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
3	/boot	is a critical part of the system, as it contains the files needed to boot the operating system.
4	/dev	directory in Linux is a special and essential directory that contains device files.
5	/etc	directory in Linux is a crucial directory that contains system-wide configuration files and shell scripts used to manage the system.
6	/home	directory in Linux is where the personal directories of all user accounts are stored.
7	/lib	directory in Linux is a critical system directory that contains essential shared libraries and kernel modules required by the system and applications.

8	/proc	directory in Linux is a special and dynamic virtual filesystem that provides a mechanism for the kernel to communicate with user space.
9	/sbin	directory is integral to system administration and maintenance, providing the tools necessary for managing and repairing the system, particularly during boot and recovery operations.
10	/tmp	directory in Linux is a temporary storage space where programs and users can store transient files that are needed only temporarily.
11	/var	directory in Linux is a key directory that holds variable data—data that is expected to change frequently during the system's operation.

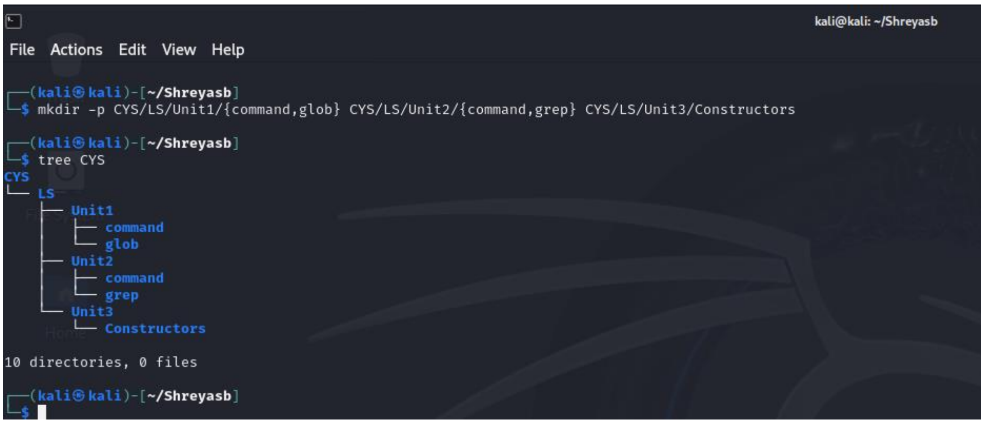
2. In Linux, there are three different files
   Regular file
   Directory
   Special file
      Block file
      Character file
      Socket file
      Pipe file

Fill the below table:

File Type	Represented by (Hint ls)	Role	How to create	Location	Screen shot
Regular file	-	Stores data such as text images	touch t1		
- Text file	-	Contains plain text	toucht1.txt		
- Compressed file	-	Stores data in a compressed format	gzip filename		
- Image	-	Stores image data	NA		
Directory	d	Contains files and other directories	mkdir filename		
Block file	b	Represents a block device	NA		
Character file	c	Represents a character device	NA		
Socket file	s	Provides interprocess communication(IPC)	NA		
pipe file	p	Used for IPC (inter – process communication)	NA		

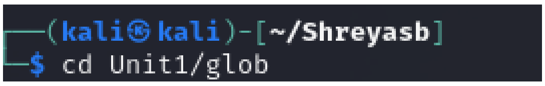
3. Globbing
a. Go back to CYS
b. Create multiple subdirectories using single command

```
LS
  Unit1
        command
         glob
  Unit2
        command
        grep
   Unit3
         Constructs
```



```
┌──(kali㉿kali)-[~/Shreyasb]
└─$ mkdir -p CYS/LS/Unit1/{command,glob} CYS/LS/Unit2/{command,grep} CYS/LS/Unit3/Constructors

┌──(kali㉿kali)-[~/Shreyasb]
└─$ tree CYS
CYS
└── LS
    ├── Unit1
    │   ├── command
    │   └── glob
    ├── Unit2
    │   ├── command
    │   └── grep
    └── Unit3
        └── Constructors

10 directories, 0 files

┌──(kali㉿kali)-[~/Shreyasb]
└─$ 
```

