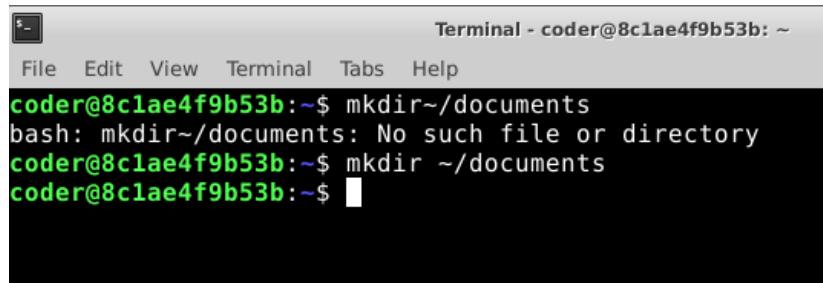


## Question 2.

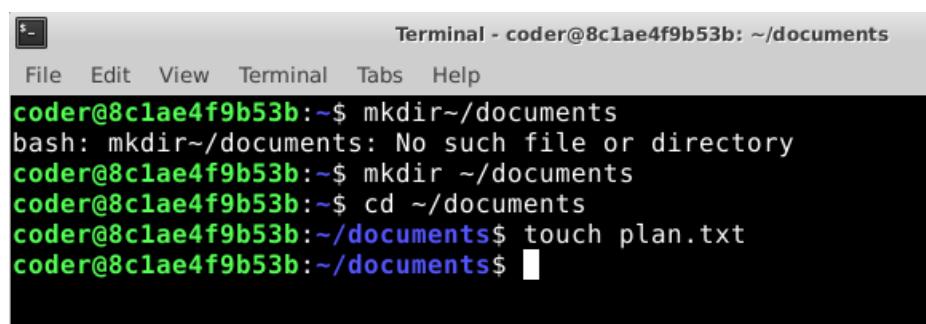
1. Command: mkdir ~/documents



```
Terminal - coder@8c1ae4f9b53b: ~
File Edit View Terminal Tabs Help
coder@8c1ae4f9b53b:~$ mkdir~/documents
bash: mkdir~/documents: No such file or directory
coder@8c1ae4f9b53b:~$ mkdir ~/documents
coder@8c1ae4f9b53b:~$ █
```

Explanation: *mkdir* creates a new directory. *~* is shorthand for your home directory (*/home/coder*). This gives you a clean folder called *documents* to store all project files.

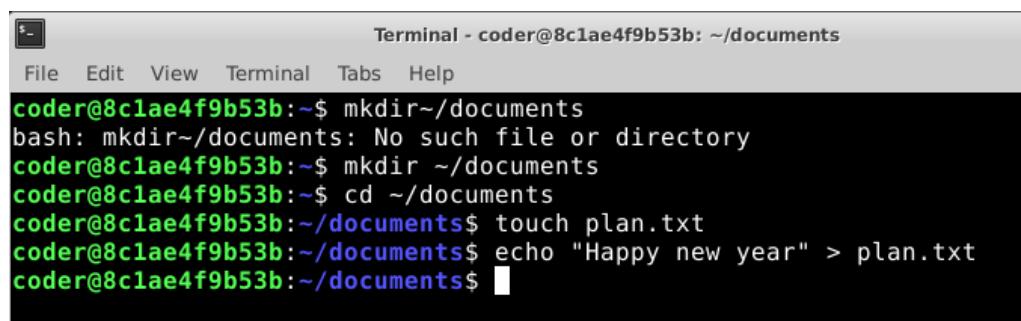
2. Command: cd ~/documents, touch plan.txt



```
Terminal - coder@8c1ae4f9b53b: ~/documents
File Edit View Terminal Tabs Help
coder@8c1ae4f9b53b:~$ mkdir~/documents
bash: mkdir~/documents: No such file or directory
coder@8c1ae4f9b53b:~$ mkdir ~/documents
coder@8c1ae4f9b53b:~$ cd ~/documents
coder@8c1ae4f9b53b:~/documents$ touch plan.txt
coder@8c1ae4f9b53b:~/documents$ █
```

Explanation: *cd* changes your current location to the new *documents* folder. *touch plan.txt* creates an empty file named *plan.txt*.

