# Linux Basics

Date : 08-Oct-2025

Wednesday

LinkedIn

GitHub

---

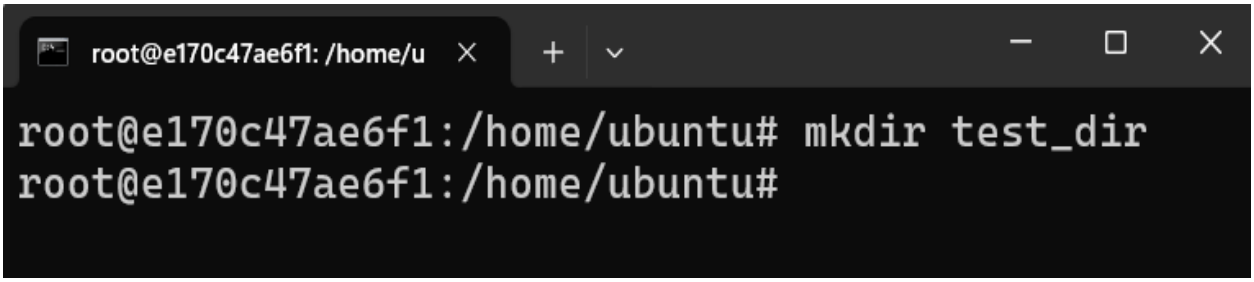
After Learning the Linux basics here is a small assignment to test myself.

It'll be going to be some commands from the class that I've taken.

## 1. Creating and renaming files/Directories

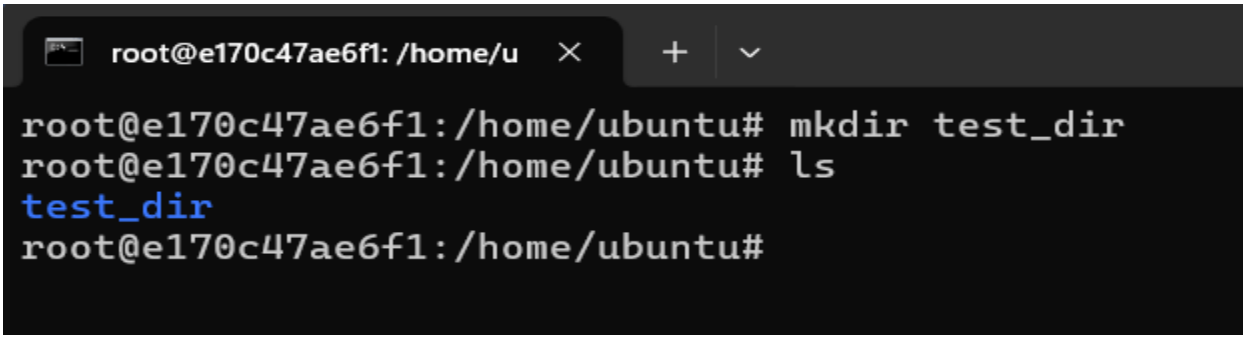
- Create a directory named *test_dir* using *mkdir*.

We are going to make a directory using the ***"mkdir"*** Command stand for Make Directory. If you are not familiar with the term Directory then the easy term for it is Folder 📂, that you have seen in your windows.




```
mkdir "dir_name"

mkdir test_dir
```

and using the command "***mkdir test_dir"*** we made ourself a directory.




By using the ***ls*** command we can see the list of files and directory.

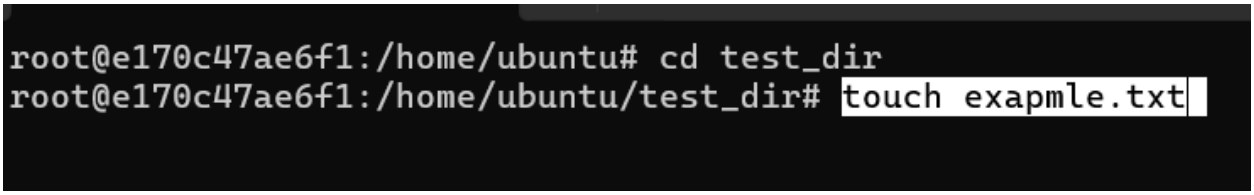
**Now we have made our "test_dir" directory**

---

- Inside *test_dir*, create a empty file called *example.txt*

Now before make a empty file, we first have to go inside the test_dir directory.

Using the ***cd*** command: ***"cd test_dir"***.




