Topic Name:

The main aim of this lab session is to provide hands-on experience on

- Explore file structure
- File management commands
- Absolute path and Relative path
- Globbing
- Scripting

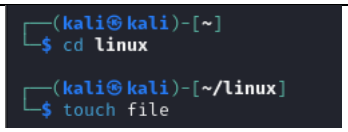
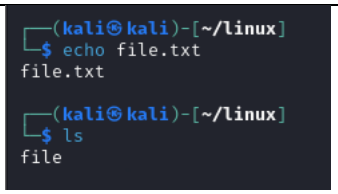
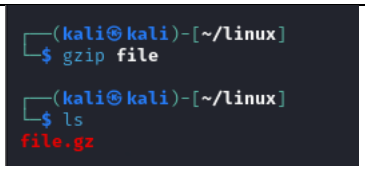
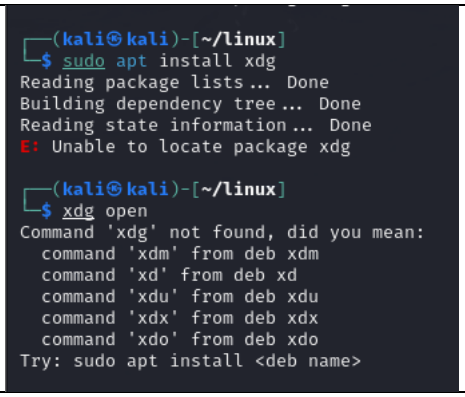
File Structure

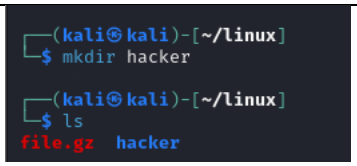
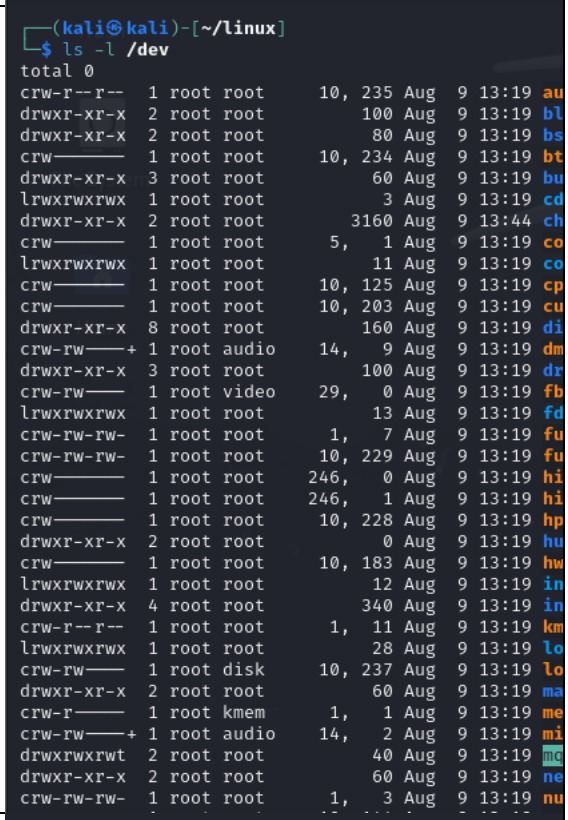
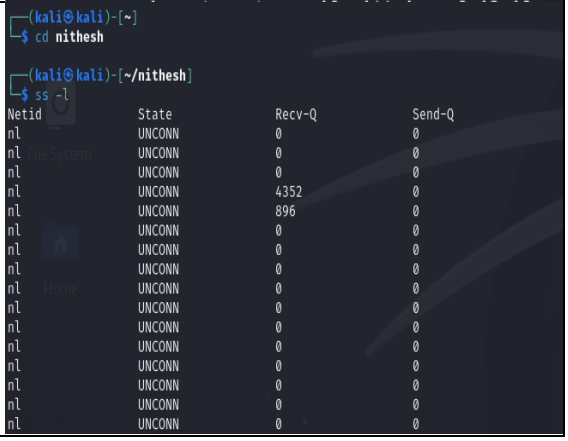
1. Under the root directory there are many files like
   /bin , /boot , /dev , /etc , ….
   Find out the importance of those files
   Example : /etc is for user account details

S.No	Directory	Usage
1	/	Root directory
2	/bin	Binary files ede
3	/boot	Contains all the files reqired for the linux boot process.
4	/dev	Special device files for all the devices.
5	/etc	Houses configuration files for system services and daemons.
6	/home	Personal workspace for each user.
7	/lib	Shared libraray images and kernel modules that are essential for booting the system and running commands in the root filesystem.
8	/proc	A virtual file system that provindes information about the systems kernel and process.
9	/sbin	System administatration commands and binaries that are primarly used by the system administrator.
10	/tmp	A temporary storage locations for files and directories that are created and accessed during system runtime.
11	/var	A standard directory that stores variable files,or filles that change frequently while the system is running.QVV
12		
13		

