

# DEV SANSKRITI VISHWAVIDYALAYA



SESSION 2018-19

PRACTICAL FILE

ON

## OPERATING SYSTEM

**SUBMITTED TO :**

Mrs. Apurva Omer

**Dept. Of Computer Science**

**SUBMITTED BY:**

Aniket Kumar

BCA 1<sup>st</sup> sem.

**Department of Computer Science,**

DSVV, Haridwar

# INDEX

S.No	Task	Page no	Remark
1.	Unix Basic Commands <ol style="list-style-type: none"> <li>1. login and logout</li> <li>2. Change password</li> <li>3. Shutdown or rebooting system</li> </ol>	4-6	
2.	Commands for Basic Utilities <ol style="list-style-type: none"> <li>1. Calender,</li> <li>2. manual</li> <li>3. date &amp; time</li> <li>4. current user</li> <li>5. knowing present working directory</li> <li>6. history of the session</li> </ol>	7-11	
3.	C commands for Directories & File listing <ol style="list-style-type: none"> <li>1. cat</li> <li>2. touch</li> <li>3. mv</li> <li>4. cp</li> <li>5. ls</li> <li>6. rmdir</li> <li>7. mkdir</li> </ol>	12-18	
4.	Commands for identifying UNIX shell <ul style="list-style-type: none"> <li>• Listing the shell variables</li> </ul>	19	
5.	Commands to change file and directory access permissions <ol style="list-style-type: none"> <li>1. chmod</li> </ol>	20	
6.	Using Pipes & filters & meta characters <ol style="list-style-type: none"> <li>1. grep</li> <li>2. sort</li> <li>3. Regular Expressions</li> </ol>	21-23	

7.	Command to deal with processes <ol style="list-style-type: none"> <li>1. Listing of processes</li> <li>2. Running foreground &amp; background processes</li> <li>3. Stopping processes</li> </ol>	24-26	
8.	Commands for communication <ol style="list-style-type: none"> <li>1. ifconfig</li> <li>2. ping</li> <li>3. telnet</li> <li>4. ssh</li> <li>5. nslookup</li> </ol>	27-29	
9.	Commands for vi editor <ol style="list-style-type: none"> <li>1. Command mode:               <ol style="list-style-type: none"> <li>1. Insert mode:</li> <li>2. Starting the vi editor</li> <li>3. vi Editing commands</li> <li>4. Moving within a file</li> <li>5. Saving and Closing the file</li> </ol> </li> </ol>	30-31	
10.	List of Environment Variables in Linux/Unix <ol style="list-style-type: none"> <li>1. Accessing Variable values</li> <li>2. Set New Environment Variables</li> <li>3. Deleting Variables</li> </ol>	32	

Signature

.....

## Task #1. Unix Basic Commands

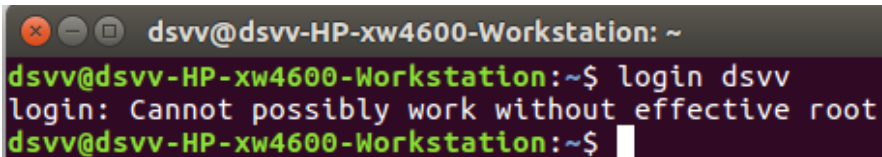
### (i). Login and Logout :

**login** : This Command is used to login the user from computer.

Syntax : `$ login <username>`

Example : `$ login dsvv`

Output :

A terminal window titled 'dsvv@dsvv-HP-xw4600-Workstation: ~' shows the command 'login dsvv' being entered. The output is 'login: Cannot possibly work without effective root'. The prompt returns to 'dsvv@dsvv-HP-xw4600-Workstation:~\$' with a cursor.

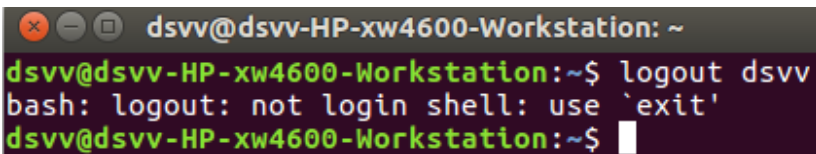
```
dsvv@dsvv-HP-xw4600-Workstation: ~  
dsvv@dsvv-HP-xw4600-Workstation:~$ login dsvv  
login: Cannot possibly work without effective root  
dsvv@dsvv-HP-xw4600-Workstation:~$
```

**logout** : This Command is used to log out the user from computer.

Syntax: `$ logout <username>`

Example : `$ logout dsvv`

Output :

A terminal window titled 'dsvv@dsvv-HP-xw4600-Workstation: ~' shows the command 'logout dsvv' being entered. The output is 'bash: logout: not login shell: use `exit`'. The prompt returns to 'dsvv@dsvv-HP-xw4600-Workstation:~\$' with a cursor.

```
dsvv@dsvv-HP-xw4600-Workstation: ~  
dsvv@dsvv-HP-xw4600-Workstation:~$ logout dsvv  
bash: logout: not login shell: use `exit`  
dsvv@dsvv-HP-xw4600-Workstation:~$
```

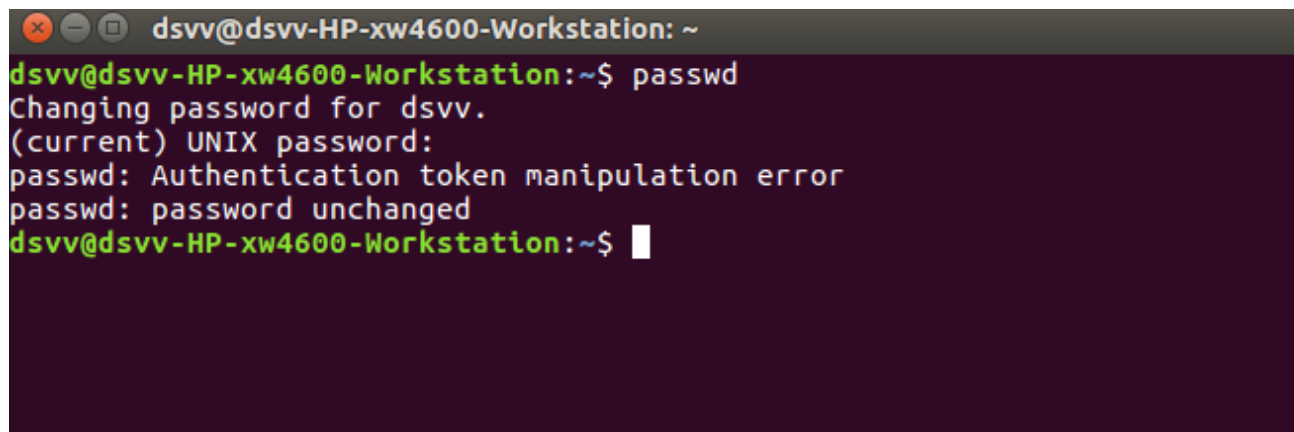
**(ii). Password :** This command is used to change the password of the currently login user or of the any account.

Syntax : `$ passwd [options] [username]`

Options:

- a : for all accounts
- d : delete the password of the named account
- e : forcibly expire the password
- k : change password only if expired
- l : lock the password of the named account
- u : unlock the password of the named account
- s : status of the current password

Output :

A terminal window titled 'dsvv@dsvv-HP-xw4600-Workstation: ~' shows a user running the 'passwd' command. The output indicates an authentication error and that the password remains unchanged.

```
dsvv@dsvv-HP-xw4600-Workstation: ~$ passwd
Changing password for dsvv.
(current) UNIX password:
passwd: Authentication token manipulation error
passwd: password unchanged
dsvv@dsvv-HP-xw4600-Workstation: ~$
```

### (iii). shutdown and rebooting :

shutdown : This command is used to shutdown the computer.