Now to make a text file we'll be using the ***touch*** command:

touch "file name"

`touch example.txt`

Now we have a file name *example.txt.*

---

- Rename *example.txt* to *renamed_example.txt* using *mv* command.

First we check the file using the *ls* command.

Now that we confirmed the file *example.txt*, now we will be renaming it.

Mv command's full form is Move, It's used to move a file/directory to different places.

```
root@e170c47ae6f1:/home/ubuntu/test_dir# ls
root@e170c47ae6f1:/home/ubuntu/test_dir# touch example.txt
root@e170c47ae6f1:/home/ubuntu/test_dir# ls
example.txt
root@e170c47ae6f1:/home/ubuntu/test_dir# mv example.txt renamed_example.txt
root@e170c47ae6f1:/home/ubuntu/test_dir#
```

```
mv “file_name” “destination_name”

mv example.txt renamed_example.txt
```

If there is no file/or directory of that name that you have given then it’ll make one and store it in.

Like we have given the destination name renamed_example.txt that file is there so it just made it right there.

```
root@e170c47ae6f1:/home/ubuntu/test_dir# ls
root@e170c47ae6f1:/home/ubuntu/test_dir# touch example.txt
root@e170c47ae6f1:/home/ubuntu/test_dir# ls
example.txt
root@e170c47ae6f1:/home/ubuntu/test_dir# mv example.txt renamed_example.txt
root@e170c47ae6f1:/home/ubuntu/test_dir#
root@e170c47ae6f1:/home/ubuntu/test_dir# ls
renamed_example.txt
root@e170c47ae6f1:/home/ubuntu/test_dir#
```

Now we check with ls for the file name.

As you can see the result, we’ve attained our goal.

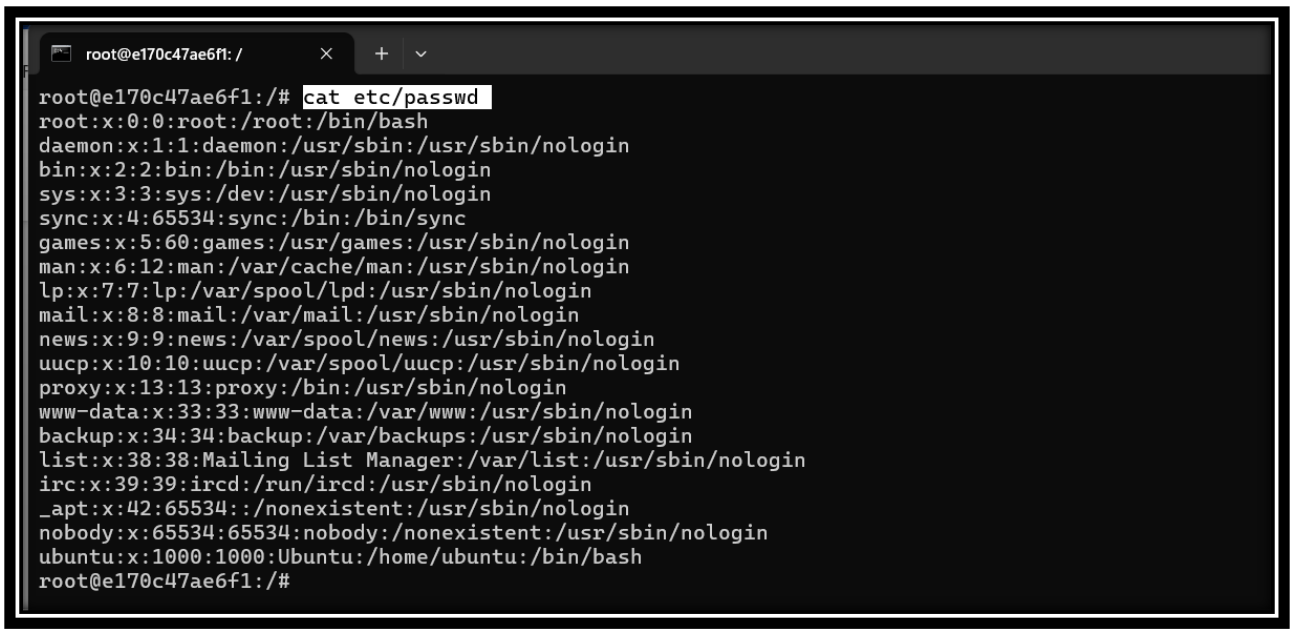
---

## 2. Viewing file content

- Use cat to display the content of /etc/passwd.

