



**ADDIS ABABA UNIVERSITY**

**ADDIS ABABA INSTITUTE OF TECHNOLOGY**

School of Information and Technology  
Engineering

## **Operating system Individual Assignment**

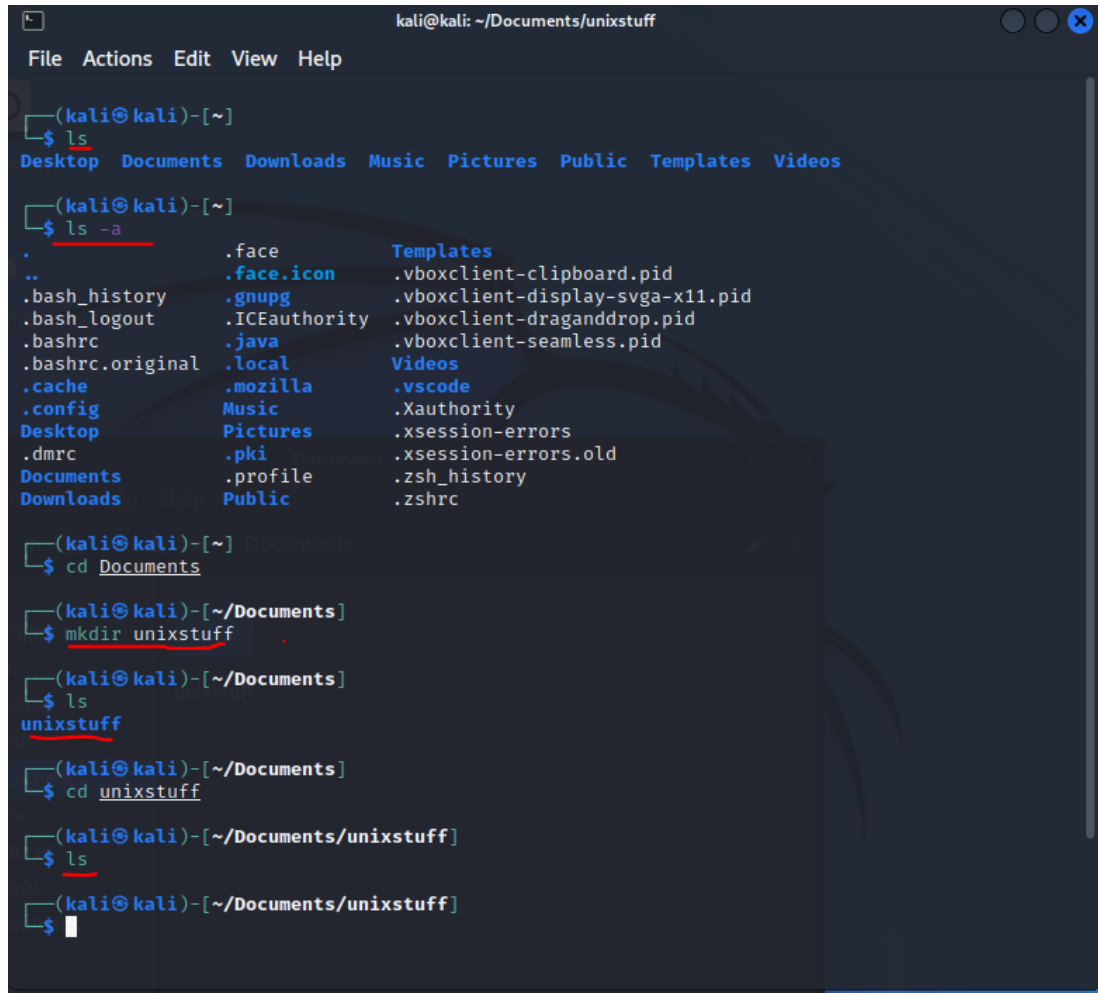
*Name : Biruk Anley*

*Id: UGR/3923/12*

*Section : 1*

# Practicing on UNIX command

- Down below I have tried to use the command “ls” to see files that I have .
- In addition tried to create a new folder “unixstuff” so that I can use it to create files with that directory



```
kali@kali: ~/Documents/unixstuff
File Actions Edit View Help

(kali@kali)-[~]
$ ls
Desktop Documents Downloads Music Pictures Public Templates Videos

(kali@kali)-[~]
$ ls -a
.          .face          Templates
..         .face.icon    .vboxclient-clipboard.pid
.bash_history .gnupg        .vboxclient-display-svga-x11.pid
.bash_logout .ICEauthority .vboxclient-draganddrop.pid
.bashrc      .java         .vboxclient-seamless.pid
.bashrc.original .local       Videos
.cache       .mozilla     .vscode
.config      Music        .Xauthority
Desktop      Pictures     .xsession-errors
.dmrc        .pki         .xsession-errors.old
Documents    .profile     .zsh_history
Downloads    Public       .zshrc

(kali@kali)-[~]
$ cd Documents

(kali@kali)-[~/Documents]
$ mkdir unixstuff

(kali@kali)-[~/Documents]
$ ls
unixstuff

(kali@kali)-[~/Documents]
$ cd unixstuff

(kali@kali)-[~/Documents/unixstuff]
$ ls

(kali@kali)-[~/Documents/unixstuff]
$
```

I used the command

- “ls: to list files
- “ls-a” to list files whose name begin with .(dot)
- “mkdir” to create folders or directory
- “cd” to change directory so that I can use it

Down below I have tried to use (.) and (..) for navigating or swithing to different directory

```
kali@kali: ~  
File Actions Edit View Help  
(kali@kali)-[~/Documents/unixstuff]  
$ cd .  
(kali@kali)-[~/Documents/unixstuff]  
$ cd ..  
(kali@kali)-[~/Documents]  
$ cd unixstuff  
(kali@kali)-[~/Documents/unixstuff]  
$ cd  
(kali@kali)-[~]  
$ cd Documents  
(kali@kali)-[~/Documents]  
$ cd unixstuff  
(kali@kali)-[~/Documents/unixstuff]  
$ pwd  
/home/kali/Documents/unixstuff  
(kali@kali)-[~/Documents/unixstuff]  
$ cd  
(kali@kali)-[~]  
$
```

I used the command

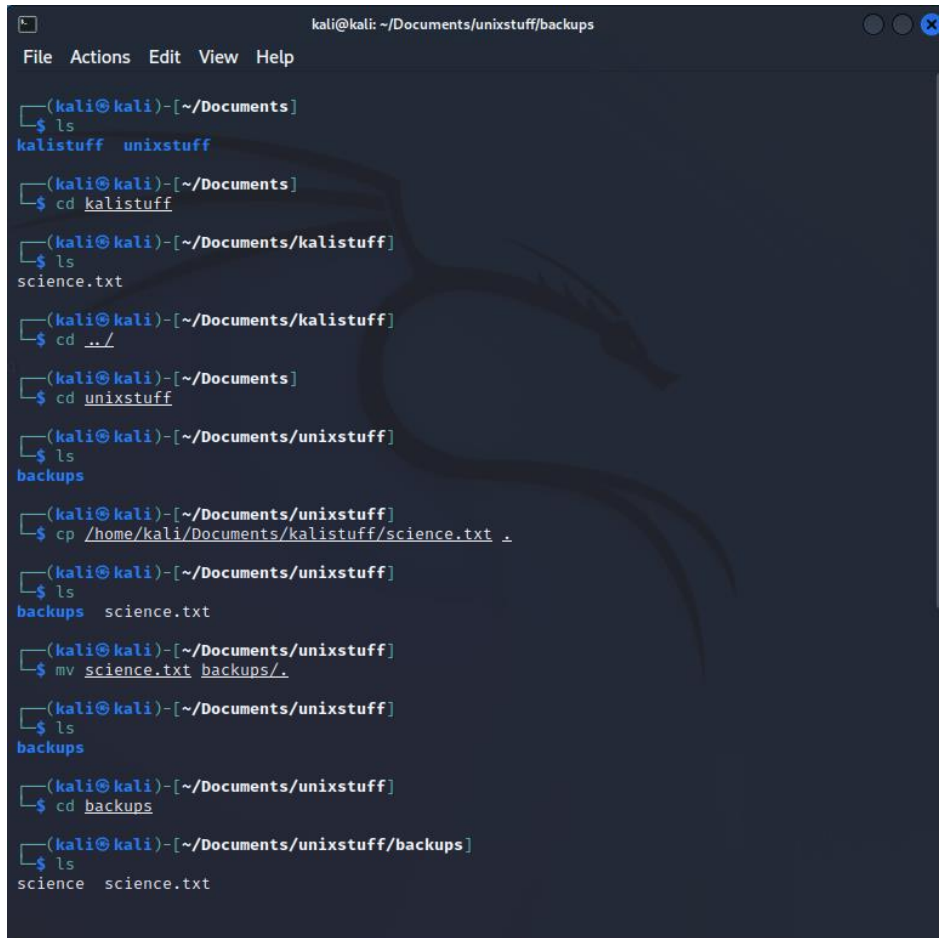
- "cd ." to see the current directory
- "cd .." to see the parent of the directory
- "pwd" to see the path of the current directory

```
kali@kali: ~/Documents/unixstuff/backups  
File Actions Edit View Help  
(kali@kali)-[~/Documents]  
$ cd unixstuff/backups  
(kali@kali)-[~/Documents/unixstuff/backups]  
$ ls  
Desktop Documents Downloads Music Pictures Public Templates Videos  
(kali@kali)-[~/Documents/unixstuff/backups]  
$ ls ../..  
kali  
(kali@kali)-[~/Documents/unixstuff/backups]  
$
```