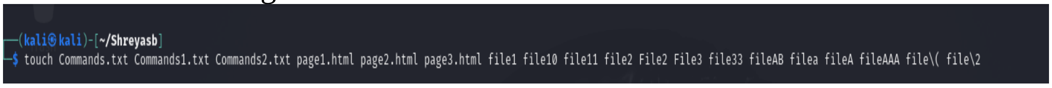
c. Navigate to unit1/glob



```
┌──(kali㉿kali)-[~/Shreyasb]
└─$ cd Unit1/glob
```

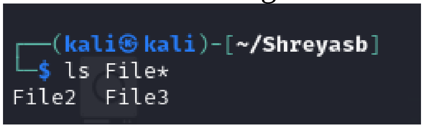
d. Create the following files :
Commands.txt
Commands1.txt
Commands2.txt
page1.html
page2.html
page3.html
file1
file10
file11
file2
File2
File3
file33
fileAB
filea
fileA
fileAAA
file(
file 2

i. List all files starting with file



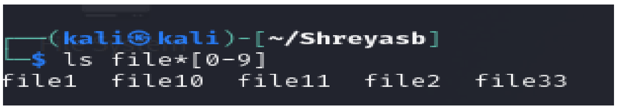
```
┌──(kali㉿kali)-[~/Shreyasb]
└─$ touch Commands.txt Commands1.txt Commands2.txt page1.html page2.html page3.html file1 file10 file11 file2 File2 File3 file33 fileAB filea fileA fileAAA file\( file\2
```

ii. List all files starting with File



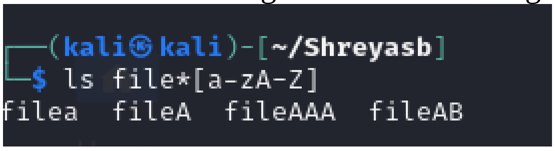
```
┌──(kali㉿kali)-[~/Shreyasb]
└─$ ls File*
File2  File3
```

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



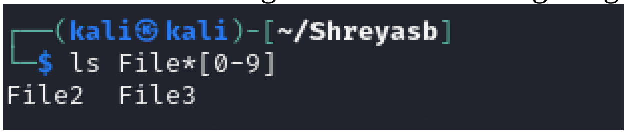
```
┌──(kali㉿kali)-[~/Shreyasb]
└─$ ls file*[0-9]
file1  file10  file11  file2  file33
```

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



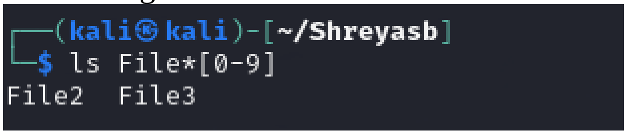
```
┌──(kali㉿kali)-[~/Shreyasb]
└─$ ls file*[a-zA-Z]
filea  fileA  fileAAA  fileAB
```

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



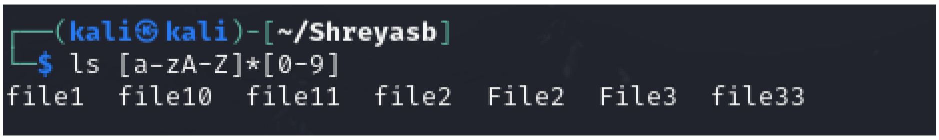
```
┌──(kali㉿kali)-[~/Shreyasb]
└─$ ls File*[0-9]
File2  File3
```

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



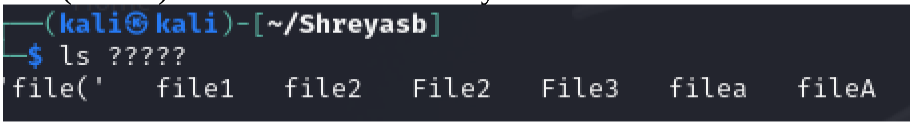
```
┌──(kali㉿kali)-[~/Shreyasb]
└─$ ls File*[0-9]
File2  File3
```

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



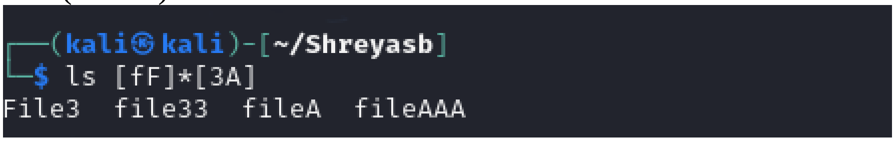
```
┌──(kali㉿kali)-[~/Shreyasb]
└─$ ls [a-zA-Z]*[0-9]
file1  file10  file11  file2  File2  File3  file33
```