Syntax : `$ shutdown [options]`

options :

- P : power off
- r : reboot
- c : cancel a pending shutdown

Output :

```
dsvv@dsvv-HP-xw4600-Workstation: ~
dsvv@dsvv-HP-xw4600-Workstation:~$ shutdown
Shutdown scheduled for Sun 2018-10-14 04:23:37 IST, use 'shutdown
-c' to cancel.
dsvv@dsvv-HP-xw4600-Workstation:~$ shutdown -c
dsvv@dsvv-HP-xw4600-Workstation:~$
```

```
dsvv@dsvv-HP-xw4600-Workstation: ~
dsvv@dsvv-HP-xw4600-Workstation:~$ shutdown -r
Shutdown scheduled for Sun 2018-10-14 04:24:33 IST, use 'shutdown
-c' to cancel.
dsvv@dsvv-HP-xw4600-Workstation:~$ shutdown -c
dsvv@dsvv-HP-xw4600-Workstation:~$
```

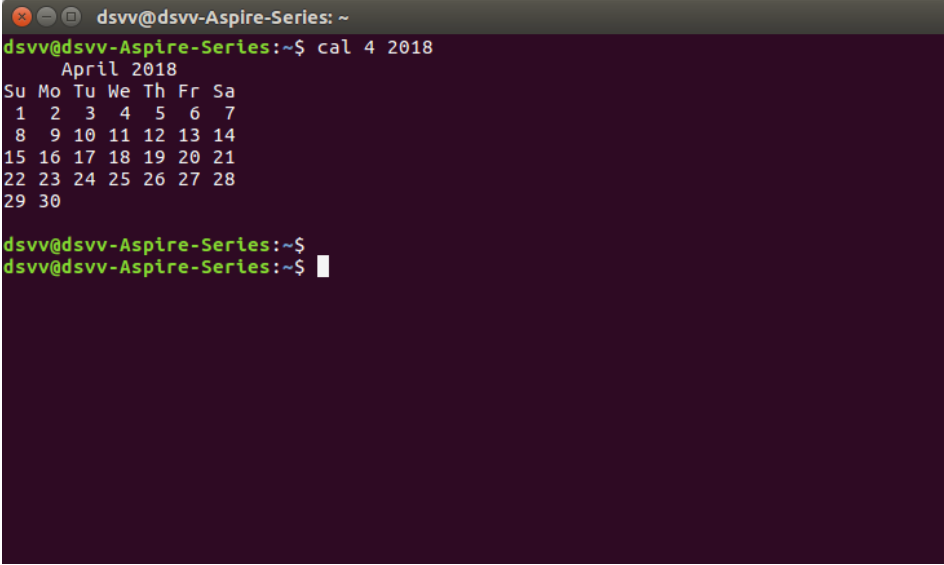
## Task #2. Commands for basic utilities

(i). **cal** : This command is used to view the calendar.

Syntax : `$ cal[[month]year]`

Example: `cal 4 2018`

Output :

A terminal window with a dark purple background. The title bar shows 'dsvv@dsvv-Aspire-Series: ~'. The prompt is 'dsvv@dsvv-Aspire-Series:~\$'. The command 'cal 4 2018' has been entered. The output shows the month of April 2018. The days of the week are listed as Su, Mo, Tu, We, Th, Fr, Sa. The dates are arranged in a grid: 1-7 in the first row, 8-14 in the second, 15-21 in the third, 22-28 in the fourth, and 29-30 in the fifth. The prompt 'dsvv@dsvv-Aspire-Series:~\$' appears again at the bottom.

```
dsvv@dsvv-Aspire-Series: ~  
dsvv@dsvv-Aspire-Series:~$ cal 4 2018  
April 2018  
Su Mo Tu We Th Fr Sa  
1 2 3 4 5 6 7  
8 9 10 11 12 13 14  
15 16 17 18 19 20 21  
22 23 24 25 26 27 28  
29 30  
  
dsvv@dsvv-Aspire-Series:~$  
dsvv@dsvv-Aspire-Series:~$
```

(ii).**manual** : This command is used to check the various options available for the commands and all the details about command.

Syntax : \$ man <command name>

Example: \$ man cal

Output:

```
dsvv@dsvv-HP-xw4600-Workstation: ~
CAL(1)                                BSD General Commands Manual                                CAL(1)

NAME
    cal, ncal - displays a calendar and the date of Easter

SYNOPSIS
    cal [-3h] [-A number] [-B number] [[month] year]
    cal [-3h] [-A number] [-B number] -m month [year]
    ncal [-3bhjJpwySM] [-A number] [-B number] [-s country_code] [[month]
    year]
    ncal [-3bhJeoSM] [-A number] [-B number] [year]
    ncal [-CN] [-H yyyy-mm-dd] [-d yyyy-mm]

DESCRIPTION
    The cal utility displays a simple calendar in traditional format and ncal
    offers an alternative layout, more options and the date of Easter. The
    new format is a little cramped but it makes a year fit on a 25x80 termi-
    nal. If arguments are not specified, the current month is displayed.

    The options are as follows:

    -h      Turns off highlighting of today.

Manual page cal(1) line 1 (press h for help or q to quit)
```



**(iii). date and time:** This command is used to display the current date and time. It also displays or calculate a date in the format we specify.

Syntax : `$ date[+format]`

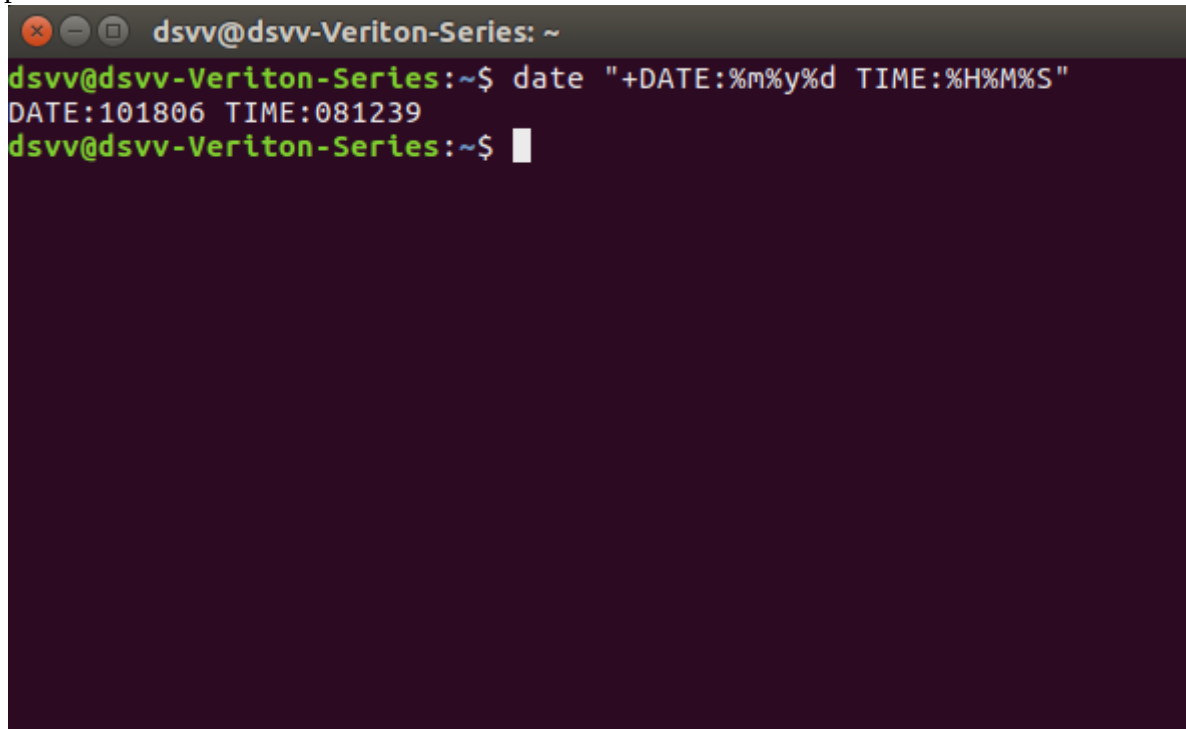
or, `$ date +%d%m%y`

Or, `$date +DATE%d%y%m`

or, `$"+DATE:%m%d%y%n TIME:%H%M%S"`

Example : `$ date "+DATE:%m%d%y%n TIME:%H%M%S"`

Output:

A terminal window titled 'dsvv@dsvv-Veriton-Series: ~' is shown. The prompt is 'dsvv@dsvv-Veriton-Series:~\$'. The user has entered the command 'date "+DATE:%m%y%d TIME:%H%M%S"'. The output is 'DATE:101806 TIME:081239'. The prompt is now 'dsvv@dsvv-Veriton-Series:~\$' with a cursor.