In the above figure I use “ls ~” to denote a user's home directory. As you see “ls~” displays home directory files which exist with home directory. In addition the command “ls~/..” tells the the root directory for any data. It display kali since my home directory for all program in my linux is Kali

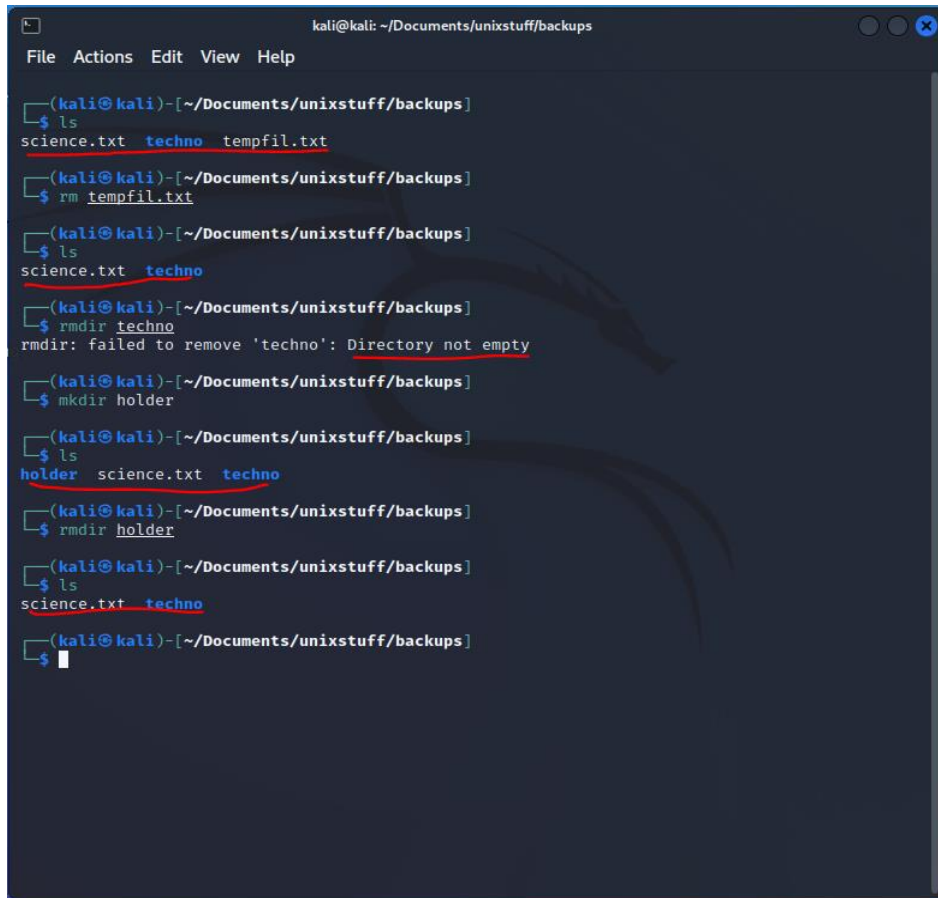
# UNIX Tutorial Two

## Copying and moving files among directory

A terminal window titled 'kali@kali: ~/Documents/unixstuff/backups' with a menu bar (File, Actions, Edit, View, Help). The terminal shows a series of commands and their outputs: 1. Prompt: (kali@kali)~[~/Documents]; Command: \$ ls; Output: kalistuff unixstuff. 2. Prompt: (kali@kali)~[~/Documents]; Command: \$ cd kalistuff; Output: (kali@kali)~[~/Documents/kalistuff]. 3. Prompt: (kali@kali)~[~/Documents/kalistuff]; Command: \$ ls; Output: science.txt. 4. Prompt: (kali@kali)~[~/Documents/kalistuff]; Command: \$ cd ../; Output: (kali@kali)~[~/Documents]. 5. Prompt: (kali@kali)~[~/Documents]; Command: \$ cd unixstuff; Output: (kali@kali)~[~/Documents/unixstuff]. 6. Prompt: (kali@kali)~[~/Documents/unixstuff]; Command: \$ ls; Output: backups. 7. Prompt: (kali@kali)~[~/Documents/unixstuff]; Command: \$ cp /home/kali/Documents/kalistuff/science.txt .; Output: (kali@kali)~[~/Documents/unixstuff]. 8. Prompt: (kali@kali)~[~/Documents/unixstuff]; Command: \$ ls; Output: backups science.txt. 9. Prompt: (kali@kali)~[~/Documents/unixstuff]; Command: \$ mv science.txt backups/.; Output: (kali@kali)~[~/Documents/unixstuff]. 10. Prompt: (kali@kali)~[~/Documents/unixstuff]; Command: \$ ls; Output: backups. 11. Prompt: (kali@kali)~[~/Documents/unixstuff]; Command: \$ cd backups; Output: (kali@kali)~[~/Documents/unixstuff/backups]. 12. Prompt: (kali@kali)~[~/Documents/unixstuff/backups]; Command: \$ ls; Output: science science.txt. The terminal has a dark background with a faint dragon watermark.

