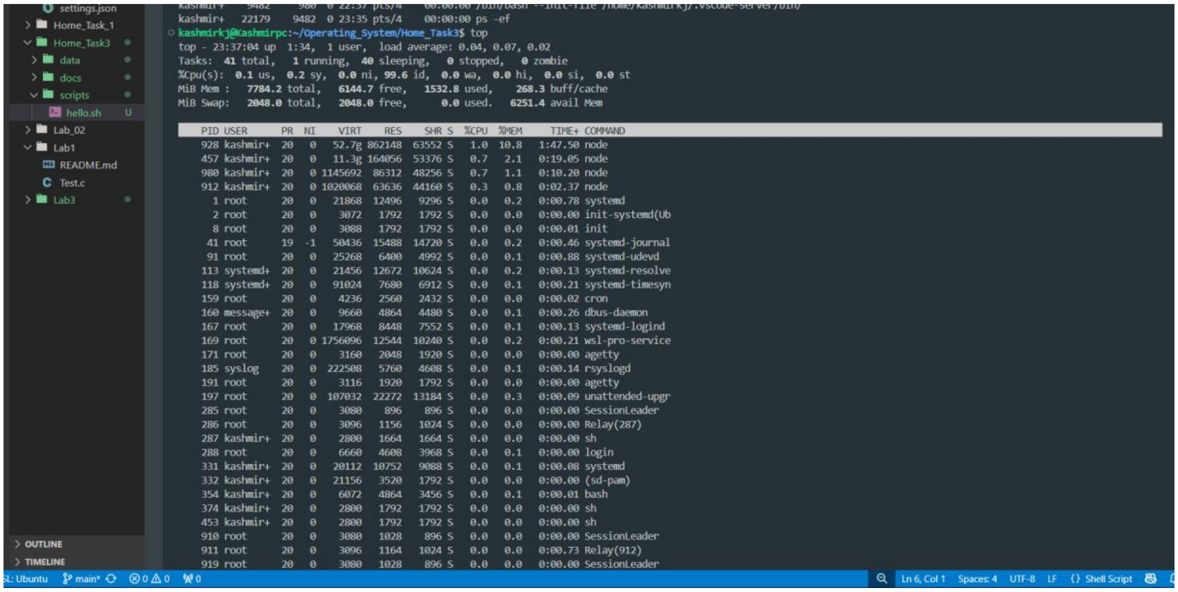
## Part 5:

## Linux Process Commands

1. Exploring Processes Use ps -ef and identify 3 processes running on your system. Note their PID, PPID, and command. Run top for 20–30 seconds. Write down: Which process is consuming the most CPU. Which process is consuming the most memory.