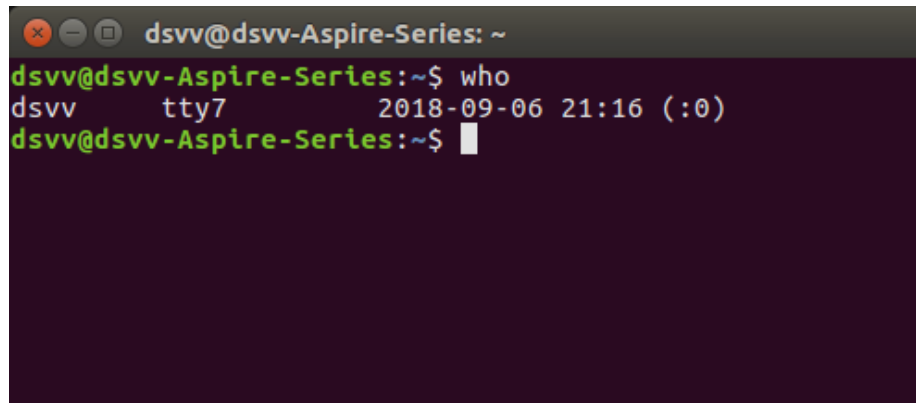
```
dsvv@dsvv-Veriton-Series:~$ date "+DATE:%m%y%d TIME:%H%M%S"
DATE:101806 TIME:081239
dsvv@dsvv-Veriton-Series:~$
```

(iv). **who** : This command displays the list of users currently logged in.

Syntax : `who[option]...[file][arg1]`

Example : `$ who`

Output:

A terminal window titled 'dsvv@dsvv-Aspire-Series: ~' showing the command 'who' being executed. The output is 'dsvv tty7 2018-09-06 21:16 (:0)'.

```
dsvv@dsvv-Aspire-Series: ~  
dsvv@dsvv-Aspire-Series:~$ who  
dsvv      tty7          2018-09-06 21:16 (:0)  
dsvv@dsvv-Aspire-Series:~$
```

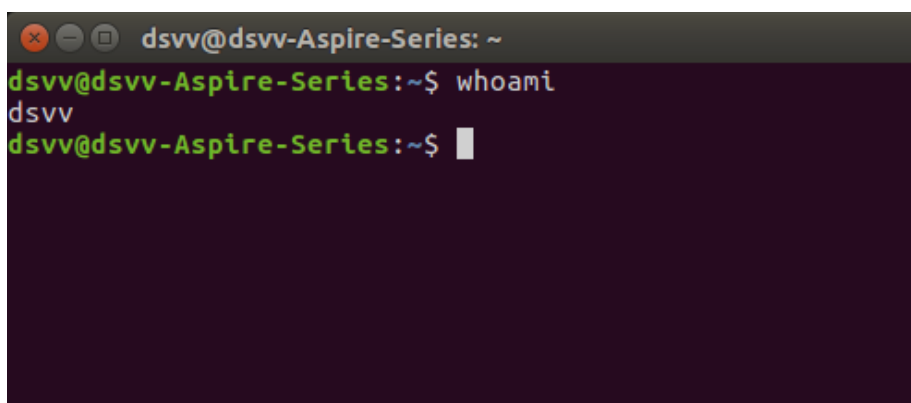
**whoami** : This command displays the user id of the currently logged in user.

Syntax : `whoami[option]`

Example : list currently logged in user

`$ whoami`

Output:

A terminal window titled 'dsvv@dsvv-Aspire-Series: ~' showing the command 'whoami' being executed. The output is 'dsvv'.

```
dsvv@dsvv-Aspire-Series: ~  
dsvv@dsvv-Aspire-Series:~$ whoami  
dsvv  
dsvv@dsvv-Aspire-Series:~$
```

(v). **pwd** : This command will display the present working directory.  
Syntax : pwd[option]

Example: \$ pwd

Output:

```
dsvv@dsvv-Aspire-Series: ~  
dsvv@dsvv-Aspire-Series:~$ pwd  
/home/dsvv  
dsvv@dsvv-Aspire-Series:~$
```

(vi). **history** : This command show you all the commands that you have used in the current terminal session.  
Syntax : \$ history

Output:

```
dsvv@dsvv-Aspire-Series: ~  
dsvv@dsvv-Aspire-Series:~$ history  
1  man  
2  manual  
3  mannual  
4  man  
5  cd ..  
6  dir  
7  dir help  
8  help dir  
9  man ls  
10 mkdir Aniket  
11 history  
12 ls  
13 ls -al  
14 ls -R  
15 ls Aniket  
16 ls -a  
17 la -A  
18 ls --author  
19 ls -A  
20 ls -l  
21 ls -L  
22 ls -Q  
23 ls *odt
```

### Task #3. Commands for basic utilities

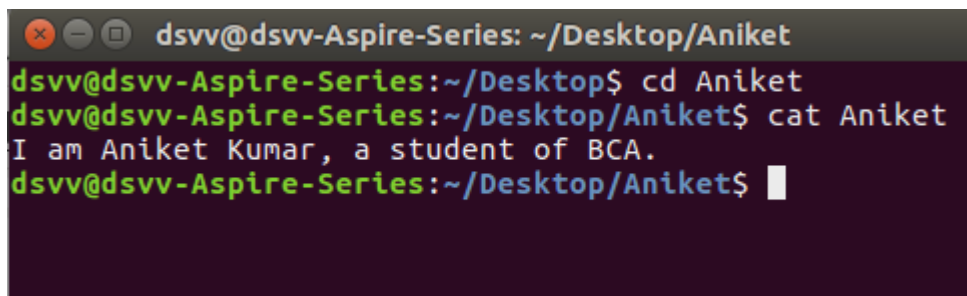
(i). **Cat commands** : This command is used for file handling, such as creating a file, opening a file, appending a file, etc.