In the above picture I tried to copy file “science.txt “ from unixstuff directory to Kalistaff and I use the command “cp “ with specifying the location address. In addition I have tried to move file which exist with in kalistaff and I tried to move file to backups

## Removing files and directory in linux



```
kali@kali: ~/Documents/unixstuff/backups
File Actions Edit View Help

(kali@kali)~/Documents/unixstuff/backups
$ ls
science.txt techno tempfil.txt

(kali@kali)~/Documents/unixstuff/backups
$ rm tempfil.txt

(kali@kali)~/Documents/unixstuff/backups
$ ls
science.txt techno

(kali@kali)~/Documents/unixstuff/backups
$ rmdir techno
rmdir: failed to remove 'techno': Directory not empty

(kali@kali)~/Documents/unixstuff/backups
$ mkdir holder

(kali@kali)~/Documents/unixstuff/backups
$ ls
holder science.txt techno

(kali@kali)~/Documents/unixstuff/backups
$ rmdir holder

(kali@kali)~/Documents/unixstuff/backups
$ ls
science.txt techno

(kali@kali)~/Documents/unixstuff/backups
$
```

In the above practice I have tried to remove file called tempfil.txt and it was removed successfully with the command “rm” and I have tried to remove directory called holder with the command “rmdir”

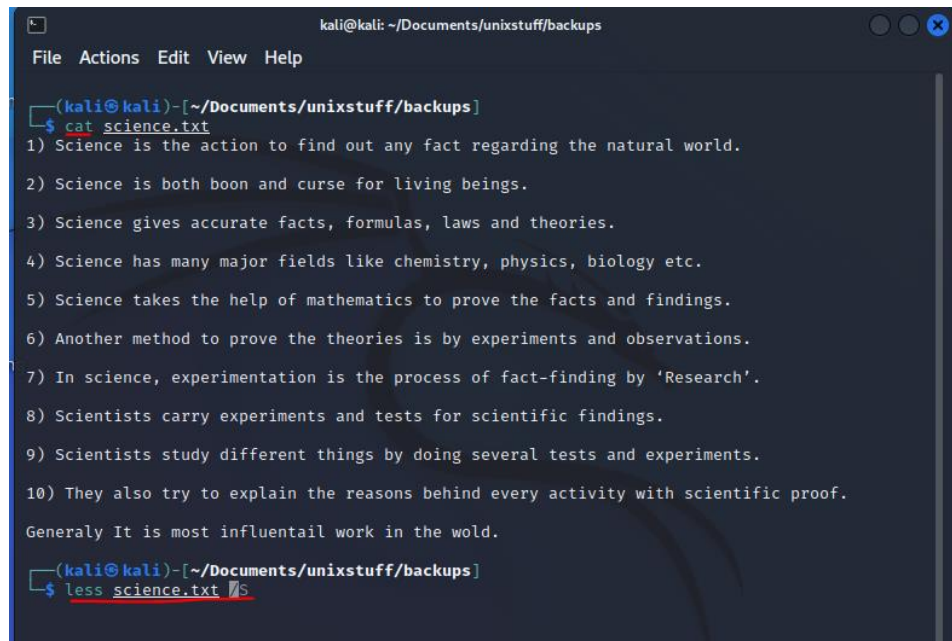
## Displaying the content of A file on the screen

There are a lot of methods to display the content of a file on the screen those methods are

- Cat
- Less
- Head and tail

Which are listed below

- I used to command “cat science.txt” to display to file on the screen. Cat, is not a dedicated file reader. The intended use of Cat is to take multiple inputs and stick them end to end. Cat is short for "concatenate", which means exactly that.

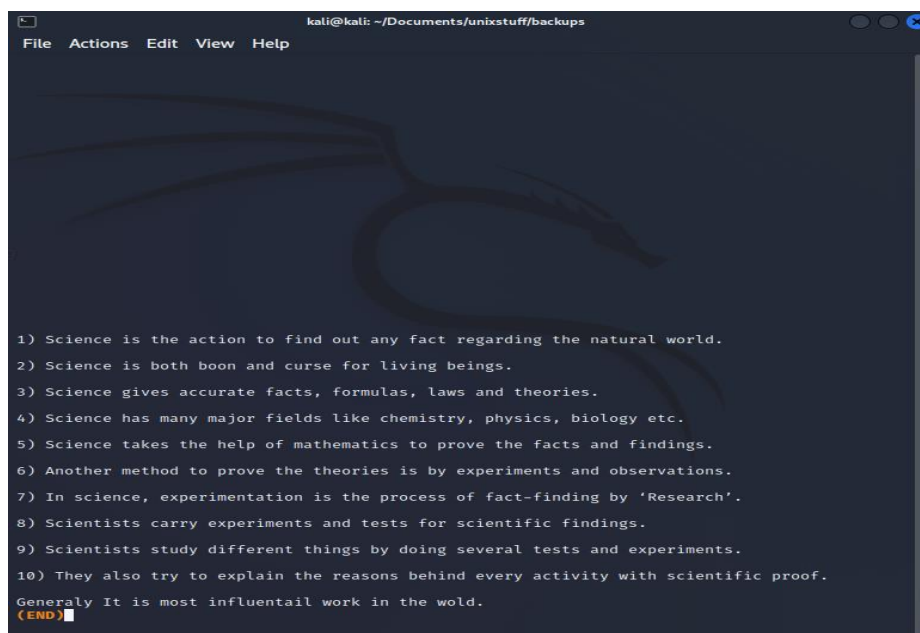


```
kali@kali: ~/Documents/unixstuff/backups
File Actions Edit View Help

(kali@kali)-[~/Documents/unixstuff/backups]
$ cat science.txt
1) Science is the action to find out any fact regarding the natural world.
2) Science is both boon and curse for living beings.
3) Science gives accurate facts, formulas, laws and theories.
4) Science has many major fields like chemistry, physics, biology etc.
5) Science takes the help of mathematics to prove the facts and findings.
6) Another method to prove the theories is by experiments and observations.
7) In science, experimentation is the process of fact-finding by 'Research'.
8) Scientists carry experiments and tests for scientific findings.
9) Scientists study different things by doing several tests and experiments.
10) They also try to explain the reasons behind every activity with scientific proof.
Generally It is most influentail work in the wold.

(kali@kali)-[~/Documents/unixstuff/backups]
$ less science.txt
```

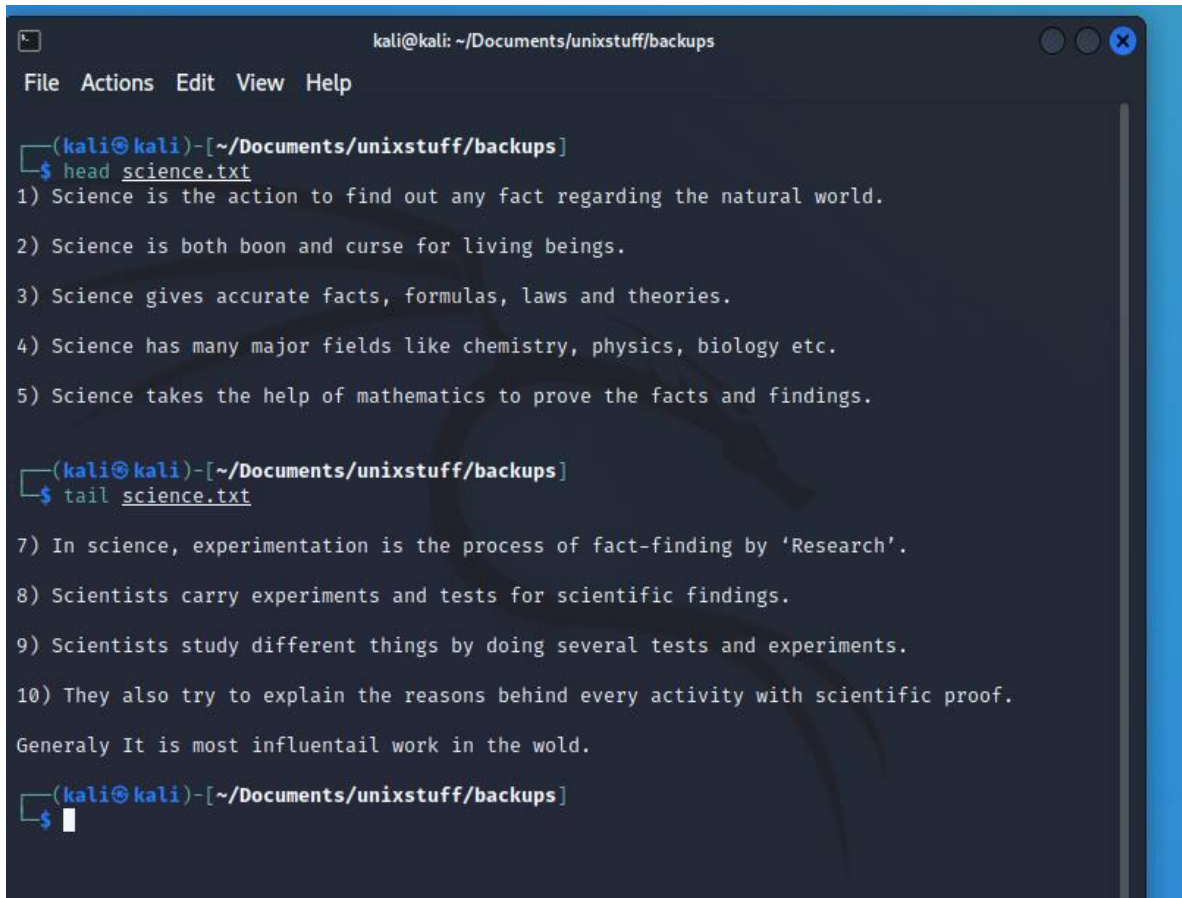
- I used the command “less.txt” to display files on the screen. Unlike that of cat, Less is a dedicated file reader that reads a file one screen at a time, and loads more of the file as you scroll through it.



```
kali@kali: ~/Documents/unixstuff/backups
File Actions Edit View Help

1) Science is the action to find out any fact regarding the natural world.
2) Science is both boon and curse for living beings.
3) Science gives accurate facts, formulas, laws and theories.
4) Science has many major fields like chemistry, physics, biology etc.
5) Science takes the help of mathematics to prove the facts and findings.
6) Another method to prove the theories is by experiments and observations.
7) In science, experimentation is the process of fact-finding by 'Research'.
8) Scientists carry experiments and tests for scientific findings.
9) Scientists study different things by doing several tests and experiments.
10) They also try to explain the reasons behind every activity with scientific proof.
Generally It is most influentail work in the wold.
(END)
```