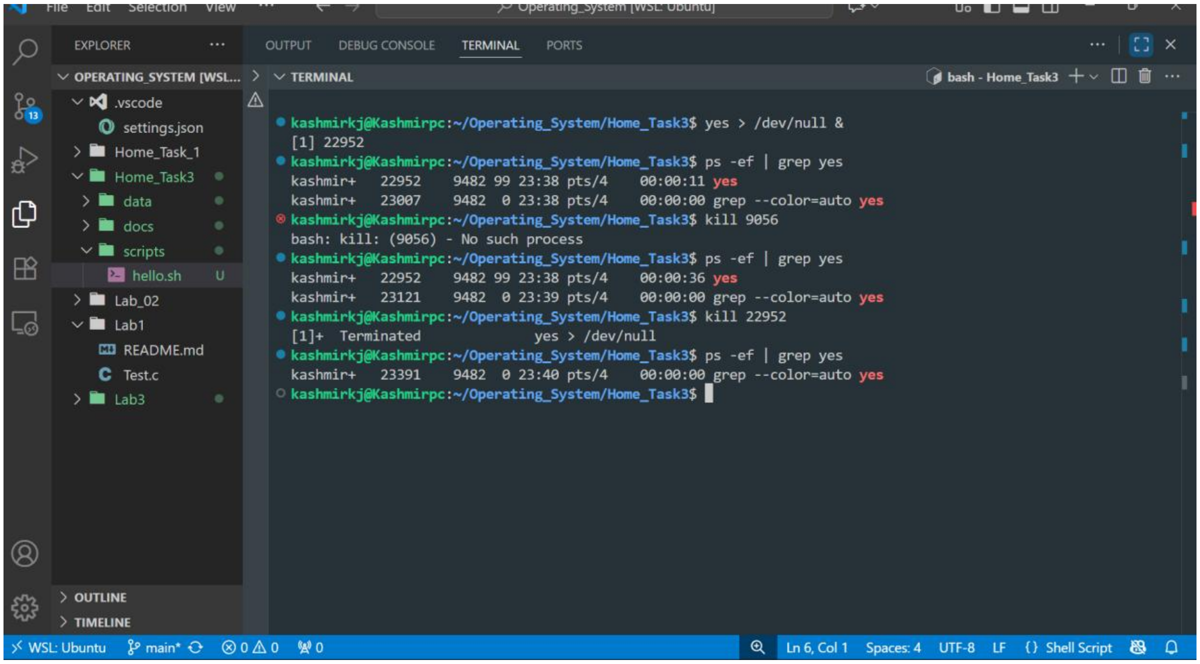


```
kashmirkj@Kashmirpc:~$ ls
F25OS  Operating_System  home_task_3  recycleBin
kashmirkj@Kashmirpc:~$ cd Operating_System/
kashmirkj@Kashmirpc:~/Operating_System$ ls
Home_Task3  Home_Task_1  Lab1  Lab3  Lab_02
kashmirkj@Kashmirpc:~/Operating_System$ cd Home_Task3
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3$ ps -ef
UID          PID    PPID  C STIME TTY          TIME CMD
root           1       0  0 22:02 ?        00:00:00 /sbin/init
root           2       1  0 22:02 ?        00:00:00 /init
root           8       2  0 22:02 ?        00:00:00 plan9 --control-socket 7 --log-level 4 --server-fd 8 --pi
root          41       1  0 22:02 ?        00:00:00 /usr/lib/systemd/systemd-journald
root          91       1  0 22:02 ?        00:00:00 /usr/lib/systemd/systemd-udevd
systemd+     113       1  0 22:02 ?        00:00:00 /usr/lib/systemd/systemd-resolved
systemd+     118       1  0 22:02 ?        00:00:00 /usr/lib/systemd/systemd-timesyncd
root         159       1  0 22:02 ?        00:00:00 /usr/sbin/cron -f -P
message+     160       1  0 22:02 ?        00:00:00 @dbus-daemon --system --address=systemd: --nofork --nopid
root         167       1  0 22:02 ?        00:00:00 /usr/lib/systemd/systemd-logind
root         169       1  0 22:02 ?        00:00:00 /usr/libexec/wsl-pro-service -vv
root         171       1  0 22:02 hvc0     00:00:00 /sbin/agetty -o -p -- \u --noclear --keep-baud - 115200,3
syslog       185       1  0 22:02 ?        00:00:00 /usr/sbin/rsyslogd -n -iNONE
root         191       1  0 22:02 tty1     00:00:00 /sbin/agetty -o -p -- \u --noclear - linux
root         197       1  0 22:02 ?        00:00:00 /usr/bin/python3 /usr/share/unattended-upgrades/unattende
root         285       2  0 22:02 ?        00:00:00 /init
root         286     285  0 22:02 ?        00:00:00 /init
kashmir+     287     286  0 22:02 pts/0    00:00:00 sh -c "$VSCODE_WSL_EXT_LOCATION/scripts/wslServer.sh" 385
root         288       2  0 22:02 pts/1    00:00:00 /bin/login -f
kashmir+     331       1  0 22:02 ?        00:00:00 /usr/lib/systemd/systemd --user
kashmir+     332     331  0 22:02 ?        00:00:00 (sd-pam)
kashmir+     354     288  0 22:02 pts/1    00:00:00 -bash
kashmir+     374     287  0 22:02 pts/0    00:00:00 sh /mnt/c/Users/Hp/.vscode/extensions/ms-vscode-remote.re
kashmir+     453     374  0 22:02 pts/0    00:00:00 sh /home/kashmirkj/.vscode-server/bin/385651c938df8a90686
kashmir+     457     453  0 22:02 pts/0    00:00:18 /home/kashmirkj/.vscode-server/bin/385651c938df8a906869ba
root         910       2  0 22:03 ?        00:00:00 /init
root         911     910  0 22:03 ?        00:00:00 /init
kashmir+     912     911  0 22:03 pts/2    00:00:02 /home/kashmirkj/.vscode-server/bin/385651c938df8a906869ba
root         919       2  0 22:03 ?        00:00:00 /init
root         920     919  0 22:03 ?        00:00:00 /init
kashmir+     921     920  0 22:03 pts/3    00:00:02 /home/kashmirkj/.vscode-server/bin/385651c938df8a906869ba
kashmir+     928     457  1 22:03 pts/0    00:01:46 /home/kashmirkj/.vscode-server/bin/385651c938df8a906869ba
kashmir+     947     457  0 22:03 pts/0    00:00:01 /home/kashmirkj/.vscode-server/bin/385651c938df8a906869ba
kashmir+     967     928  0 22:03 pts/0    00:00:02 /home/kashmirkj/.vscode-server/extensions/ms-vscode.cppto
```