Syntax : `$ cat [options]`

Various options of cat command are:

**.cat** : used to open a file :

Output:

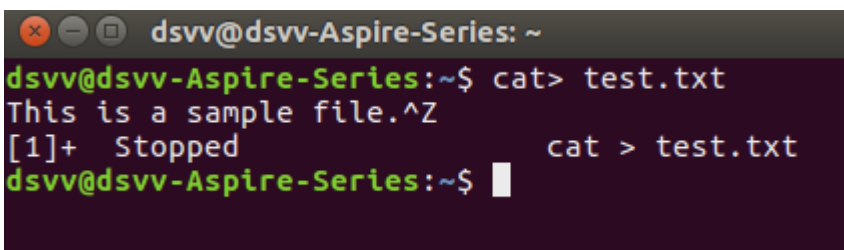
A terminal window titled 'dsvv@dsvv-Aspire-Series: ~/Desktop/Aniket'. The user enters 'cd Aniket' and then 'cat Aniket'. The output is 'I am Aniket Kumar, a student of BCA.' followed by a prompt.

```
dsvv@dsvv-Aspire-Series: ~/Desktop/Aniket
dsvv@dsvv-Aspire-Series:~/Desktop$ cd Aniket
dsvv@dsvv-Aspire-Series:~/Desktop/Aniket$ cat Aniket
I am Aniket Kumar, a student of BCA.
dsvv@dsvv-Aspire-Series:~/Desktop/Aniket$
```

**cat> filename** : creates the file.

Example : `cat> test.txt`

Output:

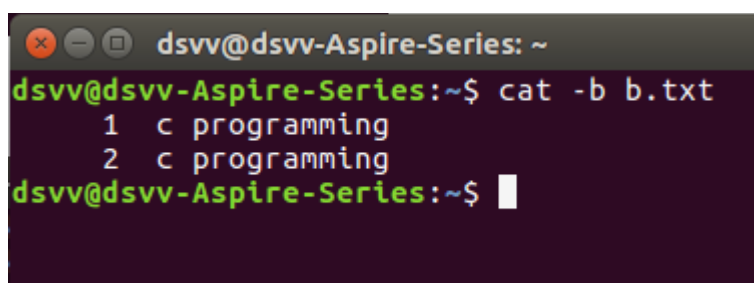
A terminal window titled 'dsvv@dsvv-Aspire-Series: ~'. The user enters 'cat> test.txt', types 'This is a sample file.^Z', and then '[1]+ Stopped cat > test.txt'. The prompt returns.

```
dsvv@dsvv-Aspire-Series: ~
dsvv@dsvv-Aspire-Series:~$ cat> test.txt
This is a sample file.^Z
[1]+  Stopped                  cat > test.txt
dsvv@dsvv-Aspire-Series:~$
```

**cat -b** : opens the file and displays the contents with line numbers.

Example : `cat -b b.txt`

Output :

A terminal window titled 'dsvv@dsvv-Aspire-Series: ~'. The user enters 'cat -b b.txt'. The output shows two lines, each preceded by a line number.

```
dsvv@dsvv-Aspire-Series: ~
dsvv@dsvv-Aspire-Series:~$ cat -b b.txt
 1  c programming
 2  c programming
dsvv@dsvv-Aspire-Series:~$
```

**cat file1>file2** : Append/Replace one file into another.

Example: cat b.txt>cat a.txt

Output:

```
dsvv@dsvv-Aspire-Series: ~  
dsvv@dsvv-Aspire-Series:~$ cat a.txt  
hkjdlgd  
hgmvsigijf,  
hgmvsigijf,  
  
hgmvsigijf,  
hgmvsigijf,  
hgmvsigijf,  
kmfhjfl  
  ihello  
gjugj  
hkl;.'klgf  
c programming  
dsvv@dsvv-Aspire-Series:~$ cat b.txt  
c programming  
c programming  
dsvv@dsvv-Aspire-Series:~$ cat b.txt>a.txt  
dsvv@dsvv-Aspire-Series:~$ cat a.txt  
c programming  
c programming  
dsvv@dsvv-Aspire-Series:~$ cat b.txt  
c programming  
c programming  
dsvv@dsvv-Aspire-Series:~$
```

**cat file1>>file2** : To add one file into another.

Example: cat c.txt>>cat b.txt

Output:

```
dsvv@dsvv-Aspire-Series:~$ cat b.txt  
c programming  
c programming  
dsvv@dsvv-Aspire-Series:~$ cat c.txt  
hkjdlgd  
hgmvsigijf,  
hgmvsigijf,  
  
hgmvsigijf,  
hgmvsigijf,  
hgmvsigijf,  
kmfhjfl  
  ihello  
gjugj  
hkl;.'klgf  
c programming  
dsvv@dsvv-Aspire-Series:~$ cat c.txt>>b.txt  
dsvv@dsvv-Aspire-Series:~$ cat b.txt  
c programming  
c programming  
hkjdlgd  
hgmvsigijf,  
hgmvsigijf,  
  
hgmvsigijf,  
hgmvsigijf,  
hgmvsigijf,  
kmfhjfl  
  ihello  
gjugj  
hkl;.'klgf  
c programming
```

(ii). **touch** : This command is used to create any file.

Syntax: \$ touch <filename>

Example: \$ touch os

Output :

```
dsvv@dsvv-Aspire-Series: ~  
dsvv@dsvv-Aspire-Series:~$ touch os  
dsvv@dsvv-Aspire-Series:~$ ls  
Aniket Desktop examples.desktop Public Videos  
a.txt desktopa Music Templates  
b.txt Documents name.c test_1.odt  
c.txt Downloads os Unix_commands _Aniket.odt  
desktop d.txt Pictures Untitled 1.odt  
dsvv@dsvv-Aspire-Series:~$
```

(iii). **mv** : This command is used rename the filename.

Syntax: \$ mv <old filename> <new filename>

Example : mv a.txt b.txt

Output :

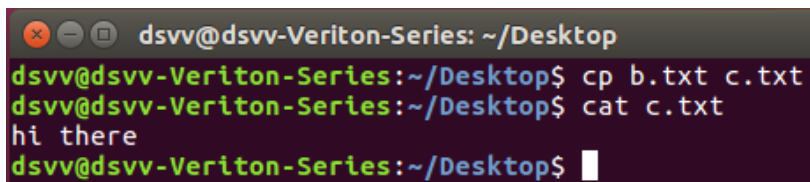
```
dsvv@dsvv-Veriton-Series: ~/Desktop  
dsvv@dsvv-Veriton-Series:~/Desktop$ mv a.txt b.txt  
dsvv@dsvv-Veriton-Series:~/Desktop$ cat a.txt  
cat: a.txt: No such file or directory  
dsvv@dsvv-Veriton-Series:~/Desktop$ cat b.txt  
dsvv@dsvv-Veriton-Series:~/Desktop$ cat b.txt  
hi there  
dsvv@dsvv-Veriton-Series:~/Desktop$
```

(iv). **cp** : This command is used to copy a file to another file.

Syntax : \$ cp <source file> <destination fiile>

Example : cp b.txt c.txt

Output :

A terminal window with a dark purple background. The title bar shows window control buttons and the text 'dsvv@dsvv-Veriton-Series: ~/Desktop'. The terminal content shows the following commands and output:

```
dsvv@dsvv-Veriton-Series:~/Desktop$ cp b.txt c.txt
dsvv@dsvv-Veriton-Series:~/Desktop$ cat c.txt
hi there
dsvv@dsvv-Veriton-Series:~/Desktop$
```

(v). **ls** – It is used to list all the directories and subdirectories present in that folder.