- I have used the command head and tail to display the content of some portion .  
I have seen that the command “ head” will display the first ten line of a file.  
The command ” tail” has displayed the last ten lines of a file.



```
kali@kali: ~/Documents/unixstuff/backups
File Actions Edit View Help

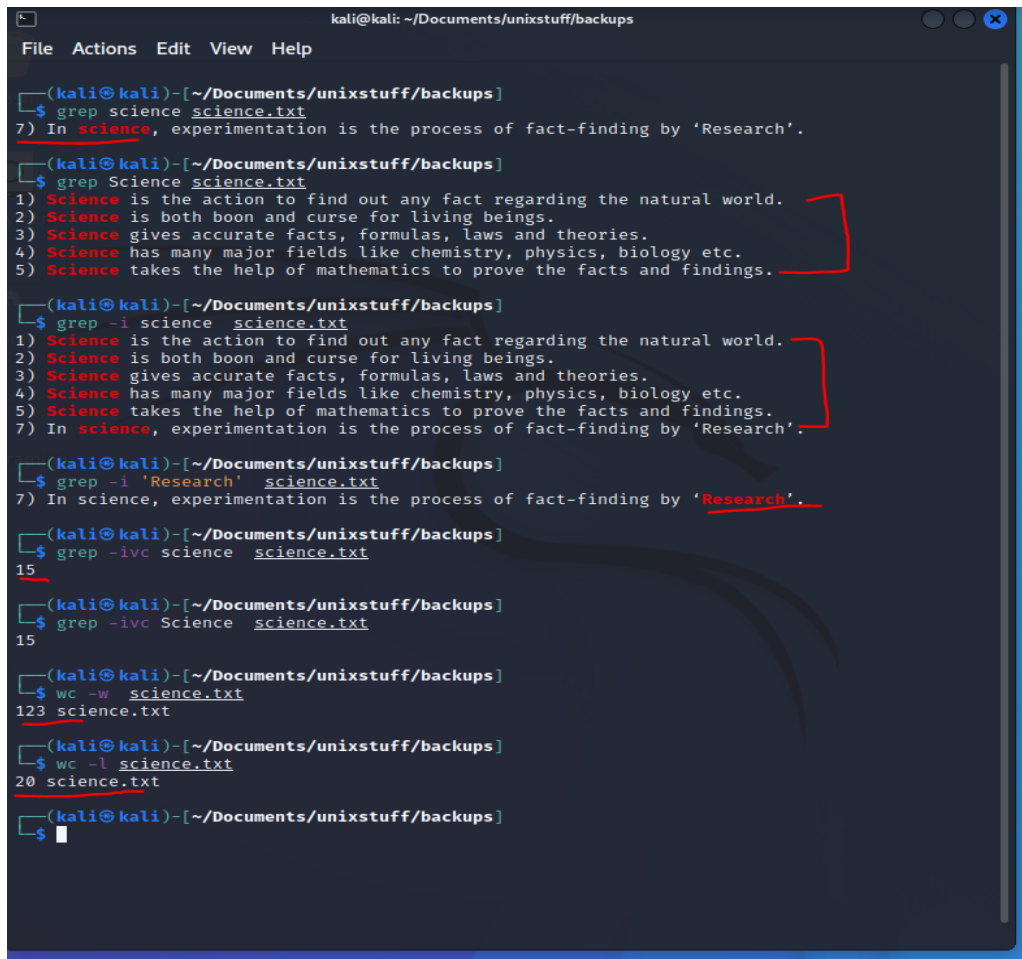
(kali@kali)~[~/Documents/unixstuff/backups]
$ head science.txt
1) Science is the action to find out any fact regarding the natural world.
2) Science is both boon and curse for living beings.
3) Science gives accurate facts, formulas, laws and theories.
4) Science has many major fields like chemistry, physics, biology etc.
5) Science takes the help of mathematics to prove the facts and findings.

(kali@kali)~[~/Documents/unixstuff/backups]
$ tail science.txt
7) In science, experimentation is the process of fact-finding by 'Research'.
8) Scientists carry experiments and tests for scientific findings.
9) Scientists study different things by doing several tests and experiments.
10) They also try to explain the reasons behind every activity with scientific proof.
Generally It is most influential work in the world.

(kali@kali)~[~/Documents/unixstuff/backups]
$
```



## Searching the content of files in linux



```
kali@kali: ~/Documents/unixstuff/backups
File Actions Edit View Help

(kali@kali)-[~/Documents/unixstuff/backups]
$ grep science science.txt
7) In science, experimentation is the process of fact-finding by 'Research'.

(kali@kali)-[~/Documents/unixstuff/backups]
$ grep Science science.txt
1) Science is the action to find out any fact regarding the natural world.
2) Science is both boon and curse for living beings.
3) Science gives accurate facts, formulas, laws and theories.
4) Science has many major fields like chemistry, physics, biology etc.
5) Science takes the help of mathematics to prove the facts and findings.

(kali@kali)-[~/Documents/unixstuff/backups]
$ grep -i science science.txt
1) Science is the action to find out any fact regarding the natural world.
2) Science is both boon and curse for living beings.
3) Science gives accurate facts, formulas, laws and theories.
4) Science has many major fields like chemistry, physics, biology etc.
5) Science takes the help of mathematics to prove the facts and findings.
7) In science, experimentation is the process of fact-finding by 'Research'.

(kali@kali)-[~/Documents/unixstuff/backups]
$ grep -i 'Research' science.txt
7) In science, experimentation is the process of fact-finding by 'Research'.

(kali@kali)-[~/Documents/unixstuff/backups]
$ grep -ivc science science.txt
15

(kali@kali)-[~/Documents/unixstuff/backups]
$ grep -ivc Science science.txt
15

(kali@kali)-[~/Documents/unixstuff/backups]
$ wc -w science.txt
123 science.txt

(kali@kali)-[~/Documents/unixstuff/backups]
$ wc -l science.txt
20 science.txt

(kali@kali)-[~/Documents/unixstuff/backups]
$
```