2. In Linux, there are three different files
   Regular file
   Directory
   Special file
      Block file
      Character file
         Socket file
         Pipe file

Fill the below table:

File Type	Represented by (Hintls)	Role	How to create	How to check	Location	Screen shot
Regular file	'-'	Stores data such as text	'Touch filename'	NA	Anywhere in file system.	
- Text file	.txt,.html,.csv.	Stores human readable text	Text	NA	Anywhere in file system.	
- Compressed file	.zip,.rar.	Stores data in compressed files	Compression utilities	NA	Anywhere in file system	
- Image	.jpg,.png,.gif.	Stores visual data	NA	NA	Anywhere in file system	

Directory	.(Dot)	Represents a Directory	Automaticallty created by file system	NA	Within File system	
Block file		Stores data in fixed size blocks	NA	NA	Storage device (HDD, SSD)	
Character file		Stores individual charecter	NA	NA	System level	
Socket file		Communicational endpoint	NA	NA	System level	

pipe file		Interprocess communication	NA	NA	System level	

```
┌──(kali㉿kali)-[~/nithesh]
└─$ mkfifo hacker

┌──(kali㉿kali)-[~/nithesh]
└─$ ls
CLI  example.txt  file  file_name  filename  hacker

┌──(kali㉿kali)-[~/nithesh]
└─$ ls -l
total 16
drwxr-xr-x 4 kali kali 4096 Aug  6 13:22 CLI
-rw-r--r-- 1 kali kali    0 Aug  6 12:30 example.txt
prw-r--r-- 1 kali kali    0 Aug  9 14:08 file
-rw-r--r-- 1 kali kali    0 Aug  6 12:38 file_name
-rw-r--r-- 1 kali kali    0 Aug  5 12:34 filename
prw-r--r-- 1 kali kali    0 Aug  9 14:09 hacker
drwxr-xr-x 2 kali kali 4096 Aug  6 14:06 new_folder
drwxr-xr-x 2 kali kali 4096 Aug  6 11:11 nithesh
drwxr-xr-x 3 kali kali 4096 Aug  6 14:06 parent_fold
prw-r--r-- 1 kali kali    0 Aug  9 14:08 pipefile
prw-r--r-- 1 kali kali    0 Aug  9 14:08 pipefile.gz
-rw-r--r-- 1 kali kali    0 Aug  6 12:54 time
```

3. Globbing : **Globbing** is a pattern matching mechanism used to specify a set of filenames. It's particularly useful for working with multiple files at once.
4. Go back to CYS

```
┌──(kali㉿kali)-[/home/kali/nithesh]
└─PS> cd cys
```

5. Create multiple subdirectories using single command

```
┌──(kali㉿kali)-[/home/kali/nithesh/cys]
└─PS> touch file1 file2 file3
```

6. LS

```
┌──(kali㉿kali)-[/home/kali/nithesh/cys]
└─PS> ls
unit1  unit2  unit3
```

7. Unit1
8. command
9. Glob

```
┌──(kali㉿kali)-[/home/kali/nithesh/cys]
└─PS> bash unit[1-3]
unit1: unit1: Is a directory
```

10. Unit2
11. command
12. Grep

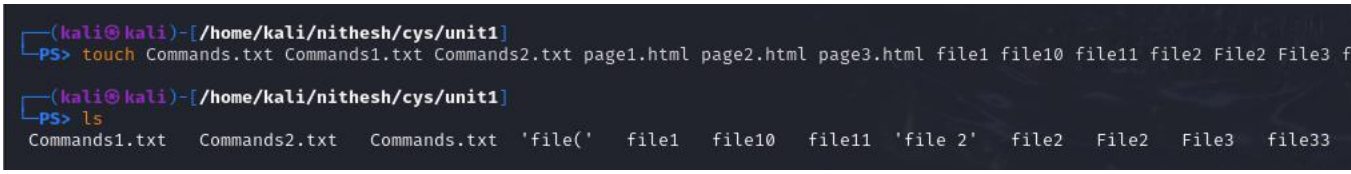
```
┌──(kali㉿kali)-[/home/kali/nithesh/cys]
└─PS> grep "pattern" unit2
/usr/bin/grep: unit2: Is a directory
```