Syntax: \$ ls [options]

Output :

```
dsvv@dsvv-Aspire-Series: ~  
dsvv@dsvv-Aspire-Series:~$ ls  
Aniket  desktop  Downloads  name.c  test_1.odt  
a.txt   Desktop  d.txt      Pictures Unix commands _Aniket.odt  
b.txt   desktop  examples.desktop Public   Untitled 1.odt  
c.txt   Documents Music      Templates Videos  
dsvv@dsvv-Aspire-Series:~$
```

options :

(a) **ls -al**

Output :

```
dsvv@dsvv-Aspire-Series:~$ ls -al  
total 384  
drwxr-xr-x 17 dsvv dsvv 4096 Sep 18 20:33 .  
drwxr-xr-x  4 root root 4096 Jul 30 15:51 ..  
drwxrwxr-x  6 dsvv dsvv 4096 Aug 29 21:53 Aniket  
-rw-rw-r--  1 dsvv dsvv  111 Sep 10 21:46 a.txt  
-rw-rw-r--  1 dsvv dsvv 12288 Sep  6 21:45 a.txt.swp  
-rw-r--r--  1 dsvv dsvv 2545 Sep 13 15:22 .bash_history  
-rw-rw-r--  1 dsvv dsvv  220 Jul 30 15:51 .bash_logout  
-rw-rw-r--  1 dsvv dsvv 3771 Jul 30 15:51 .bashrc  
-rw-rw-r--  1 dsvv dsvv  28 Sep  6 21:25 b.txt  
drwx----- 13 dsvv dsvv 4096 Sep  4 21:34 .cache  
drwx----- 17 dsvv dsvv 4096 Aug 29 21:45 .config  
-rw-rw-r--  1 dsvv dsvv  111 Sep 13 20:38 c.txt  
-rw-rw-r--  1 dsvv dsvv  111 Sep 13 20:41 desktop  
drwxr-xr-x  4 dsvv dsvv 4096 Sep 18 20:39 Desktop  
-rw-rw-r--  1 dsvv dsvv  111 Sep 13 20:40 desktopa  
-rw-rw-r--  1 dsvv dsvv  25 Aug  1 20:35 .dmrc  
drwxr-xr-x  2 dsvv dsvv 4096 Aug  1 20:35 Documents  
drwxr-xr-x  2 dsvv dsvv 4096 Sep 10 22:24 Downloads  
-rw-rw-r--  1 dsvv dsvv  28 Sep 13 15:12 d.txt  
-rw-rw-r--  1 dsvv dsvv 8980 Jul 30 15:51 examples.desktop  
drwx-----  2 dsvv dsvv 4096 Sep 18 20:18 .gconf  
drwx-----  3 dsvv dsvv 4096 Sep 18 20:17 .gnupg  
-rw-rw-r--  1 dsvv dsvv 5124 Sep 18 20:17 .ICEauthority  
drwx-----  3 dsvv dsvv 4096 Aug  1 20:35 .local  
-rw-rw-r--  1 dsvv dsvv  82 Sep 18 20:33 .lock.Unix commands _Aniket.odt#  
drwx-----  4 dsvv dsvv 4096 Aug  1 20:55 .mozilla  
drwxr-xr-x  2 dsvv dsvv 4096 Aug  1 20:35 Music  
-rw-rw-r--  1 dsvv dsvv 81 Aug 31 21:53 name.c  
drwxr-xr-x  2 dsvv dsvv 4096 Aug  1 20:35 Pictures  
-rw-rw-r--  1 dsvv dsvv 675 Jul 30 15:51 .profile  
drwxr-xr-x  2 dsvv dsvv 4096 Aug  1 20:35 Public  
-rw-rw-r--  1 dsvv dsvv  0 Aug 28 21:50 .sudo_as_admin_successful  
-rw-rw-r--  1 dsvv dsvv 12288 Aug 31 21:56 .swp  
drwxr-xr-x  2 dsvv dsvv 4096 Aug  1 20:35 Templates  
-rw-rw-r--  1 dsvv dsvv  50 Aug 28 21:56 test_1.odt  
-rw-rw-r--  1 dsvv dsvv 195669 Sep  6 21:55 Unix commands _Aniket.odt  
-rw-rw-r--  1 dsvv dsvv 8222 Aug 28 21:29 Untitled 1.odt  
drwxr-xr-x  2 dsvv dsvv 4096 Aug  1 20:35 Videos  
-rw-rw-r--  1 dsvv dsvv  63 Sep 18 20:17 .xauthority  
-rw-rw-r--  1 dsvv dsvv  82 Sep 18 20:17 .xsession-errors  
-rw-rw-r--  1 dsvv dsvv 1433 Sep 13 15:22 .xsession-errors.old
```



(b) **ls -R** : to display directories with subdirectories.

Output :

```
dsvv@dsvv-Aspire-Series:~$ ls -R
.:
Aniket  b.txt  desktop  desktopa  Downloads  examples.desktop  name.c  Public  test_1.odt  Untitled 1.odt
a.txt  c.txt  Desktop  Documents  d.txt      Music            Pictures  Templates  Unix commands _Aniket.odt  Videos

./Aniket:
Amit  test1  test2  test3

./Aniket/Amit:

./Aniket/test1:

./Aniket/test2:

./Aniket/test3:

./Desktop:
Aniket

./Desktop/Aniket:
ani  Aniket

./Desktop/Aniket/ani:
Ani1  bca.odt  lab.txt  test

./Desktop/Aniket/ani/Ani1:

./Desktop/Aniket/ani/test:
test  testc
Take a Screenshot of the Current Window
Take a Screenshot of a Selected Area

./Desktop/Aniket/ani/test/testc:
Take a Screenshot of a Selected Area

./Documents:
Screenshot

./Downloads:
screenshot
xampp-linux-x64-7.2.9-0-installer.run
gnome-screenshot

./Music:
e-screenshot

./Pictures:
uncher

./Public:
```

(c) **ls /** : list out all root directories.

Output :

```
dsvv@dsvv-Aspire-Series: ~
dsvv@dsvv-Aspire-Series:~$ ls \
>
Aniket  desktop  Downloads  name.c  test_1.odt
a.txt  Desktop  d.txt      Pictures  Unix commands _Aniket.odt
b.txt  desktopa  examples.desktop  Public  Untitled 1.odt
c.txt  Documents  Music      Templates  Videos
dsvv@dsvv-Aspire-Series:~$
```

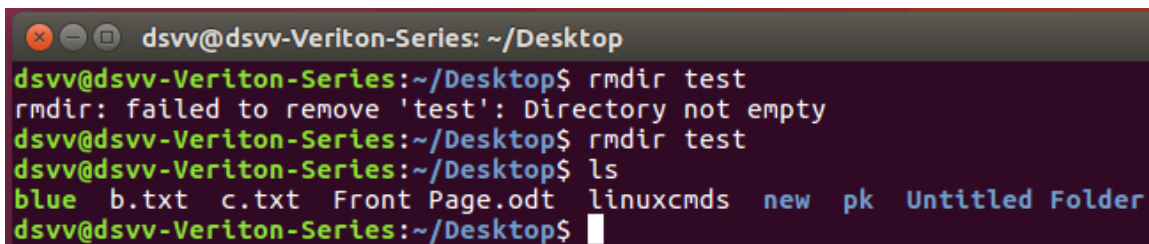
(vi). **rmdir** : This command is used to remove the directory.

Syntax : `$ rmdir [options] [directory name]`

option : `-p` : remove directory and its ancestors

Example : `$ rmdir test`

Output :

A terminal window titled 'dsvv@dsvv-Veriton-Series: ~/Desktop' showing the execution of the 'rmdir' command. The user attempts to remove a directory named 'test'. The first attempt fails with the message 'rmdir: failed to remove 'test': Directory not empty'. The second attempt also fails with the same message. Finally, the user runs 'ls', which lists the contents of the directory: 'blue b.txt c.txt Front Page.odt linuxcmds new pk Untitled Folder'.

