The main idea of this lab exercise to give hands on experiance on

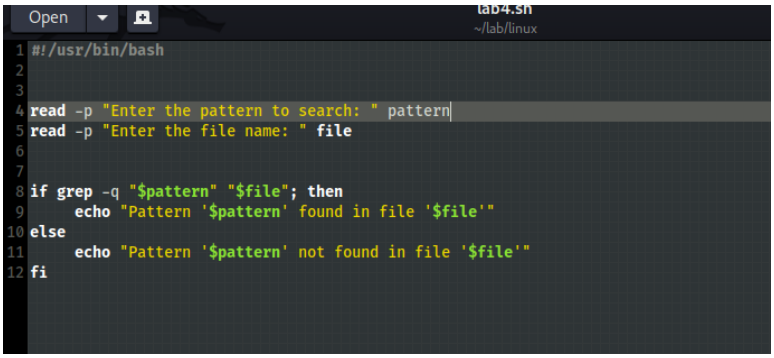
grep

constructs

command line arguments

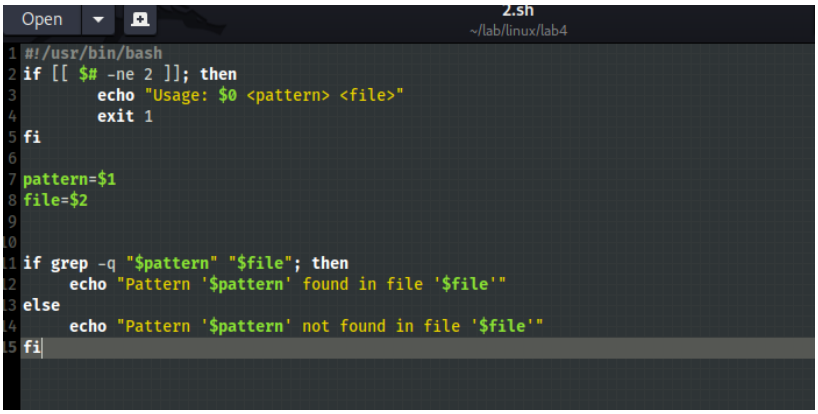
1. write a shell script to get value the pattern and file name from the user and check the pattern exists or not. If the pattern exists print the relavent

message , if pattern not found print relevent message.



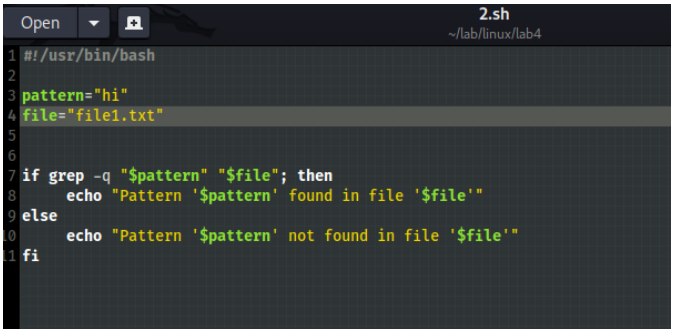

```
┌──(ceyona㉿kali)-[~/lab/linux]
└─$ bash lab4.sh
Enter the pattern to search: hi
Enter the file name: file1.txt
Pattern 'hi' found in file 'file1.txt'
```

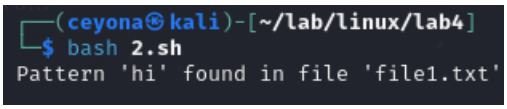
2. Modify the above script to pass the arguments from command line arguments.




```
┌──(ceyona㉿kali)-[~/lab/linux/lab4]
└─$ bash 2.sh hi file1.txt
Pattern 'hi' found in file 'file1.txt'
```

3. Modify the above script to pass the values inside the script.

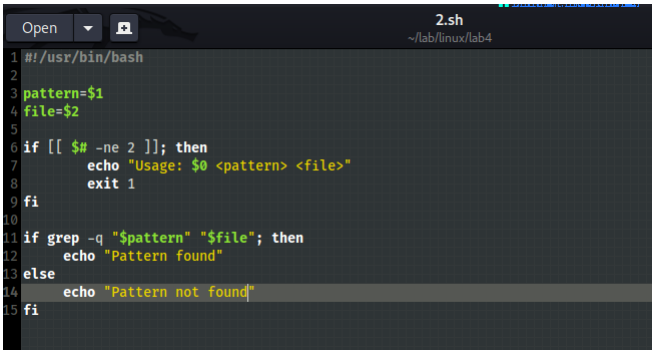


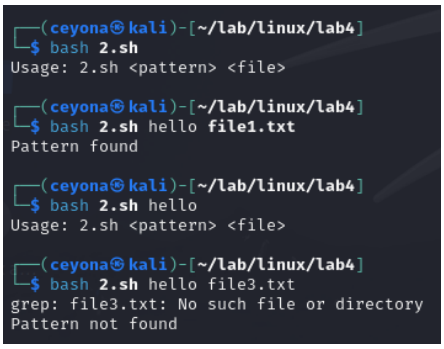


4. validate the script (script 1, script 2)

- the file exists or not

- arguments passed or not





5. Apply grep commands

Note: Make sure to u se the options -e -c -n -q -s -f -A -B -C -i -h, -l -o -w

Frame the questions (as per your choice)

to extract user information

to extract network information

to extract login details

1. To extract network information

```
┌──(ceyona㉿kali)-[~/lab/linux/lab4]
└─$ grep -A 2 -B 2 "text" file1.txt
hi
how are you
text
Are you happy
crypto lab
los lab
text
hi
how are you
text
are you happy
crypto lab
los lab
text
enter your password
```

2. To extract login details

```
┌──(ceyona㉿kali)-[~/lab/linux/lab4]
└─$ grep -c "login" file1.txt
3
```

3. Using Mutiple options

```
┌──(ceyona㉿kali)-[~/lab/linux/lab4]
└─$ grep -o -n -w "login" file1.txt
5:login
11:login
18:login
```

4. To search multiple pattern using –e

```
┌──(ceyona㉿kali)-[~/lab/linux/lab4]
└─$ grep -e "login" -e "hi"  file1.txt
hi
hi
login here
login to begin
hi
login
```