Cat command is used to view the content of a text file.

First we have to come to the `/`, we can come to it by just running ***cd*** command.

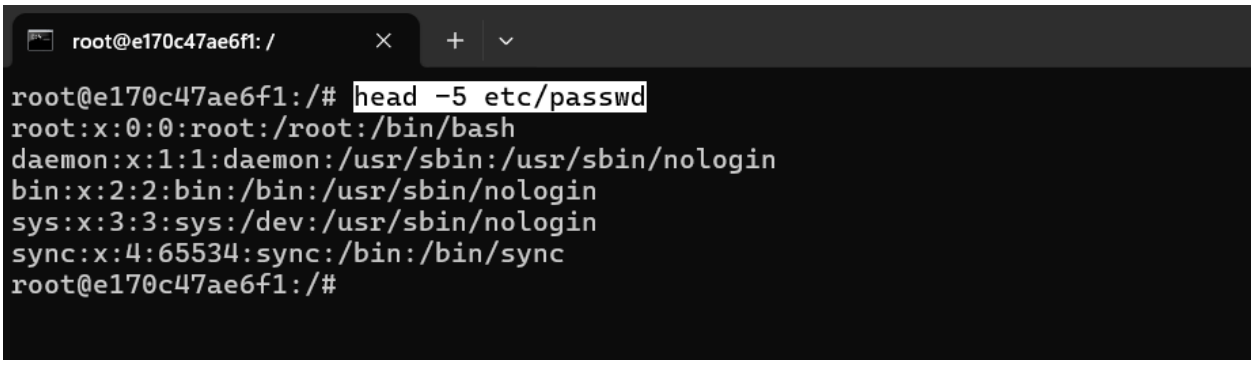



```
cat file_location/name

cat etc/passwd

etc : location    &    passwd : file name
```

The content written below *"cat etc/passwd"* is the content of the file "Passwd".

- Display only the first 5 lines of /etc/passwd using head.

The command used to show the starting line is ***head command***.



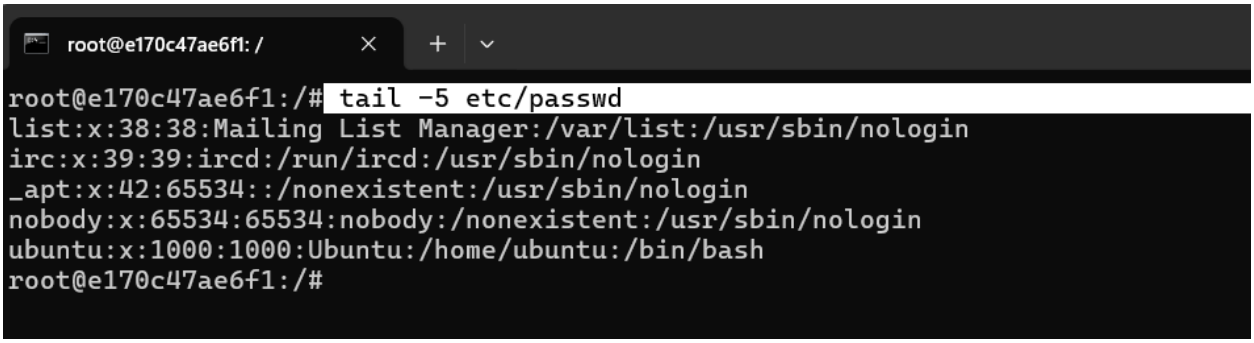

As you can see the head command shows the first 5 lines off the content.

```
head –“n” “location/name”

head -5 etc/passwd

etc/passwd : location and name of file

n : 5 {no. of lines}
```

---

- Display only the last 5 lines of /etc/passwd using tail.

The command used to show the ending line is ***tail command***.




Just like the head command tail command is used to find the last line of any file.