I have used the command

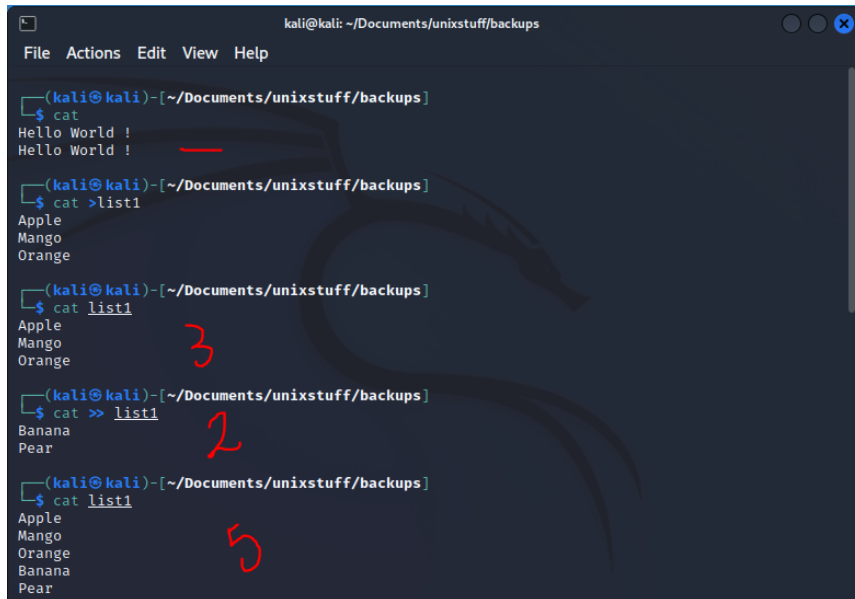
- “grep science” to find the occurrence of the word science and I have got 7
- “grep -i science science.txt” to ignore lowercase and uppercase
- “grep -i ‘research’ science.txt” to find quoted word inside science.txt
- “grep -ivc science” to find the number of lines that science is not there and I have got 15
- “wc -w science.txt” to count a number of words and I have got 123 words
- “wc -l science.txt” to find number of lines and I have got 20 lines



# UNIX Tutorial Three

## Redirection

### 1.Redirecting the output



```
kali@kali: ~/Documents/unixstuff/backups
File Actions Edit View Help

(kali@kali)~[~/Documents/unixstuff/backups]
$ cat
Hello World !
Hello World !

(kali@kali)~[~/Documents/unixstuff/backups]
$ cat >list1
Apple
Mango
Orange

(kali@kali)~[~/Documents/unixstuff/backups]
$ cat list1
Apple
Mango
Orange

(kali@kali)~[~/Documents/unixstuff/backups]
$ cat >> list1
Banana
Pear

(kali@kali)~[~/Documents/unixstuff/backups]
$ cat list1
Apple
Mango
Orange
Banana
Pear
```

The terminal window shows the process of creating and appending to a file. Red annotations are present: a red line under the first 'Hello World !', a red '3' next to the first 'cat list1' output, a red '2' next to the 'cat >> list1' command, and a red '5' next to the final 'cat list1' output.

I used the command cat to redirect both the input and the output of commands. At the first I have created a list name "list1" using the command : cat > list1 .Then I use the command " cat >> list1" to append some elements to previously one. And as we see the effect when I call out list out the original list was increased with 2 elements.

### 1.B Append another file

in this practice I have created new list named as list2 and I use command "cat list1 list2 > biglist" which is the command that is used for joining the two elements and as I saw I had 3 elems but once I merget it it form a big list which contain 8 elemnts

```
(kali㉿kali)-[~/Documents/unixstuff/backups]
$ cat > list2
watermelon
Pineapple
Strweberies

(kali㉿kali)-[~/Documents/unixstuff/backups]
$ cat list2
watermelon
Pineapple
Strweberies

(kali㉿kali)-[~/Documents/unixstuff/backups]
$ cat list1 list2 > biglist

(kali㉿kali)-[~/Documents/unixstuff/backups]
$ cat biglist
Apple
Mango
Orange
Banana
Pear
watermelon
Pineapple
Strweberies

(kali㉿kali)-[~/Documents/unixstuff/backups]
$
```

## 2.Redirecting the input

```
kali@kali: ~/Documents/unixstuff/backups
File Actions Edit View Help

(kali㉿kali)-[~/Documents/unixstuff/backups]
$ sort
dog
Cat

Cat
dog

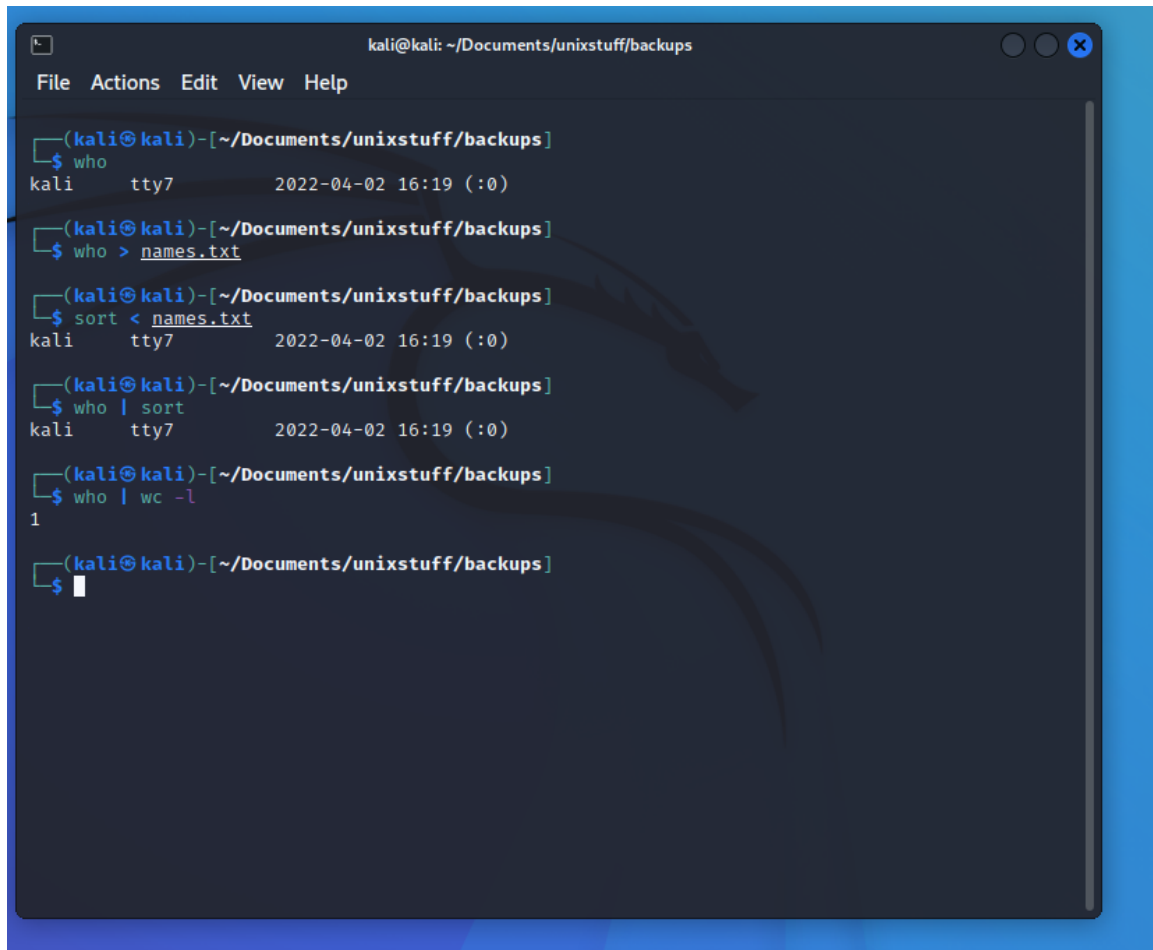
(kali㉿kali)-[~/Documents/unixstuff/backups]
$ sort <biglist
Apple
Banana
Mango
Orange
Pear
Pineapple
Strweberies
watermelon

(kali㉿kali)-[~/Documents/unixstuff/backups]
$ sort <biglist> science.txt

(kali㉿kali)-[~/Documents/unixstuff/backups]
$
```