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



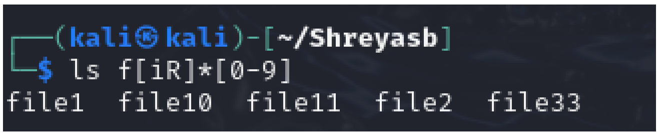
```
┌──(kali㉿kali)-[~/Shreyasb]
└─$ ls ?????
'file('   file1   file2   File2   File3   filea   fileA
```

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



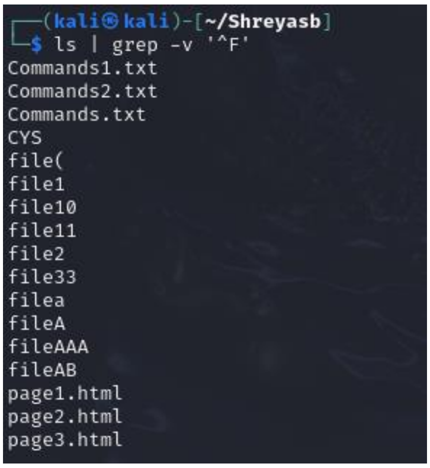
```
┌──(kali㉿kali)-[~/Shreyasb]
└─$ ls [fF]*[3A]
File3  file33  fileA  fileAAA
```

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



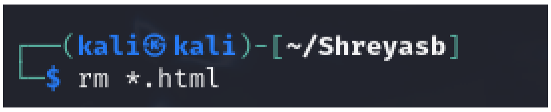
```
┌──(kali㉿kali)-[~/Shreyasb]
└─$ ls f[iR]*[0-9]
file1  file10  file11  file2  file33
```

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



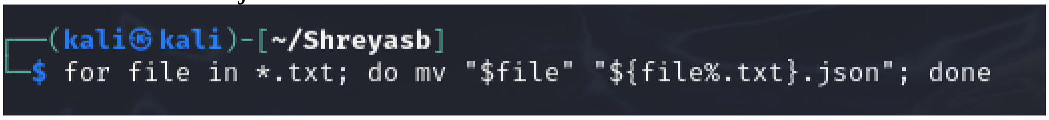
```
┌──(kali㉿kali)-[~/Shreyasb]
└─$ ls | grep -v '^F'
Commands1.txt
Commands2.txt
Commands.txt
CYS
file(
file1
file10
file11
file2
file33
filea
fileA
fileAAA
fileAB
page1.html
page2.html
page3.html
```

xii. Remove all the *.html



```
┌──(kali㉿kali)-[~/Shreyasb]
└─$ rm *.html
```

xiii. Rename *.txt to *.json



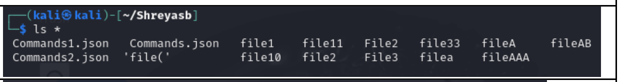
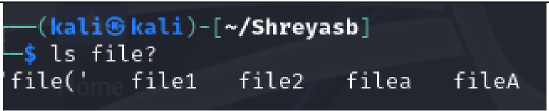
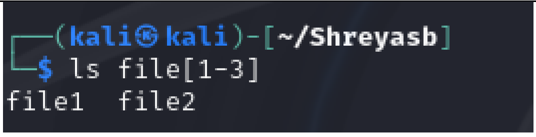
```
┌──(kali㉿kali)-[~/Shreyasb]
└─$ for file in *.txt; do mv "$file" "${file%.txt}.json"; done
```

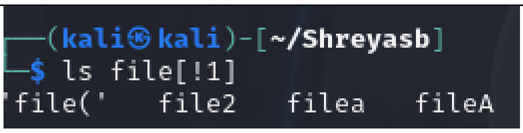
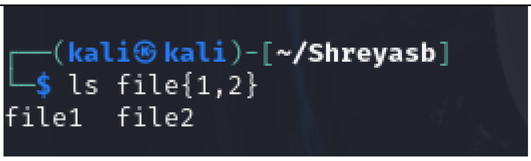
4. Absolute path and relative path