13. Unit3
14. Constructs

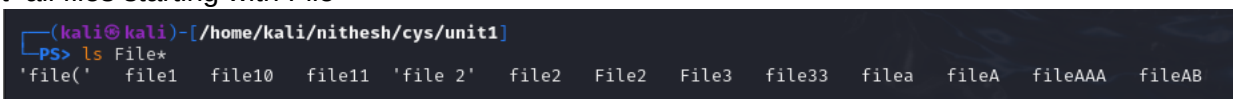
```
┌──(kali㉿kali)-[/home/kali/nithesh/cys]
└─PS> cd.. Unit3

┌──(kali㉿kali)-[/home/kali/nithesh]
└─PS> echo "Hello, World!"
Hello, World!
```

15.
16.
17. Navigate to unit1/glob

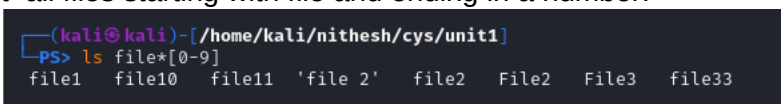
```
┌──(kali㉿kali)-[/home/kali/nithesh/cys]
└─PS> cd unit1

┌──(kali㉿kali)-[/home/kali/nithesh/cys/unit1]
└─PS> cd unit1/glob
```

18. Create the following files :
19. Commands.txt
20. Commands1.txt
21. Commands2.txt
22. page1.html
23. page2.html
24. page3.html
25. file1
26. file10
27. file11
28. file2
29. File2
30. File3
31. file33
32. fileAB
33. filea
34. fileA
35. fileAAA
36. file(
37. file 2
38. 

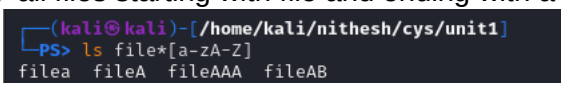
39. List all files starting with file
40. List all files starting with File



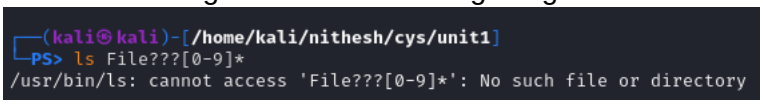
41. List all files starting with file and ending in a number.



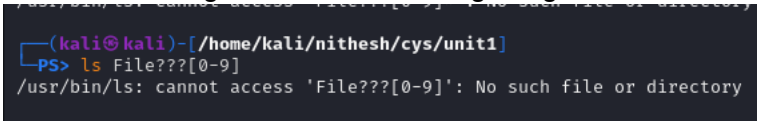
42. List all files starting with file and ending with a letter



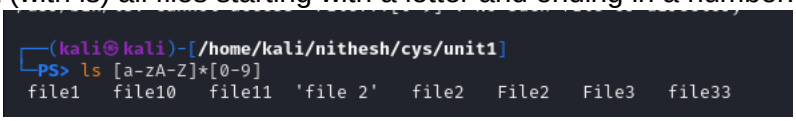
43. List all files starting with File and having a digit as fifth character.



44. List all files starting with File and having a digit as fifth character and nothing else.



45. List (with ls) all files starting with a letter and ending in a number.



46. List (with ls) all files that have exactly five characters.

```
┌──(kali㉿kali)-[/home/kali/nithesh/cys/unit1]
└─PS> ls ?????
'file('   file1   file2   File2   File3   filea   fileA
```

47. List (with ls) all files that start with f or F and end with 3 or A.

```
┌──(kali㉿kali)-[/home/kali/nithesh/cys/unit1]
└─PS> ls [fF]*[3A]
File3  file33  filea  fileA  fileAAA
```

48. List (with ls) all files that start with f have i or R as second character and end in a number.

```
┌──(kali㉿kali)-[/home/kali/nithesh/cys/unit1]
└─PS> ls f[iR]*[0-9]
file1   file10   file11  'file 2'   file2   File2   File3   file33
```

49. List all files that do not start with the letter F.

```
┌──(kali㉿kali)-[/home/kali/nithesh/cys/unit1]
└─PS> ls | grep -v '^F'
Commands1.txt
Commands2.txt
Commands.txt
file(
file1
file10
file11
file 2
file2
file33
filea
fileA
fileAAA
fileAB
page1.html
page2.html
page3.html
```

50. Remove all the *.html

```
┌──(kali㉿kali)-[/home/kali/nithesh/cys/unit1]
└─PS> rm *.html

┌──(kali㉿kali)-[/home/kali/nithesh/cys/unit1]
└─PS> rm *.html
/usr/bin/rm: cannot remove '*.html': No such file or directory

┌──(kali㉿kali)-[/home/kali/nithesh/cys/unit1]
└─PS> ls
Commands1.txt   Commands2.txt   Commands.txt  'file('   file1   file10   file11  'file 2'   file2   File2   File3   file3
```