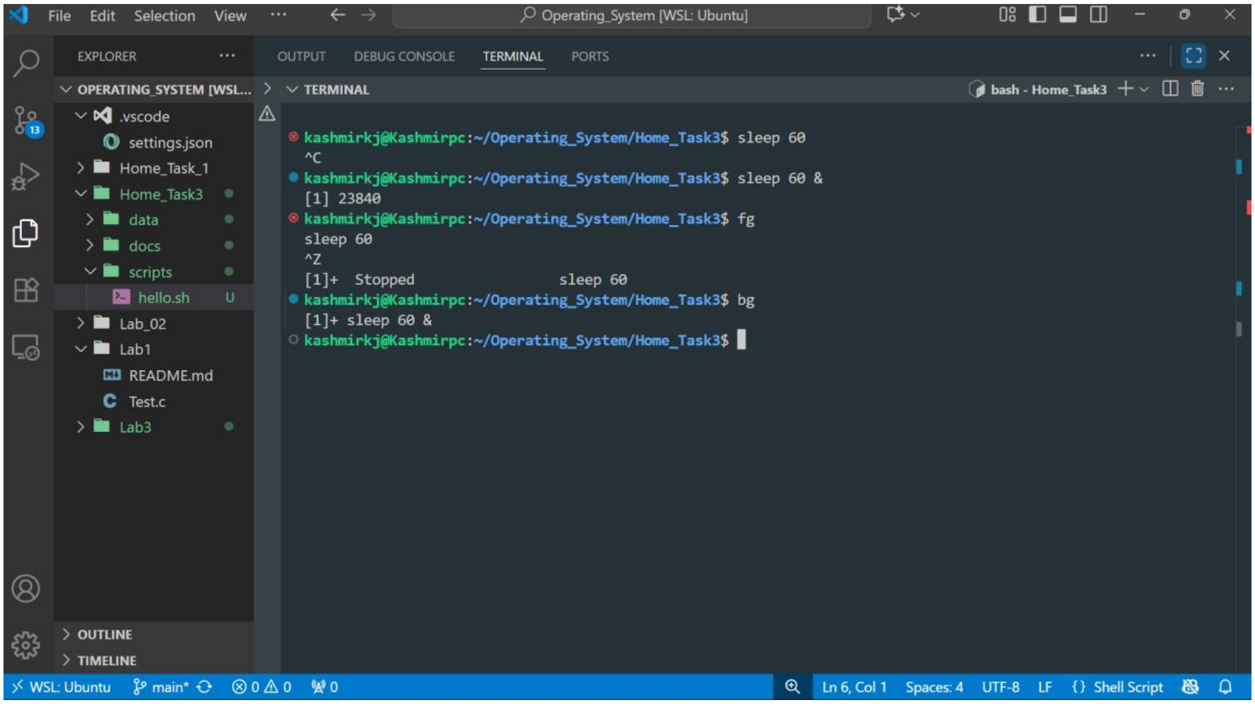
```
kashmir+    9482     980  0 22:37 pts/4    00:00:00 /bin/bash --init-file /home/kashmirkj/.vscode-server/bin/
kashmir+   22179    9482  0 23:35 pts/4    00:00:00 ps -ef
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3$ top
top - 23:37:04 up  1:34,  1 user,  load average: 0.04, 0.07, 0.02
Tasks:  41 total,   1 running,  40 sleeping,   0 stopped,   0 zombie
%Cpu(s):  0.1 us,  0.2 sy,  0.0 ni, 99.6 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
MiB Mem :   7784.2 total,   6144.7 free,   1532.8 used,    268.3 buff/cache
MiB Swap:   2048.0 total,   2048.0 free,      0.0 used.   6251.4 avail Mem

    PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM     TIME+ COMMAND
    928 kashmir+  20   0   52.7g 862148  63552 S   1.0  10.8   1:47.50 node
    457 kashmir+  20   0   11.3g 164056  53376 S   0.7   2.1   0:19.05 node
    980 kashmir+  20   0 1145692  86312  48256 S   0.7   1.1   0:10.20 node
    912 kashmir+  20   0 1020068  63636  44160 S   0.3   0.8   0:02.37 node
      1 root      20   0   21868  12496   9296 S   0.0   0.2   0:00.78 systemd
      2 root      20   0    3072   1792   1792 S   0.0   0.0   0:00.00 init-systemd(Ub
      8 root      20   0    3088   1792   1792 S   0.0   0.0   0:00.01 init
     41 root      19  -1   50436  15488  14720 S   0.0   0.2   0:00.46 systemd-journal
     91 root      20   0   25268   6400   4992 S   0.0   0.1   0:00.88 systemd-udevd
    113 systemd+  20   0   21456  12672  10624 S   0.0   0.2   0:00.13 systemd-resolve
    118 systemd+  20   0   91024   7680   6912 S   0.0   0.1   0:00.21 systemd-timesyn