In this practice I used the command “sort” to sort elemnets that are listed above so I have got sorted list and then I tried to sort out the previous list ( new list which is merged with list 1) and I had been succesfully sort out

# Pipes

A terminal window titled 'kali@kali: ~/Documents/unixstuff/backups' with a menu bar (File, Actions, Edit, View, Help). The terminal shows a series of commands and their outputs. The first command is 'who', which outputs 'kali tty7 2022-04-02 16:19 (:0)'. The second command is 'who > names.txt', which outputs nothing. The third command is 'sort < names.txt', which outputs the same 'who' output. The fourth command is 'who | sort', which also outputs the same 'who' output. The fifth command is 'who | wc -l', which outputs '1'. The sixth command is '\$' followed by a cursor, indicating the prompt is ready for input.

```
kali@kali: ~/Documents/unixstuff/backups
File Actions Edit View Help

(kali@kali)~/Documents/unixstuff/backups
$ who
kali    tty7          2022-04-02 16:19 (:0)

(kali@kali)~/Documents/unixstuff/backups
$ who > names.txt

(kali@kali)~/Documents/unixstuff/backups
$ sort < names.txt
kali    tty7          2022-04-02 16:19 (:0)

(kali@kali)~/Documents/unixstuff/backups
$ who | sort
kali    tty7          2022-04-02 16:19 (:0)

(kali@kali)~/Documents/unixstuff/backups
$ who | wc -l
1

(kali@kali)~/Documents/unixstuff/backups
$
```

- I use the command “who ” to display who is on the system with us and add some functionality to see those users sortedly.
- I didn’t have any other user than me so the only answer provided is Kali.
- I used the commnad “who | wc-l” to see how many users are available

# UNIX Tutorial Four

## The \* wildcard

```
(kali㉿kali)-[/home/kali/Documents/unixstuff]
PS> dir
Directory: /home/kali/Documents/unixstuff

Mode                LastWriteTime         Length Name
----                -
d-----         4/2/2022   7:43 PM             backjoker
d-----         4/2/2022   7:40 PM             backups
d-----         4/2/2022   7:45 PM              bob
d-----         4/2/2022   7:45 PM             bobyy
d-----         4/2/2022   7:43 PM             filer
d-----         4/2/2022   7:44 PM              job

(kali㉿kali)-[/home/kali/Documents/unixstuff]
PS> ls bo*
bob:
bobyy:

(kali㉿kali)-[/home/kali/Documents/unixstuff]
PS> ls *er
backjoker:
filer:

(kali㉿kali)-[/home/kali/Documents/unixstuff]
PS>
```

In this practice I have understood the \* function on finding match files

At the first I have 7 files. I used the command "ls bo\*" to see files which matches files that begin with the word "bo" and I have got 2 files "bob" and "bobyy"

On the other hand, I used the command "ls \*er" to see files which matches files that ends with the word "er" and I have got 2 files named as "backjoker" and "filer"

## Getting help

```
kali@kali: ~/Documents/unixstuff
File Actions Edit View Help

(kali㉿kali)-[/~/Documents/unixstuff]
$ man wc
```

```
kali@kali: ~/Documents/unixstuff
File Actions Edit View Help
WC(1) User Commands WC(1)

NAME
wc - print newline, word, and byte counts for each file

SYNOPSIS
wc [OPTION]... [FILE]...
wc [OPTION]... --files0-from=F

DESCRIPTION
Print newline, word, and byte counts for each FILE, and a total line if more than
one FILE is specified. A word is a non-zero-length sequence of characters delimited
by white space.

With no FILE, or when FILE is -, read standard input.

The options below may be used to select which counts are printed, always in the fol-
lowing order: newline, word, character, byte, maximum line length.

-c, --bytes
    print the byte counts

-m, --chars
    print the character counts

-l, --lines
    print the newline counts

--files0-from=F
    read input from the files specified by NUL-terminated names in file F; If F
    is - then read names from standard input

-L, --max-line-length
    print the maximum display width

-w, --words
    print the word counts

--help
    display this help and exit

--version
    output version information and exit

AUTHOR
Written by Paul Rubin and David MacKenzie.

REPORTING BUGS
GNU coreutils online help: <https://www.gnu.org/software/coreutils/>
Report any translation bugs to <https://translationproject.org/team/>
```

I used the command “man wc” to find out commands that is related to unix.

```
kali@kali: ~/Documents/unixstuff
File Actions Edit View Help

(kali@kali)-[~/Documents/unixstuff]
$ whatis wc
wc (1) - print newline, word, and byte counts for each file

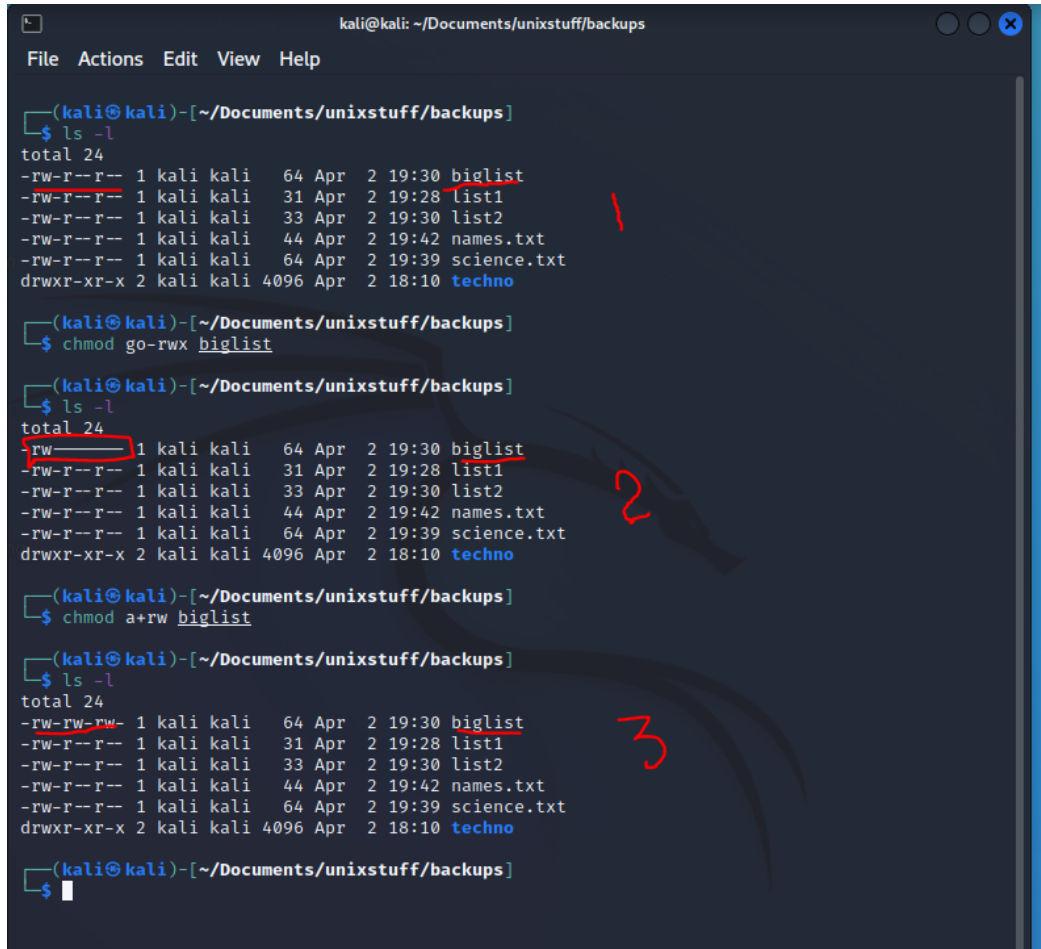
(kali@kali)-[~/Documents/unixstuff]
$ apropos copy
cifsdd (8) - convert and copy a file over SMB
COPY (7) - copy data between a file and a table
cp (1) - copy files and directories
cpgr (8) - copy with locking the given file to the password or group file
cpio (1) - copy files to and from archives
cppw (8) - copy with locking the given file to the password or group file
dd (1) - convert and copy a file
debconf-copydb (1) - copy a debconf database
git-checkout-index (1) - Copy files from the index to the working tree
gvfs-copy (1) - Deprecated equivalent of gio copy
install (1) - copy files and set attributes
llvm-objcopy-13 (1) - object copying and editing tool
mariadb-hotcopy (1) - a database backup program
mysqlhotcopy (1) - a database backup program
ntfscp (8) - copy file to an NTFS volume.
objcopy (1) - copy and translate object files
ptrepack (1) - Copy any PyTables Leaf, Group or complete subtree into another file.
rcp (1) - OpenSSH secure file copy
rsync (1) - a fast, versatile, remote (and local) file-copying tool
scp (1) - OpenSSH secure file copy
ssh-copy-id (1) - use locally available keys to authorise logins on a remote machine
svnversion (1) - Produce a compact version identifier for a working copy.
vfs_shadow_copy (8) - Expose snapshots to Windows clients as shadow copies.
vfs_shadow_copy2 (8) - Expose snapshots to Windows clients as shadow copies.
x86_64-linux-gnu-objcopy (1) - copy and translate object files

(kali@kali)-[~/Documents/unixstuff]
$
```