51. Rename *.txt to *.json
52.
    a. Go back to CYS

```
┌──(kali㉿kali)-[/home/kali/nithesh/cys/unit1]
└─PS> cd ..

┌──(kali㉿kali)-[/home/kali/nithesh/cys]
```

    b. Create multiple subdirectories using single command

LS
Unit1
command
glob
Unit2
command
grep
Unit3
constructs

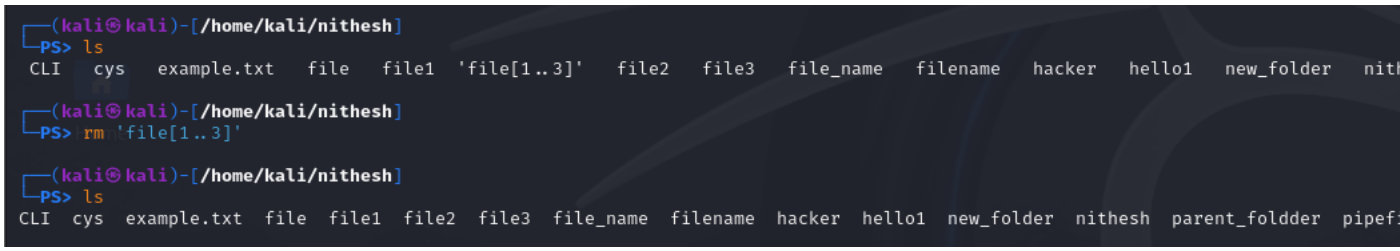
    c. Navigate to unit1/glob

d. Create the following files :
Commands.txt
Commands1.txt
Commands2.txt
page1.html
page2.html
page3.html
file1
file10
file11
file2
File2
File3
file33
fileAB
filea
fileA
fileAAA
file(
file 2

i. List all files starting with file
ii. List all files starting with File
iii. List all files starting with file and ending in a number.
iv. List all files starting with file and ending with a letter
v. List all files starting with File and having a digit as fifth character.
vi. List all files starting with File and having a digit as fifth character and nothing else.
vii. List (with ls) all files starting with a letter and ending in a number.
viii. List (with ls) all files that have exactly five characters.
ix. List (with ls) all files that start with f or F and end with 3 or A.
x. List (with ls) all files that start with f have i or R as second character and end in a number.
xi. List all files that do not start with the letter F.
xii. Remove all the *.html
xiii. Rename *.txt to *.json

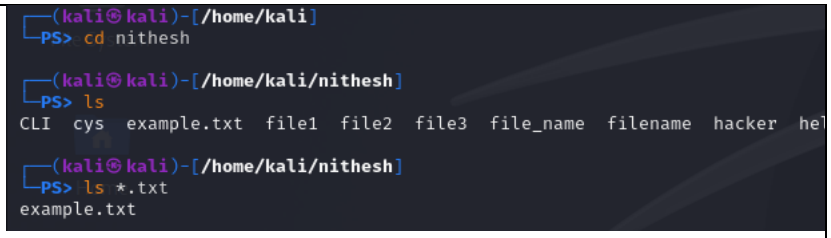
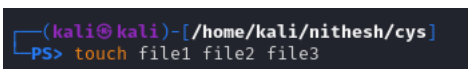
53. Absolute path and relative path

Use rm, mv, cp, ls with absolute path and relative path as per your choice.



54. Wildcards

Notation	Use	Example	Screenshot

*	Matches any number of characters	`ls *.txt` will list all	
?	Matches exactly one character	`ls file?.txt`	
[ ]	Matches any one character within the brackets	ls file[123].txt	
[!]	Matches any character that is not a member of the set characters	ls file[!123].txt	
{ }	Matches any string within	ls file{1,2,3}	