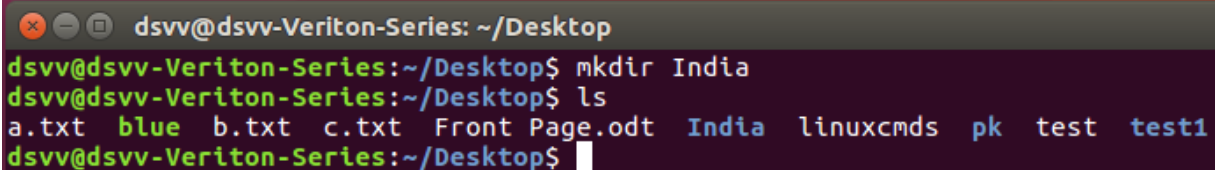
```
dsvv@dsvv-Veriton-Series: ~/Desktop
dsvv@dsvv-Veriton-Series:~/Desktop$ rmdir test
rmdir: failed to remove 'test': Directory not empty
dsvv@dsvv-Veriton-Series:~/Desktop$ rmdir test
dsvv@dsvv-Veriton-Series:~/Desktop$ ls
blue b.txt c.txt Front Page.odt linuxcmds new pk Untitled Folder
dsvv@dsvv-Veriton-Series:~/Desktop$
```

(vii). **mkdir** : This Command is used to create a directory.

Syntax : `$ mkdir`

Example : `$ mkdir India`

Output :

A terminal window titled 'dsvv@dsvv-Veriton-Series: ~/Desktop' showing the execution of the 'mkdir' command. The user runs 'mkdir India', which successfully creates a new directory. Then, the user runs 'ls', which lists the contents of the directory, including the newly created 'India' directory: 'a.txt blue b.txt c.txt Front Page.odt India linuxcmds pk test test1'.

```
dsvv@dsvv-Veriton-Series:~/Desktop$ mkdir India
dsvv@dsvv-Veriton-Series:~/Desktop$ ls
a.txt blue b.txt c.txt Front Page.odt India linuxcmds pk test test1
dsvv@dsvv-Veriton-Series:~/Desktop$
```

## Task #4 : Commands for identifying UNIX shell

**set** : This command is used for listing all the shell variables.

Syntax : `$ set | more`

Output :

```
dsvv@dsvv-HP-xw4600-Workstation: ~  
dsvv@dsvv-HP-xw4600-Workstation:~$ set  
BASH=/bin/bash  
BASHOPTS=checkwinsize:cmdhist:complete_fullquote:expand_aliases:extglob:extquote  
:force_ignore:histappend:interactive_comments:progcomp:promptvars:sourcepath  
BASH_ALIASES=()  
BASH_ARGC=()  
BASH_ARGV=()  
BASH_CMDS=()  
BASH_COMPLETION_COMPAT_DIR=/etc/bash_completion.d  
BASH_LINENO=()  
BASH_SOURCE=()  
BASH_VERSINFO=([0]="4" [1]="3" [2]="42" [3]="1" [4]="release" [5]="x86_64-pc-lin  
ux-gnu")  
BASH_VERSION='4.3.42(1)-release'  
CLUTTER_IM_MODULE=xim  
COLUMNS=80  
COMPIZ_CONFIG_PROFILE=ubuntu  
DBUS_SESSION_BUS_ADDRESS=unix:abstract=/tmp/dbus-wAsUoshoyM  
DEFAULTS_PATH=/usr/share/gconf/ubuntu.default.path  
DESKTOP_SESSION=ubuntu  
DIRSTACK=()  
DISPLAY=:0  
EUID=1000  
GDMSESSION=ubuntu  
GDM_LANG=en_US  
GNOME_DESKTOP_SESSION_ID=this-is-deprecated  
GNOME_KEYRING_CONTROL=  
GNOME_KEYRING_PID=  
GPG_AGENT_INFO=/home/dsvv/.gnupg/S.gpg-agent:0:1  
GROUPS=()  
GTK2_MODULES=overlay-scrollbar  
GTK_IM_MODULE=ibus  
GTK_MODULES=gail:atk-bridge:unity-gtk-module  
HISTCONTROL=ignoreboth  
HISTFILE=/home/dsvv/.bash_history  
HISTFILESIZE=2000  
HISTSIZE=1000  
HOME=/home/dsvv  
HOSTNAME=dsvv-HP-xw4600-Workstation  
HOSTTYPE=x86_64  
IFS=$' \t\n'  
IM_CONFIG_PHASE=1  
INSTANCE=  
JOB=dbus  
LANG=en_IN  
LANGUAGE=en_IN:en  
LESSCLOSE='/usr/bin/lesspipe %s %s'  
LESSOPEN='| /usr/bin/lesspipe %s'  
LINES=24
```

## Task #5 : commands to change file and directory access permissions

**chmod** : This command is used to change the access permission of files and directories.

Syntax : `chmod[permissions]....[file name]`

Example: `chmod 720 test.odt`

options :

Read : 4 or -r

Write : 2 or -w

execute: 1 or -x

+ : add the permissions

- : remove the permissions

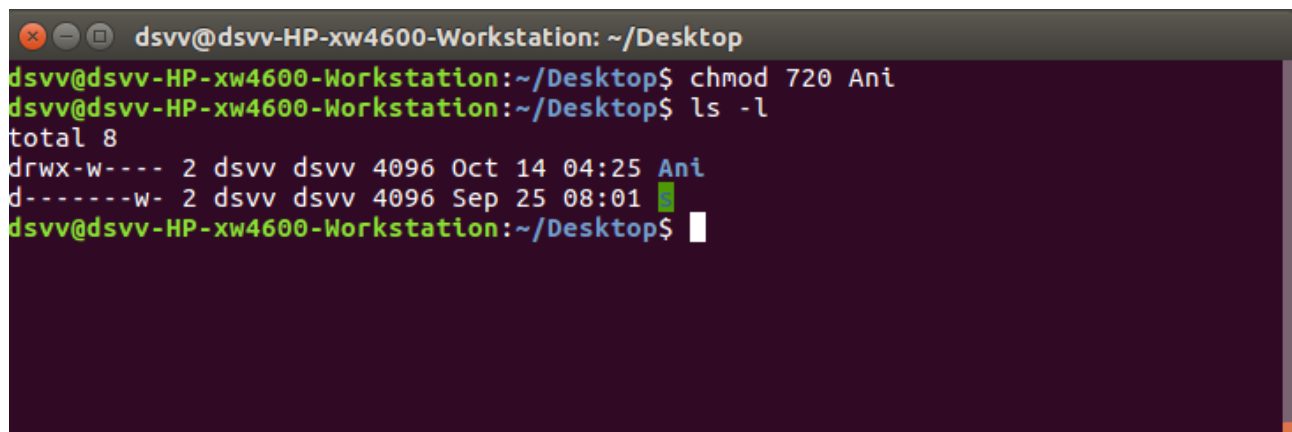
= : overwrite the permissions

acronyms: user – u

group – g

other – o

Output:

A terminal window with a dark purple background and light green text. The window title is 'dsvv@dsvv-HP-xw4600-Workstation: ~/Desktop'. The user has entered the command 'chmod 720 Ani' and then 'ls -l'. The output of 'ls -l' shows two entries: a directory 'Ani' with permissions 'drwx-w----' and a file with permissions 'd-----w-'.