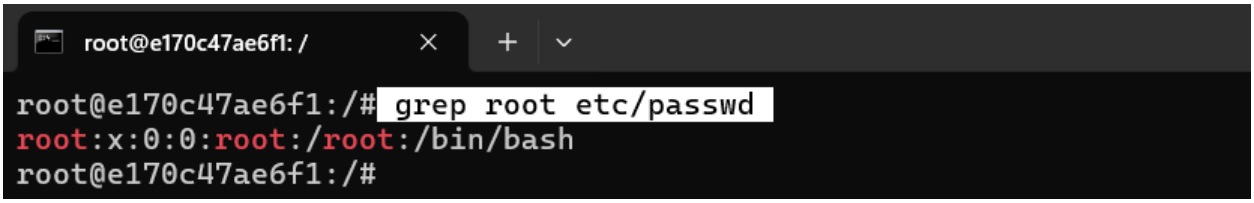
```
tail -5 etc/passwd
```

---

# 3. Searching for patterns

- Use grep to find all lines containing the word "root" in /etc/passwd.

*Grep command* is used to Find specific patterns, words, or lines in a file or output.



```
grep “word” “location/name”

grep “root” etc/passwd

word : root {the word to find}
```

By this we can find any word in the files, Generally used to find the words from the log file, for example “*grep error logfile.txt*” by this command the engineer can usually find the place of error.

And there is also a command for finding the exact number of the word is in.

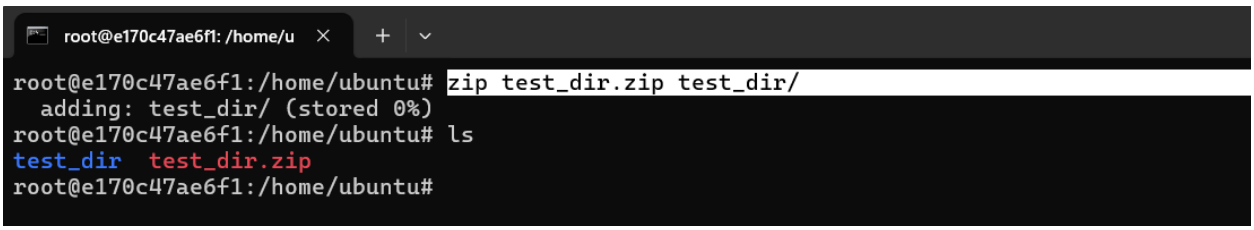

```
grep -n root etc/passwd

n : for showing the line number the
word is in, for now it’s line no. 1
```

---

# 4. Zipping and Unzipping

- Compress the test_dir directory into a file named test_dir.zip using zip.

Zip command in Linux used to compress the file and directories into the zip archive(with .zip extension).



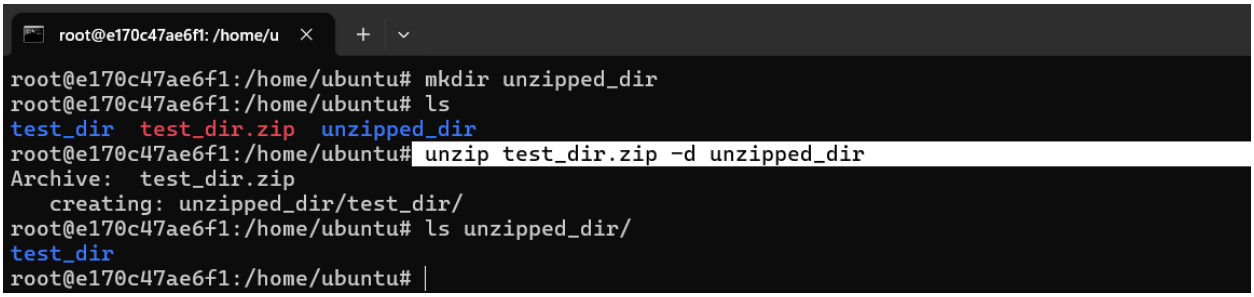
```
zip “new_file.zip” “file_name”

zip test_dir.zip test_dir

creates test_dir.zip containing test_dir
```

---

- `Unzip test_dir.zip into a new directory named unzipped_dir.`

Unzip is command used to unzip the `.zip` file.

Now we first have to make an *unzipped_dir* directory using `mkdir command` and then unzip the file inside the directory.




```
mkdir unzipped_dir

unzip “file” -d “location”

unzip test_dir.zip -d unzipped_dir

-d  : extract files into external dir.

first file name to be unzipped then -d to extract and then
location of directory in which it’ll be extraxted
```

If “-d” is not given then it will not unzip the file.