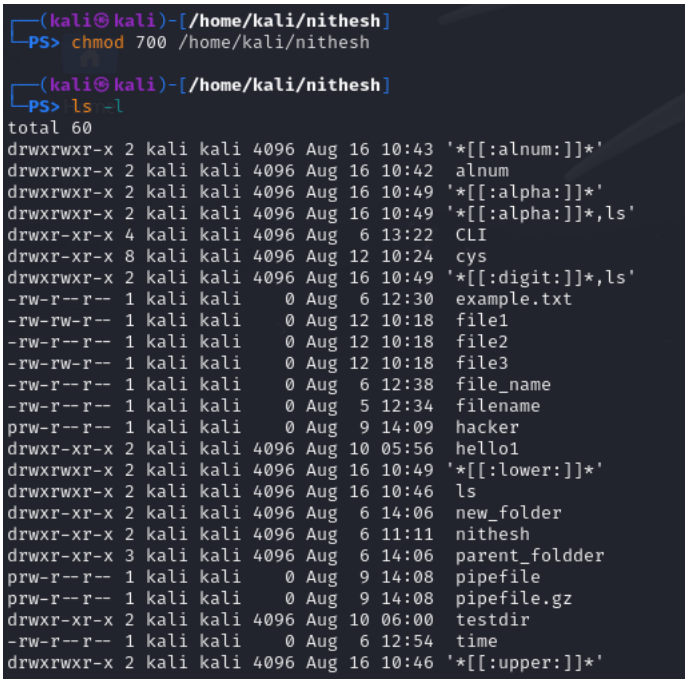
	the brac es		

More on Character class

Notation	Use	Example	Screenshot
[:alnum:]	Matches any alphanumeric character	ls *[[:alnum:]]*	
[:alpha:]	Matches any alphabetic character	ls *[[:alpha:]]*	
[:digit:]	Matches any digit (0-9)	ls *[[:digit:]]*	
[:lower:]	Matches any lowercase letter	ls *[[:lower:]]*	
[:upper:]	Matches any uppercase letter	ls *[[:upper:]]*	

4. change permission

a) Change the permission set of /work/readme.txt so that only the user (owner) can read,write, and execute it. Use absolute mode.



b) Change the permission set of /work/readme.txt so that any user can read it, the group can read/write to it and the user (owner) can read/write/execute it. Use absolute mode.

```
┌──(kali㉿kali)-[/home/kali/nithesh]
└─PS> chmod 764 /home/kali/nithesh

┌──(kali㉿kali)-[/home/kali/nithesh]
└─PS> ls -l
total 60
drwxrwxr-x 2 kali kali 4096 Aug 16 10:43 '*[[:alnum:]]*'
drwxrwxr-x 2 kali kali 4096 Aug 16 10:42  alnum
drwxrwxr-x 2 kali kali 4096 Aug 16 10:49 '*[[:alpha:]]*'
drwxrwxr-x 2 kali kali 4096 Aug 16 10:49 '*[[:alpha:]]*,ls'
drwxr-xr-x 4 kali kali 4096 Aug  6 13:22  CLI
drwxr-xr-x 8 kali kali 4096 Aug 12 10:24  cys
drwxrwxr-x 2 kali kali 4096 Aug 16 10:49 '*[[:digit:]]*,ls'
-rw-r--r-- 1 kali kali    0 Aug  6 12:30  example.txt
-rw-rw-r-- 1 kali kali    0 Aug 12 10:18  file1
-rw-r--r-- 1 kali kali    0 Aug 12 10:18  file2
-rw-rw-r-- 1 kali kali    0 Aug 12 10:18  file3
-rw-r--r-- 1 kali kali    0 Aug  6 12:38  file_name
-rw-r--r-- 1 kali kali    0 Aug  5 12:34  filename
prw-r--r-- 1 kali kali    0 Aug  9 14:09  hacker
drwxr-xr-x 2 kali kali 4096 Aug 10 05:56  hello1
drwxrwxr-x 2 kali kali 4096 Aug 16 10:49 '*[[:lower:]]*'
drwxrwxr-x 2 kali kali 4096 Aug 16 10:46  ls
drwxr-xr-x 2 kali kali 4096 Aug  6 14:06  new_folder
drwxr-xr-x 2 kali kali 4096 Aug  6 11:11  nithesh
drwxr-xr-x 3 kali kali 4096 Aug  6 14:06  parent_foldder
prw-r--r-- 1 kali kali    0 Aug  9 14:08  pipefile
prw-r--r-- 1 kali kali    0 Aug  9 14:08  pipefile.gz
drwxr-xr-x 2 kali kali 4096 Aug 10 06:00  testdir
-rw-r--r-- 1 kali kali    0 Aug  6 12:54  time
drwxrwxr-x 2 kali kali 4096 Aug 16 10:46 '*[[:upper:]]*'
```