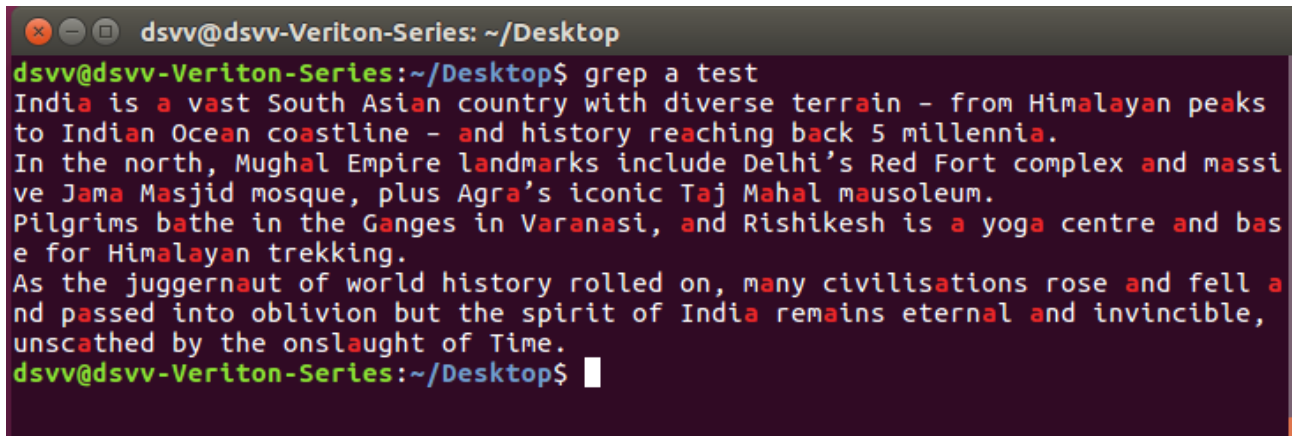
```
dsvv@dsvv-HP-xw4600-Workstation: ~/Desktop
dsvv@dsvv-HP-xw4600-Workstation:~/Desktop$ chmod 720 Ani
dsvv@dsvv-HP-xw4600-Workstation:~/Desktop$ ls -l
total 8
drwx-w---- 2 dsvv dsvv 4096 Oct 14 04:25 Ani
d-----w- 2 dsvv dsvv 4096 Sep 25 08:01 
dsvv@dsvv-HP-xw4600-Workstation:~/Desktop$
```

## Task #6 : Using pipes and filters and meta characters

(i). **grep** : This command is used for searching any word or string in a file.  
Syntax : `$ grep <word/string> filename`

Other option : `$ cat filename|grep <word/string>`

Output:

A terminal window with a dark background and light-colored text. The window title is 'dsvv@dsvv-Veriton-Series: ~/Desktop'. The prompt is 'dsvv@dsvv-Veriton-Series:~/Desktop\$'. The command 'grep a test' has been entered. The output shows a paragraph of text with the letter 'a' highlighted in red in several places: 'India is a vast South Asian country with diverse terrain - from Himalayan peaks to Indian Ocean coastline - and history reaching back 5 millennia. In the north, Mughal Empire landmarks include Delhi's Red Fort complex and massive Jama Masjid mosque, plus Agra's iconic Taj Mahal mausoleum. Pilgrims bathe in the Ganges in Varanasi, and Rishikesh is a yoga centre and base for Himalayan trekking. As the juggernaut of world history rolled on, many civilisations rose and fell and passed into oblivion but the spirit of India remains eternal and invincible, unscathed by the onslaught of Time.' The prompt 'dsvv@dsvv-Veriton-Series:~/Desktop\$' is followed by a white cursor block.

```
dsvv@dsvv-Veriton-Series: ~/Desktop
dsvv@dsvv-Veriton-Series:~/Desktop$ grep a test
India is a vast South Asian country with diverse terrain - from Himalayan peaks
to Indian Ocean coastline - and history reaching back 5 millennia.
In the north, Mughal Empire landmarks include Delhi's Red Fort complex and massi
ve Jama Masjid mosque, plus Agra's iconic Taj Mahal mausoleum.
Pilgrims bathe in the Ganges in Varanasi, and Rishikesh is a yoga centre and bas
e for Himalayan trekking.
As the juggernaut of world history rolled on, many civilisations rose and fell a
nd passed into oblivion but the spirit of India remains eternal and invincible,
unscathed by the onslaught of Time.
dsvv@dsvv-Veriton-Series:~/Desktop$
```

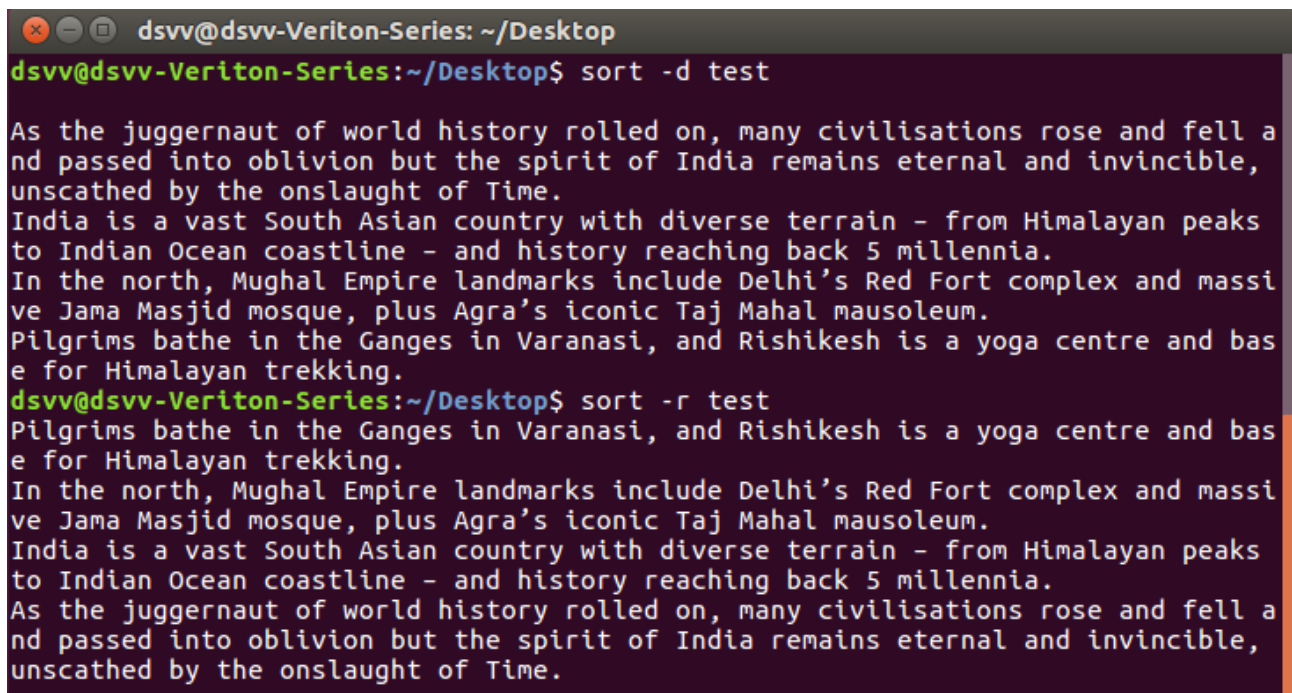
(ii). **sort** : This command is used for arranging the files.

Syntax : `$ sort [options]`

Options:

- d: dictionary order
- f: ignore case
- g: general numeric sort
- M: month sort
- h: human numeric order
- n: numeric sort
- R: random sort
- r: reverse order

Output:

A terminal window with a dark background and light-colored text. The window title is 'dsvv@dsvv-Veriton-Series: ~/Desktop'. The prompt is 'dsvv@dsvv-Veriton-Series:~/Desktop\$'. The command 'sort -d test' has been entered. The output shows a paragraph of text about India's history and culture, sorted in dictionary order. The text is: 'As the juggernaut of world history rolled on, many civilisations rose and fell and passed into oblivion but the spirit of India remains eternal and invincible, unscathed by the onslaught of Time. India is a vast South Asian country with diverse terrain - from Himalayan peaks to Indian Ocean coastline - and history reaching back 5 millennia. In the north, Mughal Empire landmarks include Delhi's Red Fort complex and massive Jama Masjid mosque, plus Agra's iconic Taj Mahal mausoleum. Pilgrims bathe in the Ganges in Varanasi, and Rishikesh is a yoga centre and base for Himalayan trekking.' The prompt 'dsvv@dsvv-Veriton-Series:~/Desktop\$' is followed by the command 'sort -r test'. The output shows the same paragraph of text, but sorted in reverse order. The text is: 'Pilgrims bathe in the Ganges in Varanasi, and Rishikesh is a yoga centre and base for Himalayan trekking. In the north, Mughal Empire landmarks include Delhi's Red Fort complex and massive Jama Masjid mosque, plus Agra's iconic Taj Mahal mausoleum. India is a vast South Asian country with diverse terrain - from Himalayan peaks to Indian Ocean coastline - and history reaching back 5 millennia. As the juggernaut of world history rolled on, many civilisations rose and fell and passed into oblivion but the spirit of India remains eternal and invincible, unscathed by the onslaught of Time.'