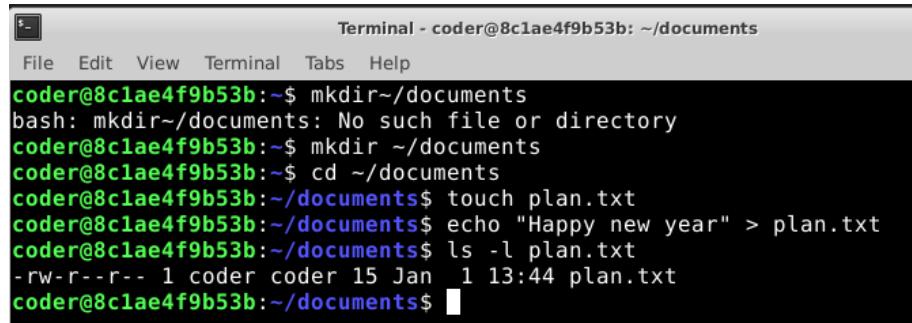
3. Command: echo "Happy new year" > plan.txt



```
Terminal - coder@8c1ae4f9b53b: ~/documents
File Edit View Terminal Tabs Help
coder@8c1ae4f9b53b:~$ mkdir~/documents
bash: mkdir~/documents: No such file or directory
coder@8c1ae4f9b53b:~$ mkdir ~/documents
coder@8c1ae4f9b53b:~$ cd ~/documents
coder@8c1ae4f9b53b:~/documents$ touch plan.txt
coder@8c1ae4f9b53b:~/documents$ echo "Happy new year" > plan.txt
coder@8c1ae4f9b53b:~/documents$ █
```

Explanation: *echo* prints the text, and *>* redirects it into *plan.txt*, overwriting the file with your chosen note. You can write any short reminder you like.

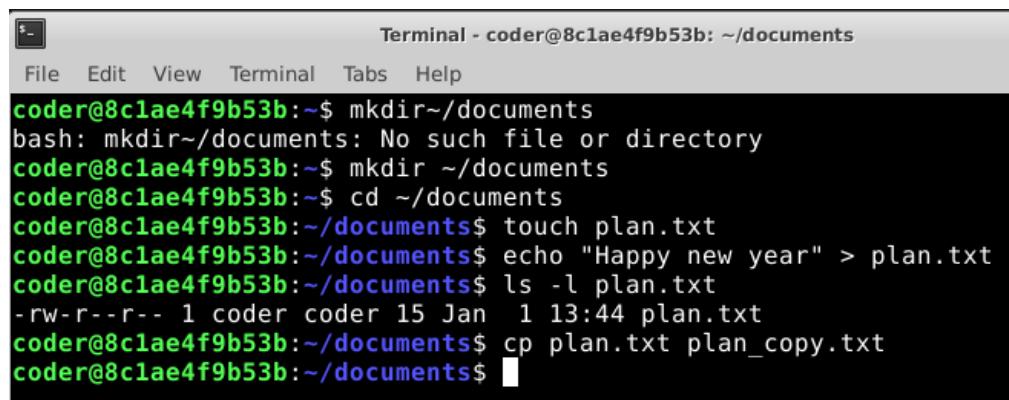
#### 4. Command: ls -l plan.txt



```
Terminal - coder@8c1ae4f9b53b: ~/documents
File Edit View Terminal Tabs Help
coder@8c1ae4f9b53b:~$ mkdir~/documents
bash: mkdir~/documents: No such file or directory
coder@8c1ae4f9b53b:~$ mkdir ~/documents
coder@8c1ae4f9b53b:~$ cd ~/documents
coder@8c1ae4f9b53b:~/documents$ touch plan.txt
coder@8c1ae4f9b53b:~/documents$ echo "Happy new year" > plan.txt
coder@8c1ae4f9b53b:~/documents$ ls -l plan.txt
-rw-r--r-- 1 coder coder 15 Jan 1 13:44 plan.txt
coder@8c1ae4f9b53b:~/documents$ █
```

Explanation: *ls -l* shows detailed (long) info: permissions (-rw-rw-r--), link count (1), owner (coder), group (coder), date/time, and filename. Your username coder must appear here as owner.