c) Change the permission set of /bin/bash so that only the user (owner) can read/write/ execute, group, and any user can execute it. However, whenever anyone executes it, it should run with the privileges of the owner user. Use absolute mode.

```
┌──(kali㉿kali)-[/home/kali/nithesh]
└─PS> chmod 551 /home/kali/nithesh

┌──(kali㉿kali)-[/home/kali/nithesh]
└─PS> ls -l
total 60
drwxrwxr-x 2 kali kali 4096 Aug 16 10:43 '*[[:alnum:]]*'
drwxrwxr-x 2 kali kali 4096 Aug 16 10:42  alnum
drwxrwxr-x 2 kali kali 4096 Aug 16 10:49 '*[[:alpha:]]*'
drwxrwxr-x 2 kali kali 4096 Aug 16 10:49 '*[[:alpha:]]*,ls'
drwxr-xr-x 4 kali kali 4096 Aug  6 13:22  CLI
drwxr-xr-x 8 kali kali 4096 Aug 12 10:24  cys
drwxrwxr-x 2 kali kali 4096 Aug 16 10:49 '*[[:digit:]]*,ls'
-rw-r--r-- 1 kali kali    0 Aug  6 12:30  example.txt
-rw-rw-r-- 1 kali kali    0 Aug 12 10:18  file1
-rw-r--r-- 1 kali kali    0 Aug 12 10:18  file2
-rw-rw-r-- 1 kali kali    0 Aug 12 10:18  file3
-rw-r--r-- 1 kali kali    0 Aug  6 12:38  file_name
-rw-r--r-- 1 kali kali    0 Aug  5 12:34  filename
prw-r--r-- 1 kali kali    0 Aug  9 14:09  hacker
drwxr-xr-x 2 kali kali 4096 Aug 10 05:56  hello1
drwxrwxr-x 2 kali kali 4096 Aug 16 10:49 '*[[:lower:]]*'
drwxrwxr-x 2 kali kali 4096 Aug 16 10:46  ls
drwxr-xr-x 2 kali kali 4096 Aug  6 14:06  new_folder
drwxr-xr-x 2 kali kali 4096 Aug  6 11:11  nithesh
drwxr-xr-x 3 kali kali 4096 Aug  6 14:06  parent_foldder
prw-r--r-- 1 kali kali    0 Aug  9 14:08  pipefile
prw-r--r-- 1 kali kali    0 Aug  9 14:08  pipefile.gz
drwxr-xr-x 2 kali kali 4096 Aug 10 06:00  testdir
-rw-r--r-- 1 kali kali    0 Aug  6 12:54  time
drwxrwxr-x 2 kali kali 4096 Aug 16 10:46 '*[[:upper:]]*'
```

d) Change the permission set of /work/readme.txt so that only the user (owner) can read, write, and execute it. Use relative mode.

```
┌──(kali㉿kali)-[/home/kali/nithesh]
└─PS> ls -l
total 60
drwxrwxr-x 2 kali kali 4096 Aug 16 10:43 '*[[:alnum:]]*'
drwxrwxr-x 2 kali kali 4096 Aug 16 10:42  alnum
drwxrwxr-x 2 kali kali 4096 Aug 16 10:49 '*[[:alpha:]]*'
drwxrwxr-x 2 kali kali 4096 Aug 16 10:49 '*[[:alpha:]]*,ls'
drwxr-xr-x 4 kali kali 4096 Aug  6 13:22  CLI
drwxr-xr-x 8 kali kali 4096 Aug 12 10:24  cys
drwxrwxr-x 2 kali kali 4096 Aug 16 10:49 '*[[:digit:]]*,ls'
-rw-r--r-- 1 kali kali    0 Aug  6 12:30  example.txt
-rw-rw-r-- 1 kali kali    0 Aug 12 10:18  file1
-rw-r--r-- 1 kali kali    0 Aug 12 10:18  file2
-rw-rw-r-- 1 kali kali    0 Aug 12 10:18  file3
-rw-r--r-- 1 kali kali    0 Aug  6 12:38  file_name
-rw-r--r-- 1 kali kali    0 Aug  5 12:34  filename
prw-r--r-- 1 kali kali    0 Aug  9 14:09  hacker
drwxr-xr-x 2 kali kali 4096 Aug 10 05:56  hello1
drwxrwxr-x 2 kali kali 4096 Aug 16 10:49 '*[[:lower:]]*'
drwxrwxr-x 2 kali kali 4096 Aug 16 10:46  ls
drwxr-xr-x 2 kali kali 4096 Aug  6 14:06  new_folder
drwxr-xr-x 2 kali kali 4096 Aug  6 11:11  nithesh
drwxr-xr-x 3 kali kali 4096 Aug  6 14:06  parent_foldder
prw-r--r-- 1 kali kali    0 Aug  9 14:08  pipefile
prw-r--r-- 1 kali kali    0 Aug  9 14:08  pipefile.gz
drwxr-xr-x 2 kali kali 4096 Aug 10 06:00  testdir
-rw-r--r-- 1 kali kali    0 Aug  6 12:54  time
drwxrwxr-x 2 kali kali 4096 Aug 16 10:46 '*[[:upper:]]*'
```