    159 root      20   0    4236   2560   2432 S   0.0   0.0   0:00.02 cron
    160 message+  20   0    9660   4864   4480 S   0.0   0.1   0:00.26 dbus-daemon
    167 root      20   0   17968   8448   7552 S   0.0   0.1   0:00.13 systemd-logind
    169 root      20   0 1756096  12544  10240 S   0.0   0.2   0:00.21 wsl-pro-service
    171 root      20   0    3160   2048   1920 S   0.0   0.0   0:00.00 agetty
    185 syslog    20   0  222508   5760   4608 S   0.0   0.1   0:00.14 rsyslogd
    191 root      20   0    3116   1920   1792 S   0.0   0.0   0:00.00 agetty
    197 root      20   0  107032  22272  13184 S   0.0   0.3   0:00.09 unattended-upgr
    285 root      20   0    3080    896    896 S   0.0   0.0   0:00.00 SessionLeader
    286 root      20   0    3096   1156   1024 S   0.0   0.0   0:00.00 Relay(287)
    287 kashmir+  20   0    2800   1664   1664 S   0.0   0.0   0:00.00 sh
    288 root      20   0    6660   4608   3968 S   0.0   0.1   0:00.00 login
    331 kashmir+  20   0   20112  10752   9088 S   0.0   0.1   0:00.08 systemd
    332 kashmir+  20   0   21156   3520   1792 S   0.0   0.0   0:00.00 (sd-pam)
    354 kashmir+  20   0    6072   4864   3456 S   0.0   0.1   0:00.01 bash
    374 kashmir+  20   0    2800   1792   1792 S   0.0   0.0   0:00.00 sh
    453 kashmir+  20   0    2800   1792   1792 S   0.0   0.0   0:00.00 sh
    910 root      20   0    3080   1028    896 S   0.0   0.0   0:00.00 SessionLeader
    911 root      20   0    3096   1164   1024 S   0.0   0.0   0:00.73 Relay(912)
    919 root      20   0    3080   1028    896 S   0.0   0.0   0:00.00 SessionLeader
```