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

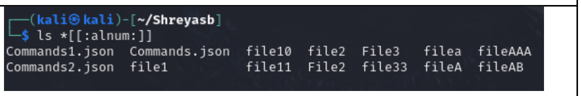
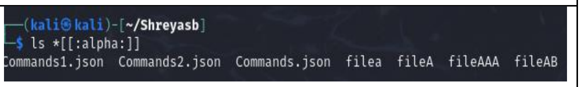
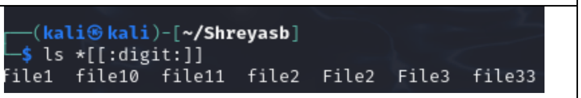
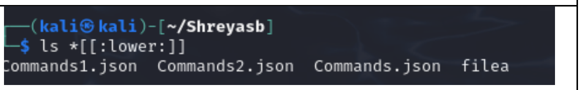
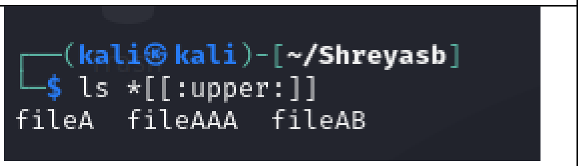
- Absolute **Path:** Specifies the location of a file or directory from the root directory. It is a complete path.
- Relative **Path:** Specifies the location of a file or directory relative to the current working directory.

5. Wildcards

Notation	Use	Example	Screenshot
*	One or many	ls *	 ┌──(kali㉿kali)-[~/Shreyasb] └─$ ls * Commands1.json Commands.json file1 file11 File2 file33 fileA fileAB Commands2.json 'file(' file10 file2 File3 filea fileAAA
?	Match only one character	ls file?	 ┌──(kali㉿kali)-[~/Shreyasb] └─$ ls file? 'file(' file1 file2 filea fileA
[ ]	Used to match single character from a set of specified characters	ls file[1-3]	 ┌──(kali㉿kali)-[~/Shreyasb] └─$ ls file[1-3] file1 file2

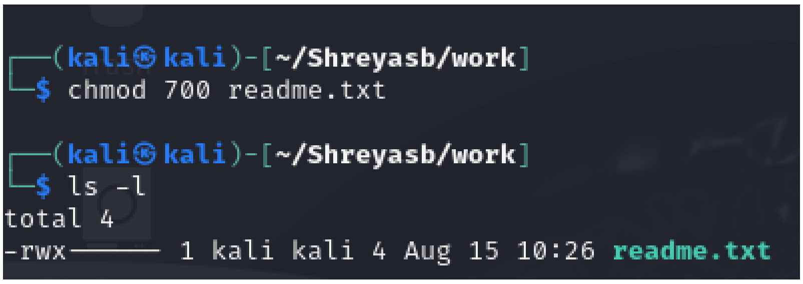
[!]	Matches any character that is not a member of the set characters	ls file[!1]	 ┌──(kali㉿kali)-[~/Shreyasb] └─$ ls file[!1] 'file(' file2 filea fileA
{ }	Used to generate multiple arguments by separating the values with commas	ls file{1,2}	 ┌──(kali㉿kali)-[~/Shreyasb] └─$ ls file{1,2} file1 file2

More on Character class

Notation	Use	Example	Screenshot
[:alnum:]	Matches any alphanumeric character	ls *[[:alnum:]]	 ┌──(kali㉿kali)-[~/Shreyasb] └─$ ls *[[:alnum:]] Commands1.json Commands.json file10 file2 File3 filea fileAAA Commands2.json file1 file11 File2 file33 fileA fileAB
[:alpha:]	Matches any alphabetic character	ls *[[:alpha:]]	 ┌──(kali㉿kali)-[~/Shreyasb] └─$ ls *[[:alpha:]] Commands1.json Commands2.json Commands.json filea fileA fileAAA fileAB
[:digit:]	Matches any numeric digit (0-9).	ls *[[:digit:]]	 ┌──(kali㉿kali)-[~/Shreyasb] └─$ ls *[[:digit:]] file1 file10 file11 file2 File2 File3 file33
[:lower:]	Matches any lowercase alphabetic character	ls *[[:lower:]]	 ┌──(kali㉿kali)-[~/Shreyasb] └─$ ls *[[:lower:]] Commands1.json Commands2.json Commands.json filea
[:upper:]	Matches any uppercase alphabetic character (A-Z)	ls *[[:upper:]]	 ┌──(kali㉿kali)-[~/Shreyasb] └─$ ls *[[:upper:]] fileA fileAAA fileAB