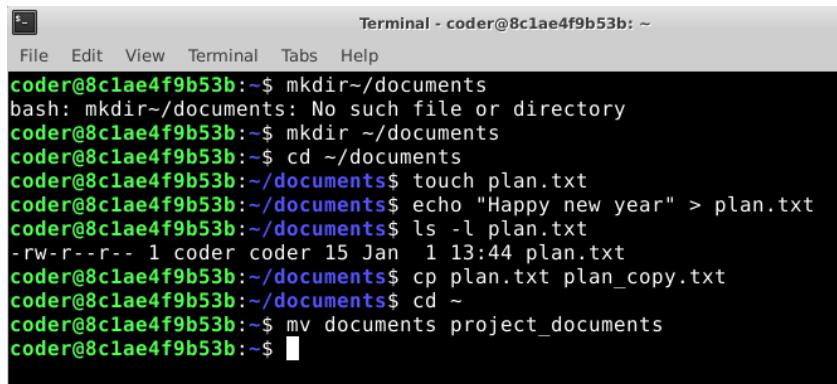
#### 5. Command: cp plan.txt plan\_copy.txt



```
Terminal - coder@8c1ae4f9b53b: ~/documents
File Edit View Terminal Tabs Help
coder@8c1ae4f9b53b:~$ mkdir~/documents
bash: mkdir~/documents: No such file or directory
coder@8c1ae4f9b53b:~$ mkdir ~/documents
coder@8c1ae4f9b53b:~$ cd ~/documents
coder@8c1ae4f9b53b:~/documents$ touch plan.txt
coder@8c1ae4f9b53b:~/documents$ echo "Happy new year" > plan.txt
coder@8c1ae4f9b53b:~/documents$ ls -l plan.txt
-rw-r--r-- 1 coder coder 15 Jan 1 13:44 plan.txt
coder@8c1ae4f9b53b:~/documents$ cp plan.txt plan_copy.txt
coder@8c1ae4f9b53b:~/documents$ █
```

Explanation: *cp* (copy) creates an identical duplicate named *plan\_copy.txt* in the same directory. Both files now have the same content.

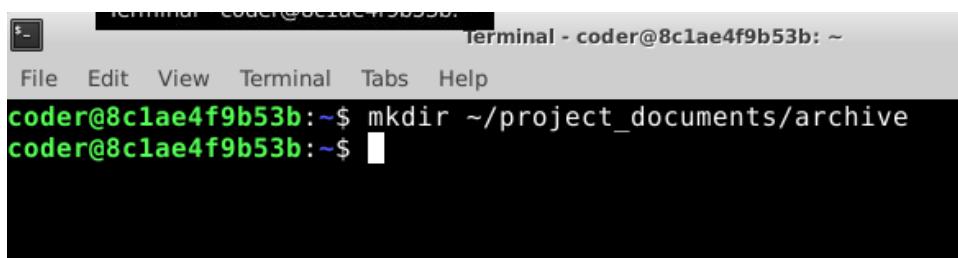
## 6. Command: cd ~, mv documents project\_documents



```
Terminal - coder@8c1ae4f9b53b: ~
File Edit View Terminal Tabs Help
coder@8c1ae4f9b53b:~$ mkdir ~/documents
bash: mkdir~/documents: No such file or directory
coder@8c1ae4f9b53b:~$ mkdir ~/documents
coder@8c1ae4f9b53b:~$ cd ~/documents
coder@8c1ae4f9b53b:~/documents$ touch plan.txt
coder@8c1ae4f9b53b:~/documents$ echo "Happy new year" > plan.txt
coder@8c1ae4f9b53b:~/documents$ ls -l plan.txt
-rw-r--r-- 1 coder coder 15 Jan 1 13:44 plan.txt
coder@8c1ae4f9b53b:~/documents$ cp plan.txt plan_copy.txt
coder@8c1ae4f9b53b:~/documents$ cd ~
coder@8c1ae4f9b53b:~$ mv documents project_documents
coder@8c1ae4f9b53b:~$
```

Explanation: *mv* (move) is also used for renaming. This renames the folder from *documents* to *project\_documents*. All files inside (plan.txt and plan\_copy.txt) automatically move with it.