2. Practice with Infinite Process Start: yes > /dev/null & Locate its PID using ps -ef | grep yes . Kill it using kill and verify using



```
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3$ yes > /dev/null &
[1] 22952
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3$ ps -ef | grep yes
kashmir+   22952    9482 99 23:38 pts/4    00:00:11 yes
kashmir+   23007    9482  0 23:38 pts/4    00:00:00 grep --color=auto yes
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3$ kill 9056
bash: kill: (9056) - No such process
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3$ ps -ef | grep yes
kashmir+   22952    9482 99 23:38 pts/4    00:00:36 yes
kashmir+   23121    9482  0 23:39 pts/4    00:00:00 grep --color=auto yes
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3$ kill 22952
[1]+  Terminated              yes > /dev/null
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3$ ps -ef | grep yes
kashmir+   23391    9482  0 23:40 pts/4    00:00:00 grep --color=auto yes
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3$ 
```

3. Foreground & Background Jobs Run ps . sleep 60 in foreground and terminate it with Ctrl + C. Run sleep 60 & in background, bring it to foreground with then resume in background using bg . fg , stop with Ctrl + Z



```
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3$ sleep 60
^C
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3$ sleep 60 &
[1] 23840
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3$ fg
sleep 60
^Z
[1]+  Stopped                 sleep 60
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3$ bg
[1]+ sleep 60 &
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3$ 
```

# Part 6: C Programs on Processes

## Program 1 – Exec with top

1. Modify your existing exec program so that the **child process runs the top command** using one of the exec family functions (e.g., execlp).
2. Run the program.
3. In another terminal window, use the command:
4. top
   instead of:
   ps -ef | grep top
   to find the **child's process ID (PID)**.
5. Use the child's PID to manually kill the process using:
6. kill <PID>