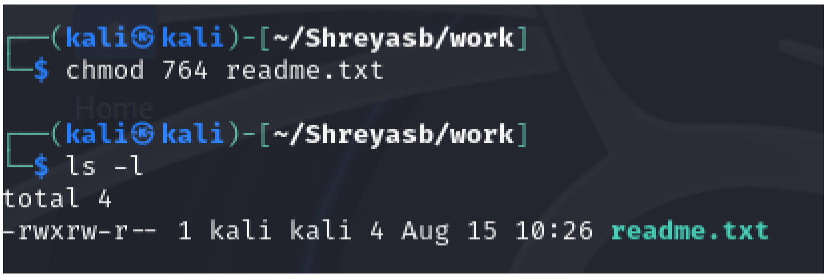
4. change permission

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



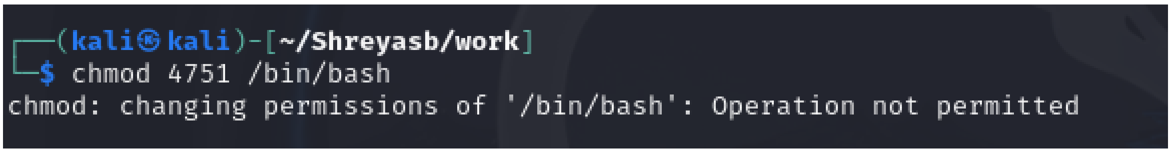
```
┌──(kali㉿kali)-[~/Shreyasb/work]
└─$ chmod 700 readme.txt

┌──(kali㉿kali)-[~/Shreyasb/work]
└─$ ls -l
total 4
-rwx------ 1 kali kali 4 Aug 15 10:26 readme.txt
```

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.



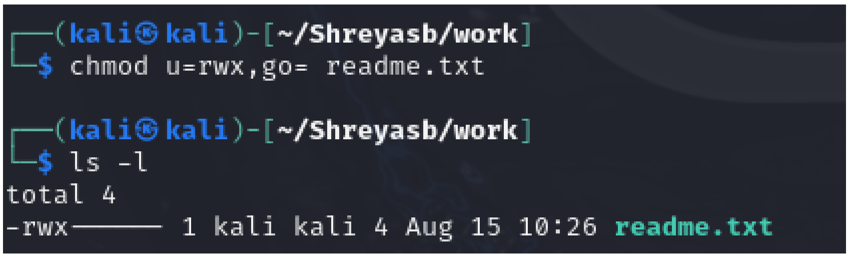
```
┌──(kali㉿kali)-[~/Shreyasb/work]
└─$ chmod 764 readme.txt

┌──(kali㉿kali)-[~/Shreyasb/work]
└─$ ls -l
total 4
-rwxrw-r-- 1 kali kali 4 Aug 15 10:26 readme.txt
```

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.



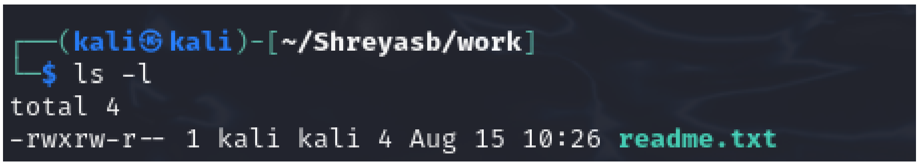
```
┌──(kali㉿kali)-[~/Shreyasb/work]
└─$ chmod 4751 /bin/bash
chmod: changing permissions of '/bin/bash': Operation not permitted
```

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



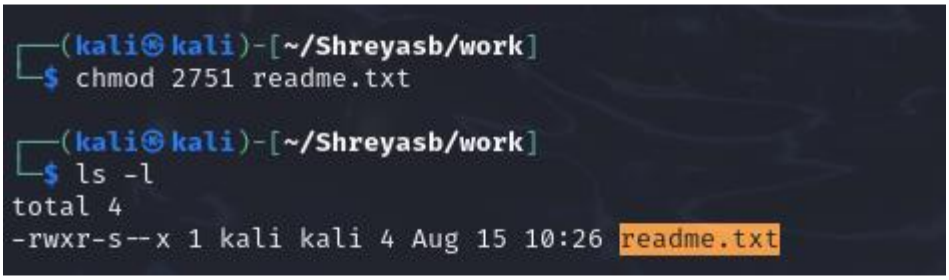
```
┌──(kali㉿kali)-[~/Shreyasb/work]
└─$ chmod u=rwx,go= readme.txt

┌──(kali㉿kali)-[~/Shreyasb/work]
└─$ ls -l
total 4
-rwx------ 1 kali kali 4 Aug 15 10:26 readme.txt
```

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.



