

1. Create a directory named Unix\_test.

**COMMAND:** `mkdir Unix_test`

2. Change to the Unix\_test directory.

**COMMAND:** `cd Unix_test`

3. Use a text editor (not cat) to create a text file named file1 in the Unix\_test directory, with the content:

**COMMAND:** `touch file1`

4. Copy file1 to file2.

**COMMAND:** `cp -r file1 file2`

5. List the contents of the current directory (Unix\_test).

**COMMAND:** `ls`

6. Change the permissions of file1 to allow read and write access from the owner, group and public. List the contents of the directory to show the new permissions.

**COMMAND:** `chmod u+g+x file1`

```
-rw-r--r-- 1 CPUT 197121 0 Aug  1 10:52 file1
```

```
-rw-r--r-- 1 CPUT 197121 0 Aug  1 10:39 file2
```

7. Delete file1. List the contents of the directory to show the effect.  
rm file1

**COMMAND:** `ls -l`

```
total 0
```

```
-rw-r--r-- 1 CPUT 197121 0 Aug  1 10:39 file2
```

8. Use head to display the top 3 lines of file2.

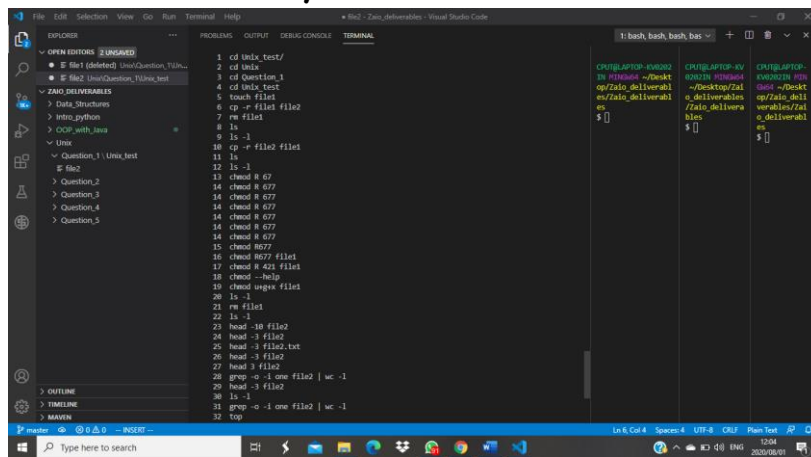
**COMMAND:** `head -3 file2`

9. Use cat, grep and wc to count the occurrences of "one" in file2.

**COMMAND:** `grep -o -i one file2 | wc -l`

10. Display all processes associated with the current user, split into pages so it does not scroll off the terminal.

# COMMAND: history



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

- EXPLORER:** Shows a file tree with folders like `DATA_DELIVERABLES`, `intro_python`, `OCOP_with_java`, and `Unix`. The `Unix` folder is expanded, showing subfolders like `Question_1`, `Question_2`, `Question_3`, `Question_4`, and `Question_5`.
- TERMINAL:** Displays a list of commands executed in a terminal session, numbered 1 through 32. The commands include file operations like `cd`, `touch`, `cp`, `rm`, `ls`, `cp -r`, `head`, `cat`, `grep`, and `top`.
- Terminal Windows:** Three terminal windows are open, each showing a prompt `$ []` and a cursor.