“-d” is used to extract in the external directory, if you want to extract in the same directory then you can just :

```
unzip test_dir.zip
```

---

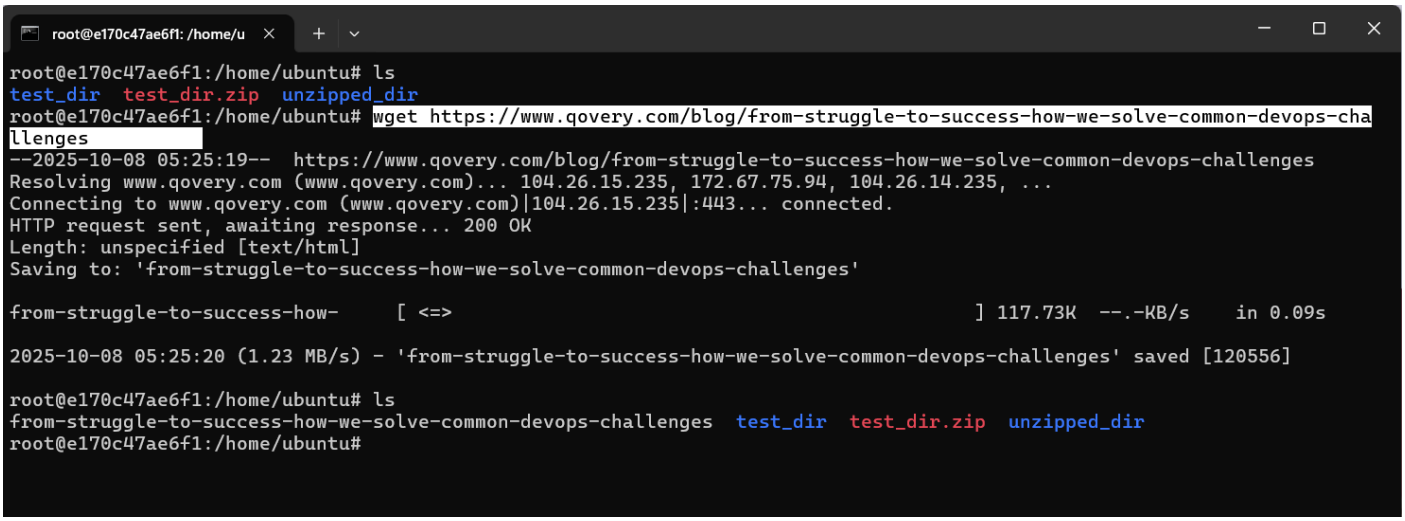
# 5. Downloading Files

- Use wget to download a file from a URL

wget is used to get the file from the internet.

It downloads the file, and there is a similar command that is "curl", it's fetches the text from the URL.

In this we are going to use the link { https://www.qovery.com/blog/from-struggle-to-success-how-we-solve-common-devops-challenges }




```
wget "link"

wget https://www.qovery.com/blog/from-struggle-to-success-how-we-solve-common-devops-challenges
```

In this wget command make get the details from the link and make a text file and stored the output in it.

# 6. Changing Permission

- Create a file nameed secure.txt and change it’s permission to read-only for everyone using chmod.

First we have to make an file named secure.txt, then we’ll change it’s permission to read only.