e) Change the permission set of /work/readme.txt so that any user can read it, the group can read/write to it and the user (owner) can read/write/execute it. Use relative mode.

```
┌──(kali㉿kali)-[/home/kali/nithesh]
└─PS> chmod u=rwx,g=rw,o=r /home/kali/nithesh

┌──(kali㉿kali)-[/home/kali/nithesh]
└─PS> ls -l
total 60
drwxrwxr-x 2 kali kali 4096 Aug 16 10:43 '*[[:alnum:]]*'
drwxrwxr-x 2 kali kali 4096 Aug 16 10:42  alnum
drwxrwxr-x 2 kali kali 4096 Aug 16 10:49 '*[[:alpha:]]*'
drwxrwxr-x 2 kali kali 4096 Aug 16 10:49 '*[[:alpha:]]*,ls'
drwxr-xr-x 4 kali kali 4096 Aug  6 13:22  CLI
drwxr-xr-x 8 kali kali 4096 Aug 12 10:24  cys
drwxrwxr-x 2 kali kali 4096 Aug 16 10:49 '*[[:digit:]]*,ls'
-rw-r--r-- 1 kali kali    0 Aug  6 12:30  example.txt
-rw-rw-r-- 1 kali kali    0 Aug 12 10:18  file1
-rw-r--r-- 1 kali kali    0 Aug 12 10:18  file2
-rw-rw-r-- 1 kali kali    0 Aug 12 10:18  file3
-rw-r--r-- 1 kali kali    0 Aug  6 12:38  file_name
-rw-r--r-- 1 kali kali    0 Aug  5 12:34  filename
prw-r--r-- 1 kali kali    0 Aug  9 14:09  hacker
drwxr-xr-x 2 kali kali 4096 Aug 10 05:56  hello1
drwxrwxr-x 2 kali kali 4096 Aug 16 10:49 '*[[:lower:]]*'
drwxrwxr-x 2 kali kali 4096 Aug 16 10:46  ls
drwxr-xr-x 2 kali kali 4096 Aug  6 14:06  new_folder
drwxr-xr-x 2 kali kali 4096 Aug  6 11:11  nithesh
drwxr-xr-x 3 kali kali 4096 Aug  6 14:06  parent_foldder
prw-r--r-- 1 kali kali    0 Aug  9 14:08  pipefile
prw-r--r-- 1 kali kali    0 Aug  9 14:08  pipefile.gz
drwxr-xr-x 2 kali kali 4096 Aug 10 06:00  testdir
-rw-r--r-- 1 kali kali    0 Aug  6 12:54  time
drwxrwxr-x 2 kali kali 4096 Aug 16 10:46 '*[[:upper:]]*'
```

f) Change the permission set of /work/readme.txt so that only the user (owner) can read/write/ execute, group, and any user can execute it. However, whenever anyone executes it, it should run with the privileges of the group. Use absolute mode.

```
┌──(kali㉿kali)-[/home/kali/nithesh]
└─PS> chmod 2751 /home/kali/nithesh

┌──(kali㉿kali)-[/home/kali/nithesh]
└─PS> ls -l
total 60
drwxrwxr-x 2 kali kali 4096 Aug 16 10:43 '*[[:alnum:]]*'
drwxrwxr-x 2 kali kali 4096 Aug 16 10:42  alnum
drwxrwxr-x 2 kali kali 4096 Aug 16 10:49 '*[[:alpha:]]*'
drwxrwxr-x 2 kali kali 4096 Aug 16 10:49 '*[[:alpha:]]*,ls'
drwxr-xr-x 4 kali kali 4096 Aug  6 13:22  CLI
drwxr-xr-x 8 kali kali 4096 Aug 12 10:24  cys
drwxrwxr-x 2 kali kali 4096 Aug 16 10:49 '*[[:digit:]]*,ls'
-rw-r--r-- 1 kali kali    0 Aug  6 12:30  example.txt
-rw-rw-r-- 1 kali kali    0 Aug 12 10:18  file1
-rw-r--r-- 1 kali kali    0 Aug 12 10:18  file2
-rw-rw-r-- 1 kali kali    0 Aug 12 10:18  file3
-rw-r--r-- 1 kali kali    0 Aug  6 12:38  file_name
-rw-r--r-- 1 kali kali    0 Aug  5 12:34  filename
prw-r--r-- 1 kali kali    0 Aug  9 14:09  hacker
drwxr-xr-x 2 kali kali 4096 Aug 10 05:56  hello1
drwxrwxr-x 2 kali kali 4096 Aug 16 10:49 '*[[:lower:]]*'
drwxrwxr-x 2 kali kali 4096 Aug 16 10:46  ls
drwxr-xr-x 2 kali kali 4096 Aug  6 14:06  new_folder
drwxr-xr-x 2 kali kali 4096 Aug  6 11:11  nithesh
drwxr-xr-x 3 kali kali 4096 Aug  6 14:06  parent_foldder
prw-r--r-- 1 kali kali    0 Aug  9 14:08  pipefile
prw-r--r-- 1 kali kali    0 Aug  9 14:08  pipefile.gz
drwxr-xr-x 2 kali kali 4096 Aug 10 06:00  testdir
-rw-r--r-- 1 kali kali    0 Aug  6 12:54  time
drwxrwxr-x 2 kali kali 4096 Aug 16 10:46 '*[[:upper:]]*'
```