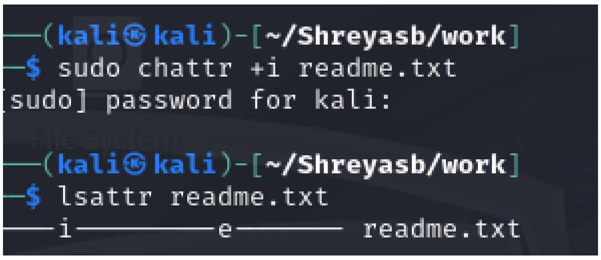
```
┌──(kali㉿kali)-[~/Shreyasb/work]
└─$ ls -l
total 4
-rwxrw-r-- 1 kali kali 4 Aug 15 10:26 readme.txt
```

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.



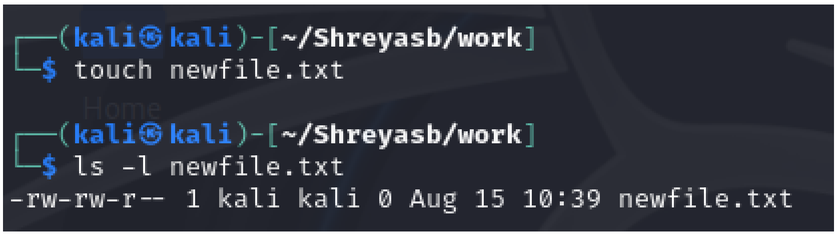
```
┌──(kali㉿kali)-[~/Shreyasb/work]
└─$ chmod 2751 readme.txt

┌──(kali㉿kali)-[~/Shreyasb/work]
└─$ ls -l
total 4
-rwxr-s--x 1 kali kali 4 Aug 15 10:26 readme.txt
```

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.



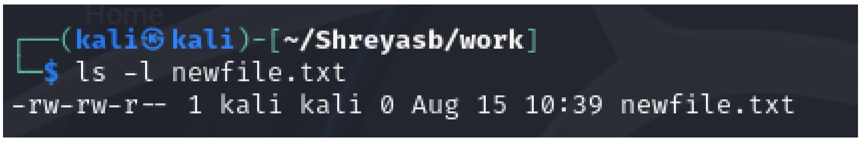
```
──(kali㉿kali)-[~/Shreyasb/work]
─$ sudo chattr +i readme.txt
sudo] password for kali:

──(kali㉿kali)-[~/Shreyasb/work]
─$ lsattr readme.txt
───i─────────e─────── readme.txt
```

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



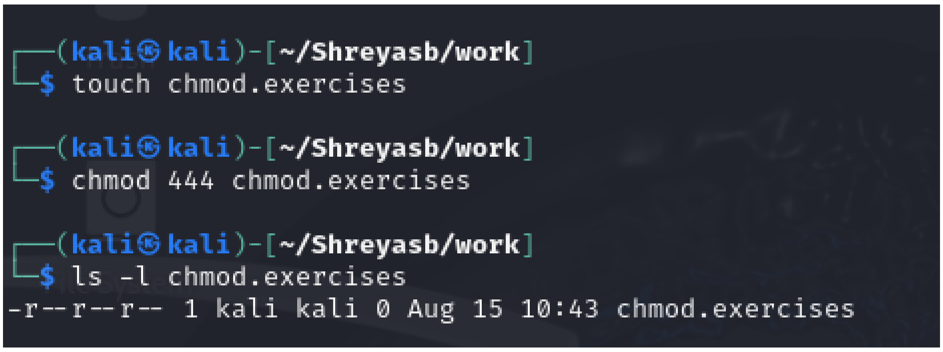
```
┌──(kali㉿kali)-[~/Shreyasb/work]
└─$ touch newfile.txt

┌──(kali㉿kali)-[~/Shreyasb/work]
└─$ ls -l newfile.txt
-rw-rw-r-- 1 kali kali 0 Aug 15 10:39 newfile.txt
```