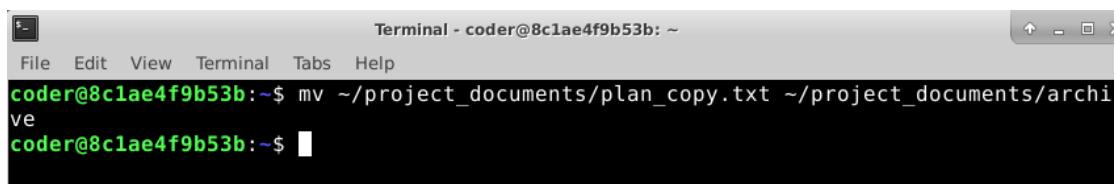
## 7. Command: mkdir ~/project\_documents/archive



```
Terminal - coder@8c1ae4f9b53b: ~
File Edit View Terminal Tabs Help
coder@8c1ae4f9b53b:~$ mkdir ~/project_documents/archive
coder@8c1ae4f9b53b:~$
```

Explanation: Creates a new subfolder called *archive* inside *project\_documents* – perfect for storing old or backup files.

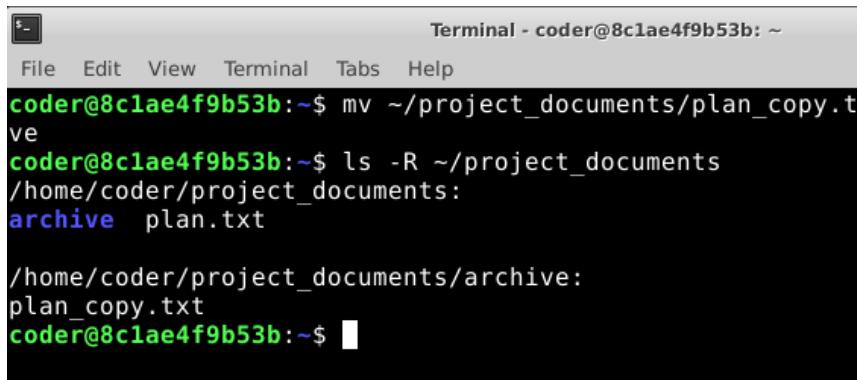
## 8. Command: mv ~/project\_documents/plan\_copy.txt ~/project\_documents/archive/



```
Terminal - coder@8c1ae4f9b53b: ~
File Edit View Terminal Tabs Help
coder@8c1ae4f9b53b:~$ mv ~/project_documents/plan_copy.txt ~/project_documents/archive/
coder@8c1ae4f9b53b:~$
```

Explanation: *mv* moves *plan\_copy.txt* from the main project folder into the *archive* subfolder, keeping things organized.

9. Command: ls -R ~/project\_documents

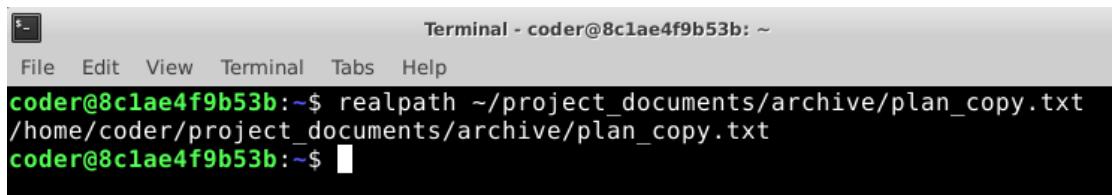


```
Terminal - coder@8c1ae4f9b53b: ~
File Edit View Terminal Tabs Help
coder@8c1ae4f9b53b:~$ mv ~/project_documents/plan_copy.txt archive
coder@8c1ae4f9b53b:~$ ls -R ~/project_documents
/home/coder/project_documents:
archive  plan.txt

/home/coder/project_documents/archive:
plan_copy.txt
coder@8c1ae4f9b53b:~$ █
```

Explanation: *ls -R* means “recursive”. It lists the contents of *project\_documents* and then drills down into every subdirectory (here, *archive*), showing the complete tree structure.

10. Command: realpath ~/project\_documents/archive/plan\_copy.txt



```
Terminal - coder@8c1ae4f9b53b: ~
File Edit View Terminal Tabs Help
coder@8c1ae4f9b53b:~$ realpath ~/project_documents/archive/plan_copy.txt
/home/coder/project_documents/archive/plan_copy.txt
coder@8c1ae4f9b53b:~$ █
```

Explanation: This prints the full absolute path of the moved file, confirming it is now located inside the archive folder.

These commands demonstrate core Linux file management skills: creating, copying, moving, renaming, and inspecting files/directories — all safely within your own user space.