### Program 2 – Incomplete Program

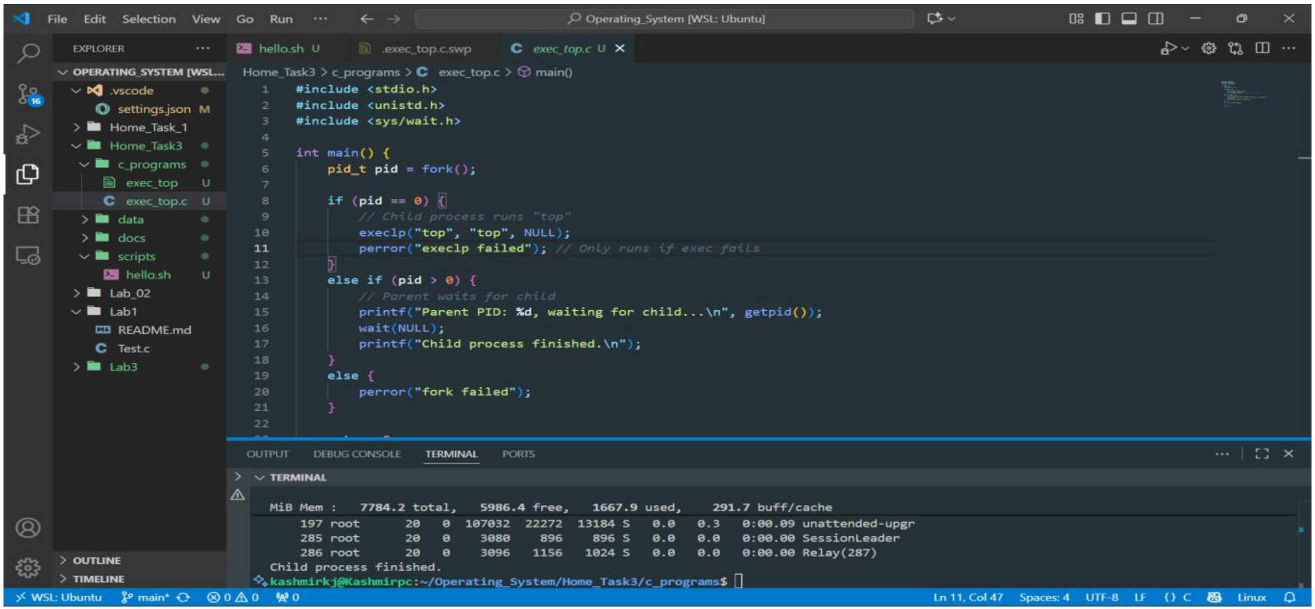
Complete the following C program by filling in the missing parts:

```
#include <stdio.h>
#include <unistd.h>
#include <sys/wait.h>

int main() {
   pid_t pid = fork();
```

```
  if (pid == 0) {
    // TODO: Replace this child process with the "date" command using execlp
    // Hint: execlp("date", "date", NULL);
  } else {
    // TODO: Make parent wait for child before printing "Child finished"
  }
  return 0;
}
```