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



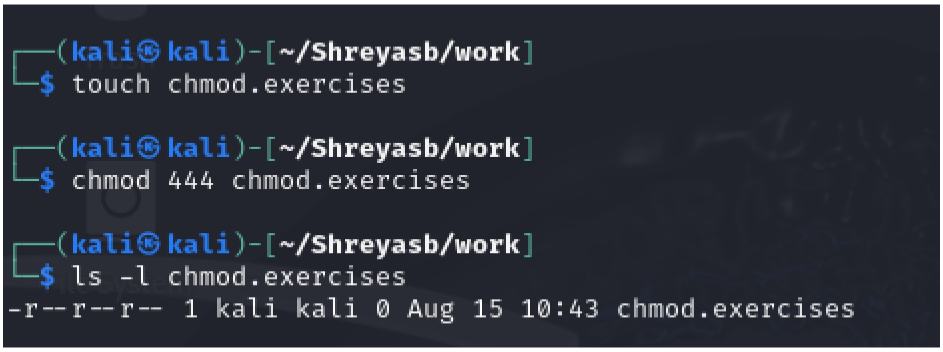
```
┌──(kali㉿kali)-[~/Shreyasb/work]
└─$ ls -l newfile.txt
-rw-rw-r-- 1 kali kali 0 Aug 15 10:39 newfile.txt
```

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



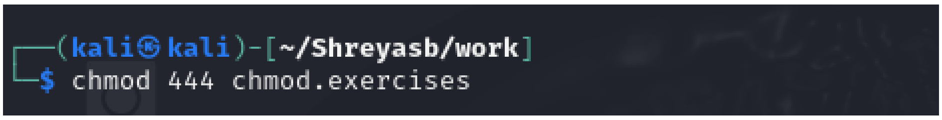
```
┌──(kali㉿kali)-[~/Shreyasb/work]
└─$ touch chmod.exercises

┌──(kali㉿kali)-[~/Shreyasb/work]
└─$ chmod 444 chmod.exercises

┌──(kali㉿kali)-[~/Shreyasb/work]
└─$ ls -l chmod.exercises
-r--r--r-- 1 kali kali 0 Aug 15 10:43 chmod.exercises
```

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



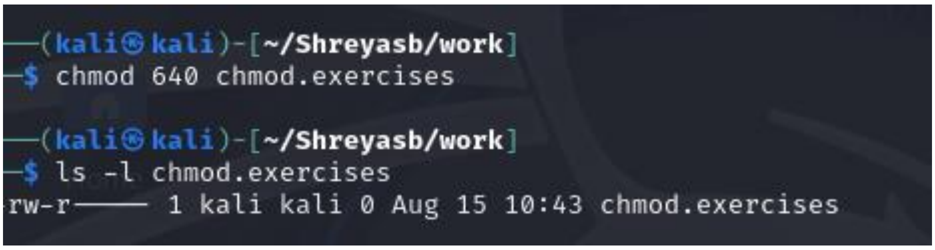
```
┌──(kali㉿kali)-[~/Shreyasb/work]
└─$ touch chmod.exercises

┌──(kali㉿kali)-[~/Shreyasb/work]
└─$ chmod 444 chmod.exercises

┌──(kali㉿kali)-[~/Shreyasb/work]
└─$ ls -l chmod.exercises
-r--r--r-- 1 kali kali 0 Aug 15 10:43 chmod.exercises
```

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



```
┌──(kali㉿kali)-[~/Shreyasb/work]
└─$ chmod 444 chmod.exercises
```

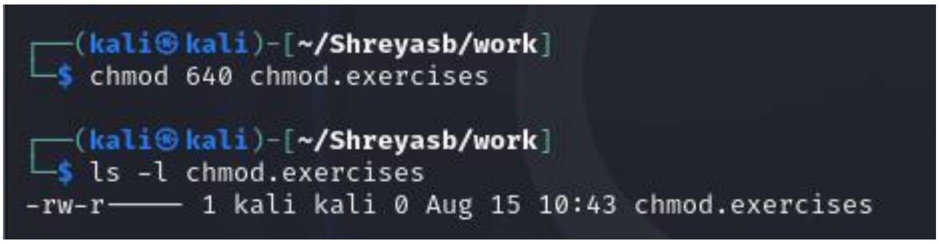
m) Change chmod.exercises permissions to -rw-r-----



```
──(kali㉿kali)-[~/Shreyasb/work]
─$ chmod 640 chmod.exercises

──(kali㉿kali)-[~/Shreyasb/work]
─$ ls -l chmod.exercises
rw-r----- 1 kali kali 0 Aug 15 10:43 chmod.exercises
```