I used the command “aprops copy” to get description about the exact command which is reltaed to copying files from one directory to another directory.

# UNIX Tutorial Five

## File security in Unix



The screenshot shows a terminal window with the following commands and output:

```
kali@kali: ~/Documents/unixstuff/backups
File Actions Edit View Help

(kali@kali)~/Documents/unixstuff/backups
$ ls -l
total 24
-rw-r--r-- 1 kali kali 64 Apr 2 19:30 biglist
-rw-r--r-- 1 kali kali 31 Apr 2 19:28 list1
-rw-r--r-- 1 kali kali 33 Apr 2 19:30 list2
-rw-r--r-- 1 kali kali 44 Apr 2 19:42 names.txt
-rw-r--r-- 1 kali kali 64 Apr 2 19:39 science.txt
drwxr-xr-x 2 kali kali 4096 Apr 2 18:10 techno

(kali@kali)~/Documents/unixstuff/backups
$ chmod go-rwx biglist

(kali@kali)~/Documents/unixstuff/backups
$ ls -l
total 24
-rw-r--r-- 1 kali kali 64 Apr 2 19:30 biglist
-rw-r--r-- 1 kali kali 31 Apr 2 19:28 list1
-rw-r--r-- 1 kali kali 33 Apr 2 19:30 list2
-rw-r--r-- 1 kali kali 44 Apr 2 19:42 names.txt
-rw-r--r-- 1 kali kali 64 Apr 2 19:39 science.txt
drwxr-xr-x 2 kali kali 4096 Apr 2 18:10 techno

(kali@kali)~/Documents/unixstuff/backups
$ chmod a+rw biglist

(kali@kali)~/Documents/unixstuff/backups
$ ls -l
total 24
-rw-rw-rw- 1 kali kali 64 Apr 2 19:30 biglist
-rw-r--r-- 1 kali kali 31 Apr 2 19:28 list1
-rw-r--r-- 1 kali kali 33 Apr 2 19:30 list2
-rw-r--r-- 1 kali kali 44 Apr 2 19:42 names.txt
-rw-r--r-- 1 kali kali 64 Apr 2 19:39 science.txt
drwxr-xr-x 2 kali kali 4096 Apr 2 18:10 techno

(kali@kali)~/Documents/unixstuff/backups
$
```

Handwritten red annotations are present: a '1' next to the first 'ls -l' output, a '2' next to the second 'ls -l' output, and a '3' next to the third 'ls -l' output.

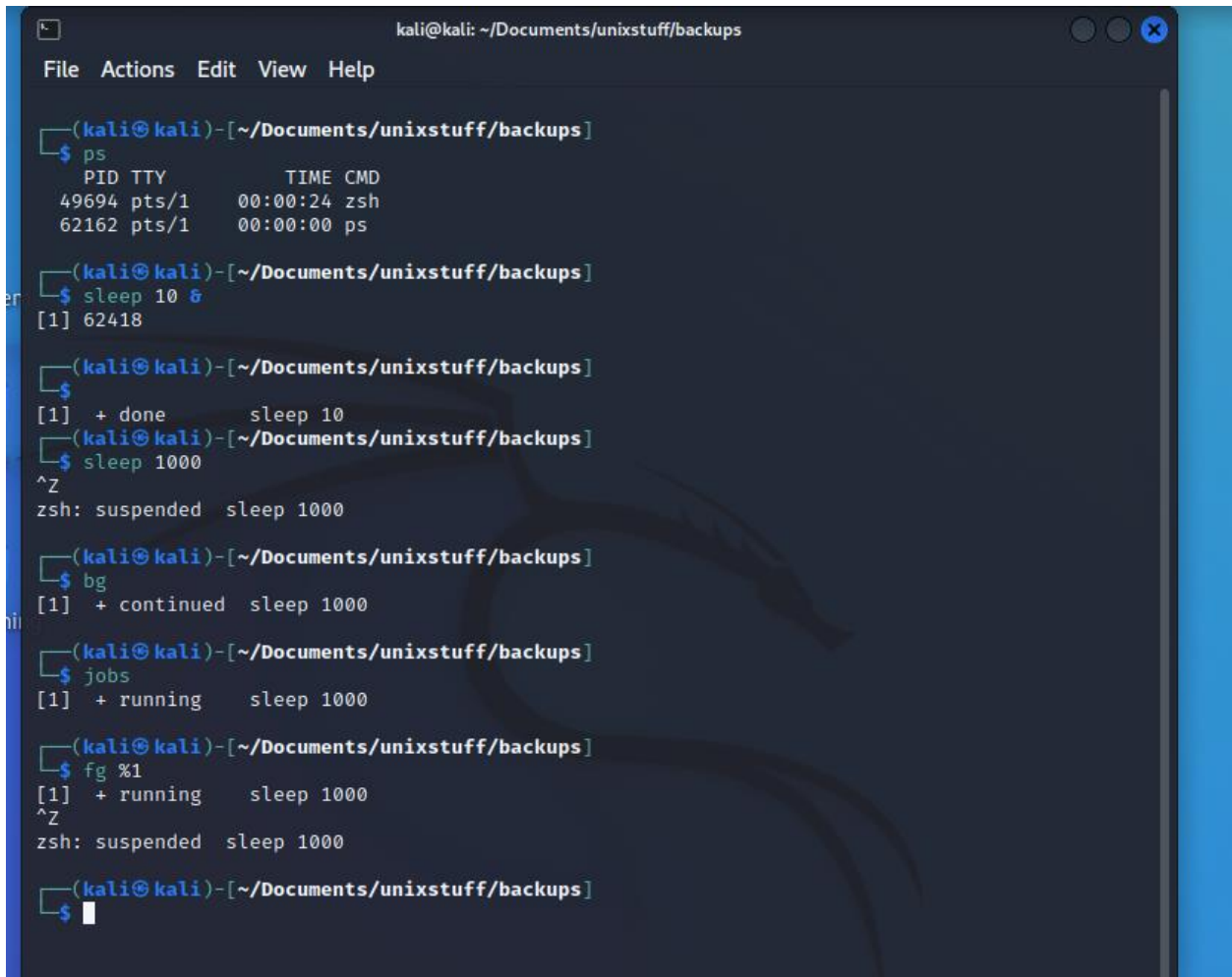
In this practice I have given access rights for different files

I used the command:

- “ls -l” to know about more the associated access rights and big list has the permission to read only
- “chmod go-rwx biglist” to remove read and write operation permission on the file “biglist”  
And I have watched that the Permission to read in A big list operation is now removed.
- “chmod a+rw biglist” to give the read and write operation for files that exist on biglist and I saw  
That A user can have the permission to read and write so I learnt how to change the permission of files