g) Change the permission set of /work/readme.txt so that only the owner can rename or delete this file while maintaining the existing permissions. Use absolute mode.

```
┌──(kali㉿kali)-[/home/kali/nithesh]
└─PS> chmod +t /home/kali/nithesh

┌──(kali㉿kali)-[/home/kali/nithesh]
└─PS> ls -l
total 60
drwxrwxr-x 2 kali kali 4096 Aug 16 10:43 '*[[:alnum:]]*'
drwxrwxr-x 2 kali kali 4096 Aug 16 10:42  alnum
drwxrwxr-x 2 kali kali 4096 Aug 16 10:49 '*[[:alpha:]]*'
drwxrwxr-x 2 kali kali 4096 Aug 16 10:49 '*[[:alpha:]]*,ls'
drwxr-xr-x 4 kali kali 4096 Aug  6 13:22  CLI
drwxr-xr-x 8 kali kali 4096 Aug 12 10:24  cys
drwxrwxr-x 2 kali kali 4096 Aug 16 10:49 '*[[:digit:]]*,ls'
-rw-r--r-- 1 kali kali    0 Aug  6 12:30  example.txt
-rw-rw-r-- 1 kali kali    0 Aug 12 10:18  file1
-rw-r--r-- 1 kali kali    0 Aug 12 10:18  file2
-rw-rw-r-- 1 kali kali    0 Aug 12 10:18  file3
-rw-r--r-- 1 kali kali    0 Aug  6 12:38  file_name
-rw-r--r-- 1 kali kali    0 Aug  5 12:34  filename
prw-r--r-- 1 kali kali    0 Aug  9 14:09  hacker
drwxr-xr-x 2 kali kali 4096 Aug 10 05:56  hello1
drwxrwxr-x 2 kali kali 4096 Aug 16 10:49 '*[[:lower:]]*'
drwxrwxr-x 2 kali kali 4096 Aug 16 10:46  ls
drwxr-xr-x 2 kali kali 4096 Aug  6 14:06  new_folder
drwxr-xr-x 2 kali kali 4096 Aug  6 11:11  nithesh
drwxr-xr-x 3 kali kali 4096 Aug  6 14:06  parent_foldder
prw-r--r-- 1 kali kali    0 Aug  9 14:08  pipefile
prw-r--r-- 1 kali kali    0 Aug  9 14:08  pipefile.gz
drwxr-xr-x 2 kali kali 4096 Aug 10 06:00  testdir
-rw-r--r-- 1 kali kali    0 Aug  6 12:54  time
drwxrwxr-x 2 kali kali 4096 Aug 16 10:46 '*[[:upper:]]*'
```

h) What are the default permissions for the new file?

The default permissions for a new file are typically `rw-r--r--` (644)

i) What was the command to view the file permissions?

`ls -l`

j) Change chmod.exercises permissions to -r--r--r—

chmod 444 chmod.exercises

k) Change the file permissions to Read only for the owner, group and all other users.

chmod 444 <filename>

l) What was the command for changing the file permissions to -r--r--r--?

chmod 444 <filename>

m) Change chmod.exercises permissions to -rw-r-----

chmod 640 chmod.exercises

n) Change the file permissions to match the following:

   a. owner: Read and Write
   b. group: Read
   c. other: no permissions (None)
      chmod 640 <filename>

o) What was the command for changing the file permissions to -rw-r-----?
chmod 640 <filename>

p) Change chmod.exercises permissions to -rwxr-x—x
chmod 751 chmod.exercises

q) Change the file permissions to match the following:
   a. owner: Read, Write and Execute
   b. group: Read and Execute
   c. other: Execute
      chmod 751 <filename>

r) What was the command for changing the file permissions to -rwxr-x--x?

chmod 751 <filename>

Evaluation :

Marks : 10 (Deadline : 4 – Originality :3 – Completeness :3)

Deadline: 06.08.2024

In life there are no shortcuts. All things are connected. For success there is no fast lane. Work hard. Focus your energy, practice, remain honest, Truthful, loyal and committed.

-unknown