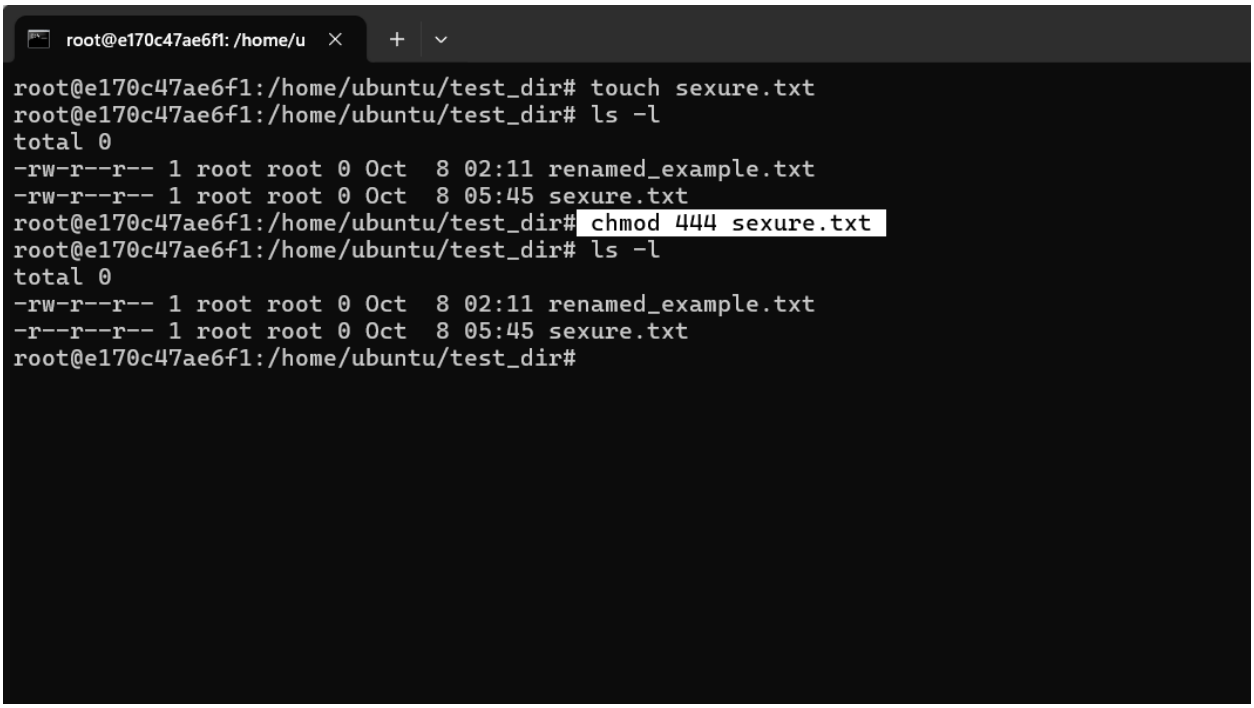
Chmod values are as follow :

**Read {r}:** 4

**Write {w}:** 2

**Execute {x}:** 1

```
3: execute + write || 5 : read + execute ||  6: read + write
```




```
chmod “value”  “file/dir_name”

chmod 444 secure.txt

After this the secure.txt permission is changed to read only
```

# 7. Working with Environment Variables

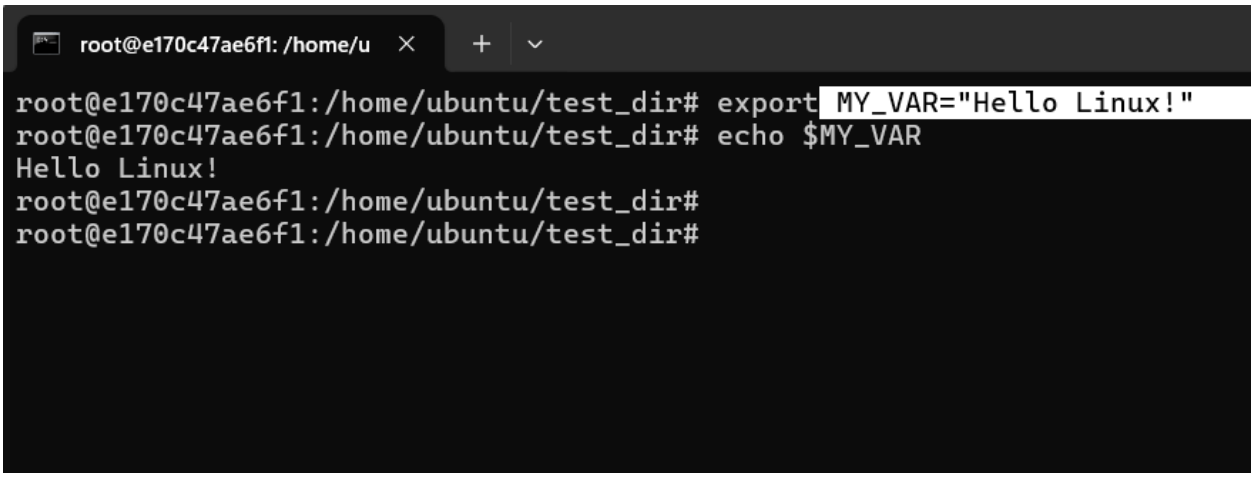
- Use export to set a new environment variable called my_var with the value “Hello Linux!”

“export” command is used to set the environment variables.

In easy word you can say that you are allowing value to the variable, often used for the frequently used commands.

Let’s assume I use man{manual} command often then what can I do is, I can set the variable mn=man, by this I can use the man command with the word of my preference.

Let’s give you variable a value:




```
export “variable”=“value”

export MY_VAR=“Hello Linux!”

Here variable is MY_VAR

And it’s value is “Hello Linux”
```