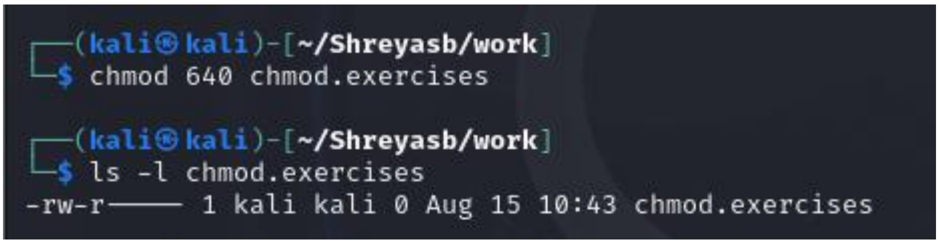
n) Change the file permissions to match the following:

a. owner: Read and Write
b. group: Read
c. other: no permissions (None)



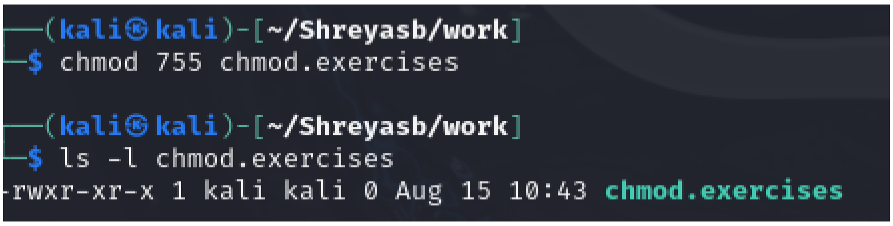
```
┌──(kali㉿kali)-[~/Shreyasb/work]
└─$ chmod 640 chmod.exercises

┌──(kali㉿kali)-[~/Shreyasb/work]
└─$ ls -l chmod.exercises
-rw-r----- 1 kali kali 0 Aug 15 10:43 chmod.exercises
```

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



```
┌──(kali㉿kali)-[~/Shreyasb/work]
└─$ chmod 640 chmod.exercises

┌──(kali㉿kali)-[~/Shreyasb/work]
└─$ ls -l chmod.exercises
-rw-r----- 1 kali kali 0 Aug 15 10:43 chmod.exercises
```

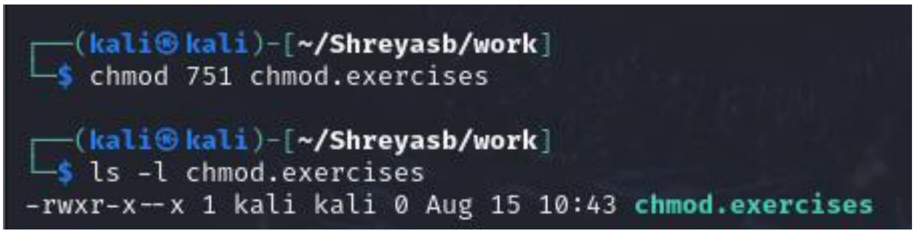
p) Change chmod.exercises permissions to -rwxr-x—x



```
──(kali㉿kali)-[~/Shreyasb/work]
─$ chmod 755 chmod.exercises

──(kali㉿kali)-[~/Shreyasb/work]
─$ ls -l chmod.exercises
-rwxr-xr-x 1 kali kali 0 Aug 15 10:43 chmod.exercises
```

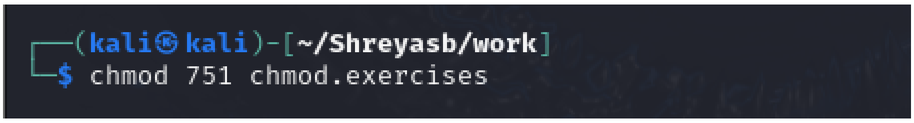
q) Change the file permissions to match the following:

a. owner: Read, Write and Execute
b. group: Read and Execute
c. other: Execute



```
┌──(kali㉿kali)-[~/Shreyasb/work]
└─$ chmod 751 chmod.exercises

┌──(kali㉿kali)-[~/Shreyasb/work]
└─$ ls -l chmod.exercises
-rwxr-x--x 1 kali kali 0 Aug 15 10:43 chmod.exercises
```

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



```
┌──(kali㉿kali)-[~/Shreyasb/work]
└─$ chmod 751 chmod.exercises
```

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