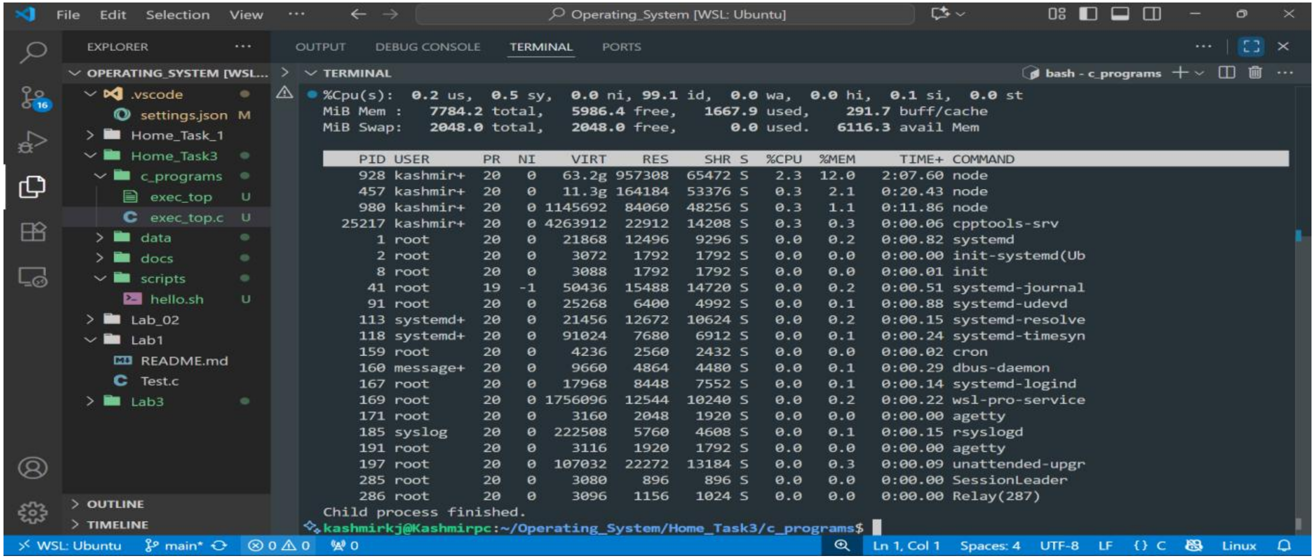
**Task:**

- Complete the missing parts of the program.
- Compile and run it.
- Take a **screenshot of the terminal output** showing:
    - The date command output (from the child process)
    - The "Child finished" message (from the parent process)



```c
#include <stdio.h>
#include <unistd.h>
#include <sys/wait.h>

int main() {
    pid_t pid = fork();

    if (pid == 0) {
        // Child process runs "top"
        execlp("top", "top", NULL);
        perror("execlp failed"); // Only runs if exec fails
    }
    else if (pid > 0) {
        // Parent waits for child
        printf("Parent PID: %d, waiting for child...\n", getpid());
        wait(NULL);
        printf("Child process finished.\n");
    }
    else {
        perror("fork failed");
    }

```

```
MiB Mem :   7784.2 total,   5986.4 free,   1667.9 used,    291.7 buff/cache
    197 root      20   0  107032  22272  13184 S   0.0   0.3   0:00.09 unattended-upgr
    285 root      20   0    3080    896    896 S   0.0   0.0   0:00.00 SessionLeader
    286 root      20   0    3096   1156   1024 S   0.0   0.0   0:00.00 Relay(287)
Child process finished.
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/c_programs$
```



```
%Cpu(s):  0.2 us,  0.5 sy,  0.0 ni, 99.1 id,  0.0 wa,  0.0 hi,  0.1 si,  0.0 st
MiB Mem :   7784.2 total,   5986.4 free,   1667.9 used,    291.7 buff/cache
MiB Swap:   2048.0 total,   2048.0 free,      0.0 used.   6116.3 avail Mem

    PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM     TIME+ COMMAND
    928 kashmir+  20   0   63.2g 957308  65472 S   2.3  12.0   2:07.60 node
    457 kashmir+  20   0   11.3g 164184  53376 S   0.3   2.1   0:20.43 node
    980 kashmir+  20   0 1145692  84060  48256 S   0.3   1.1   0:11.86 node
  25217 kashmir+  20   0 4263912  22912  14208 S   0.3   0.3   0:00.06 cpptools-srv
      1 root      20   0   21868  12496   9296 S   0.0   0.2   0:00.82 systemd
      2 root      20   0    3072   1792   1792 S   0.0   0.0   0:00.00 init-systemd(Ub
      8 root      20   0    3088   1792   1792 S   0.0   0.0   0:00.01 init
     41 root      19  -1   50436  15488  14720 S   0.0   0.2   0:00.51 systemd-journal
     91 root      20   0   25268   6400   4992 S   0.0   0.1   0:00.88 systemd-udevd
    113 systemd+  20   0   21456  12672  10624 S   0.0   0.2   0:00.15 systemd-resolve
    118 systemd+  20   0   91024   7680   6912 S   0.0   0.1   0:00.24 systemd-timesyn
    159 root      20   0    4236   2560   2432 S   0.0   0.0   0:00.02 cron
    160 message+  20   0    9660   4864   4480 S   0.0   0.1   0:00.29 dbus-daemon
    167 root      20   0   17968   8448   7552 S   0.0   0.1   0:00.14 systemd-logind
    169 root      20   0 1756096  12544  10240 S   0.0   0.2   0:00.22 wsl-pro-service
    171 root      20   0    3160   2048   1920 S   0.0   0.0   0:00.00 agetty
    185 syslog    20   0  222508   5760   4608 S   0.0   0.1   0:00.15 rsyslogd
    191 root      20   0    3116   1920   1792 S   0.0   0.0   0:00.00 agetty
    197 root      20   0  107032  22272  13184 S   0.0   0.3   0:00.09 unattended-upgr
    285 root      20   0    3080    896    896 S   0.0   0.0   0:00.00 SessionLeader
    286 root      20   0    3096   1156   1024 S   0.0   0.0   0:00.00 Relay(287)
Child process finished.
kashmirkj@Kashmirpc:~/Operating_System/Home_Task3/c_programs$
```