## Process and Jobs



```
kali@kali: ~/Documents/unixstuff/backups
File Actions Edit View Help

(kali@kali)-[~/Documents/unixstuff/backups]
$ ps
  PID TTY          TIME CMD
 49694 pts/1        00:00:24 zsh
 62162 pts/1        00:00:00 ps

(kali@kali)-[~/Documents/unixstuff/backups]
$ sleep 10 &
[1] 62418

(kali@kali)-[~/Documents/unixstuff/backups]
$
[1] + done          sleep 10

(kali@kali)-[~/Documents/unixstuff/backups]
$ sleep 1000
^Z
zsh: suspended    sleep 1000

(kali@kali)-[~/Documents/unixstuff/backups]
$ bg
[1] + continued    sleep 1000

(kali@kali)-[~/Documents/unixstuff/backups]
$ jobs
[1] + running      sleep 1000

(kali@kali)-[~/Documents/unixstuff/backups]
$ fg %1
[1] + running      sleep 1000
^Z
zsh: suspended    sleep 1000

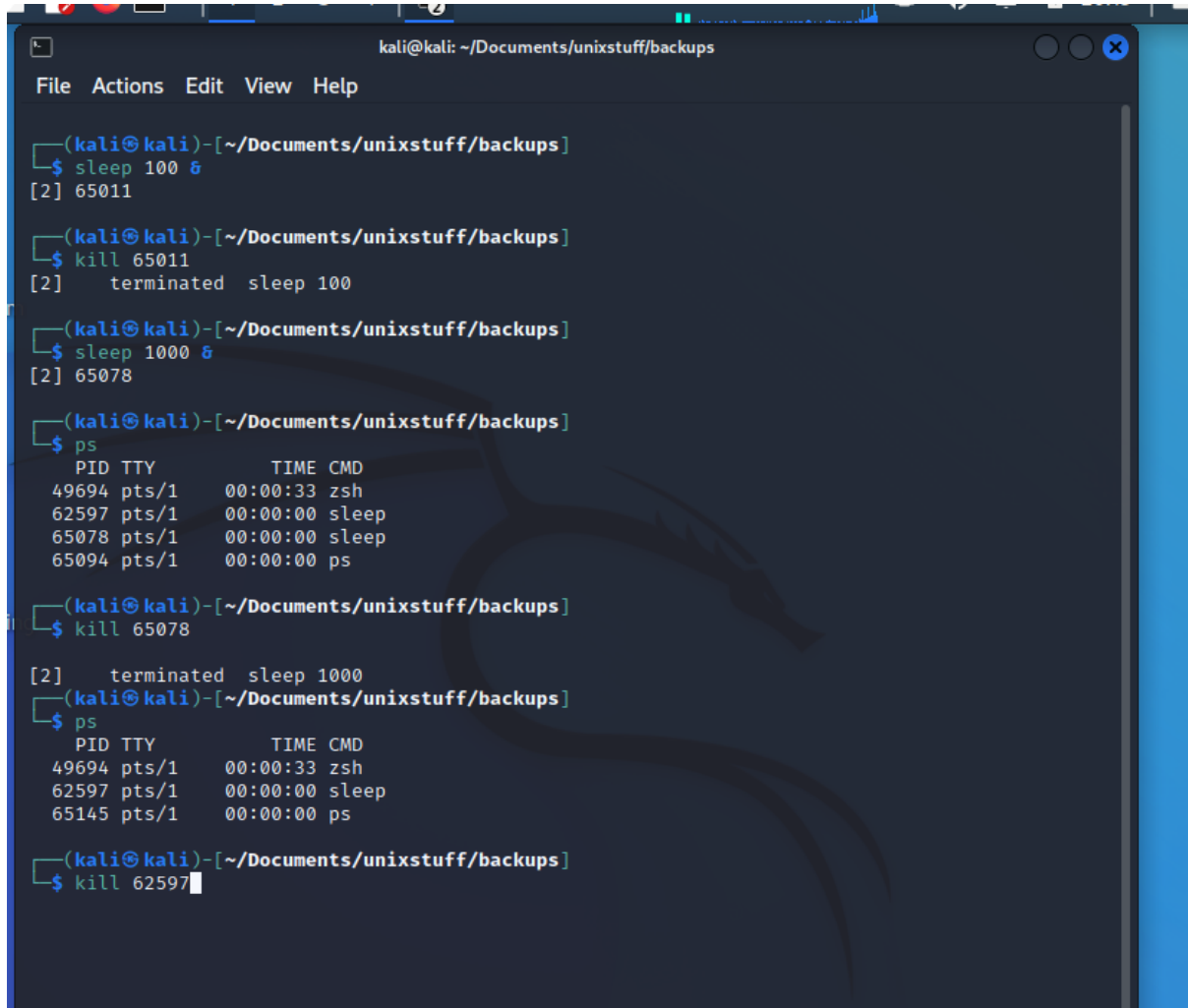
(kali@kali)-[~/Documents/unixstuff/backups]
$
```

In the above command I have been practiced on process and jobs

I used the command :

- “ps” to see information about the current executable process. As a result I have seen 2 process currently executed
- “sleep 10&” to create a process tha run the background it implies that the sleep process will wait 10 seconds before reterning the command prompt. But the & operation makes the process to run in the backgorud and returns the prmpt straight away which allows to run other programs while waiting the sleep process
- “jobs” to see what process are runnig currently and I have seen that sleep 1000 was running in the background.
- “fg %1” to restart sleep 1000 and I have seen that sleep 1000 is now suspended.

## Killing A process

A terminal window titled 'kali@kali: ~/Documents/unixstuff/backups' with a menu bar (File, Actions, Edit, View, Help). The terminal shows a sequence of commands to create and kill processes. First, 'sleep 100 &' is run, spawning process 65011. Then 'kill 65011' is run, resulting in 'terminated sleep 100'. Next, 'sleep 1000 &' is run, spawning process 65078. Then 'ps' is run, showing four processes: zsh (PID 49694), sleep (PID 62597), sleep (PID 65078), and ps (PID 65094). Then 'kill 65078' is run, resulting in 'terminated sleep 1000'. Another 'ps' is run, showing three processes: zsh (PID 49694), sleep (PID 62597), and ps (PID 65145). Finally, 'kill 62597' is entered at the prompt.

```
(kali@kali)-[~/Documents/unixstuff/backups]
$ sleep 100 &
[2] 65011

(kali@kali)-[~/Documents/unixstuff/backups]
$ kill 65011
[2] terminated sleep 100

(kali@kali)-[~/Documents/unixstuff/backups]
$ sleep 1000 &
[2] 65078

(kali@kali)-[~/Documents/unixstuff/backups]
$ ps
  PID TTY          TIME CMD
 49694 pts/1        00:00:33 zsh
 62597 pts/1        00:00:00 sleep
 65078 pts/1        00:00:00 sleep
 65094 pts/1        00:00:00 ps

(kali@kali)-[~/Documents/unixstuff/backups]
$ kill 65078
[2] terminated sleep 1000

(kali@kali)-[~/Documents/unixstuff/backups]
$ ps
  PID TTY          TIME CMD
 49694 pts/1        00:00:33 zsh
 62597 pts/1        00:00:00 sleep
 65145 pts/1        00:00:00 ps

(kali@kali)-[~/Documents/unixstuff/backups]
$ kill 62597
```

I have practiced to kill a process which is shown above

I have used the command :

- "sleep 100" to create a process that runs in the background
- "ps" to see information about the current executable process. As a result I have seen 4 processes that are currently executed
- "kill 65078" to kill off the process that I have been created before which is "sleep 100"
- "ps" to see how many are left and I have seen that 3 remain  
At the first I have 4 processes that are executed and after I kill one of the processes, 3 remain so killing the process works.