**(iii). Regular expressions:** They are special characters, which help to search data matching complex patterns and are shortened as **regexp|regex**

**Basic Regular expressions:**

- . : replaces any character
- ^ : matches the start of string
- \$ : matches the end of string
- \* : matches up 0 or more times the preceding character
- \ : represent special character
- () : groups regular expression
- ? : matches up exactly one character

**Interval Regular expression**

**{n}**

- number of times the letter appeared.

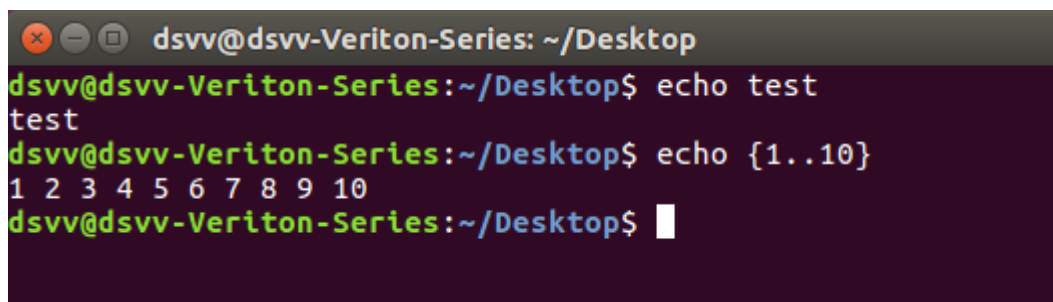
Syntax: cat filename|grep -E l/{2}

**Braces Regular expression**

To print/display anything.

Syntax: echo{}

Output:



```
dsvv@dsvv-Veriton-Series: ~/Desktop
dsvv@dsvv-Veriton-Series:~/Desktop$ echo test
test
dsvv@dsvv-Veriton-Series:~/Desktop$ echo {1..10}
1 2 3 4 5 6 7 8 9 10
dsvv@dsvv-Veriton-Series:~/Desktop$
```

## Task #7 Command to deal with processes

(i). **top** : This command is used for listing the current running processes.

Syntax: \$ top

Output:

```
dsvv@dsvv-HP-xw4600-Workstation: ~
dsvv@dsvv-HP-xw4600-Workstation:~$ top

top - 03:23:16 up 1:18, 1 user, load average: 0.09, 0.09, 0.15
Tasks: 198 total, 1 running, 197 sleeping, 0 stopped, 0 zombie
%Cpu(s): 4.3 us, 2.9 sy, 10.4 ni, 81.4 id, 0.9 wa, 0.0 hi, 0.0 si, 0.0 st
KiB Mem : 4029864 total, 2378056 free, 734044 used, 917764 buff/cache
KiB Swap: 39062524 total, 39062524 free, 0 used. 3230688 avail Mem

  PID USER      PR  NI   VIRT   RES   SHR  S  %CPU  %MEM    TIME+  COMMAND
  771 root        20   0  248524  67992  27396 S   5.9   1.7   4:37.70 Xorg
 2917 dsvv        20   0  661064  34480  27764 S   5.9   0.9   0:00.18 gnome-ter
rm+
 2934 dsvv        20   0   48868   3808   3208 R   5.9   0.1   0:00.01 top
    1 root        20   0  119820   6000   4100 S   0.0   0.1   0:01.49 systemd
    2 root        20   0     0     0     0 S   0.0   0.0   0:00.00 kthreadd
    3 root        20   0     0     0     0 S   0.0   0.0   0:00.38 ksoftirq
d/0
    5 root         0 -20     0     0     0 S   0.0   0.0   0:00.00 kworker/
0:+
    7 root        20   0     0     0     0 S   0.0   0.0   0:02.14 rcu_sche
d
    8 root        20   0     0     0     0 S   0.0   0.0   0:00.00 rcu_bh
    9 root        rt    0     0     0     0 S   0.0   0.0   0:00.00 migratio
n/0
   10 root        rt    0     0     0     0 S   0.0   0.0   0:00.00 watchdog
/0
   11 root        rt    0     0     0     0 S   0.0   0.0   0:00.01 watchdog
/1
   12 root        rt    0     0     0     0 S   0.0   0.0   0:00.00 migratio
n/1
   13 root        20   0     0     0     0 S   0.0   0.0   0:00.63 ksoftirq
d/1
   15 root         0 -20     0     0     0 S   0.0   0.0   0:00.00 kworker/
1:+
   16 root        20   0     0     0     0 S   0.0   0.0   0:00.00 kdevtmpfs
s
   17 root         0 -20     0     0     0 S   0.0   0.0   0:00.00 netns

[1]+  Stopped                  top
```



- (ii). **fg** : This command is used to check foreground processes.  
Syntax : \$ fg

Output:

```
dsvv@dsvv-HP-xw4600-Workstation: ~
clear
dsvv@dsvv-HP-xw4600-Workstation:~$ firefox

^Z
[1]+  Stopped                  firefox
dsvv@dsvv-HP-xw4600-Workstation:~$ fg
firefox
dsvv@dsvv-HP-xw4600-Workstation:~$ █
```

**bg**: This command is used to check background processes

Syntax : \$ bg

Output:

```
dsvv@dsvv-HP-xw4600-Workstation: ~
dsvv@dsvv-HP-xw4600-Workstation:~$ firefox

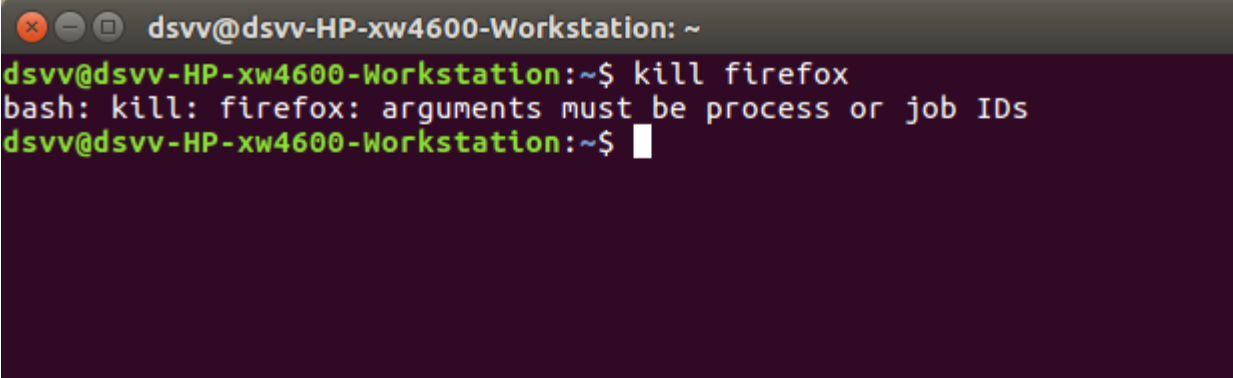
^Z
[1]+  Stopped                  firefox
dsvv@dsvv-HP-xw4600-Workstation:~$ bg
[1]+  firefox &
dsvv@dsvv-HP-xw4600-Workstation:~$ █
```

(iii). **kill** : This command is used for stopping a process.

Syntax: \$ kill <PID>

Example : \$ kill firefox

Output:



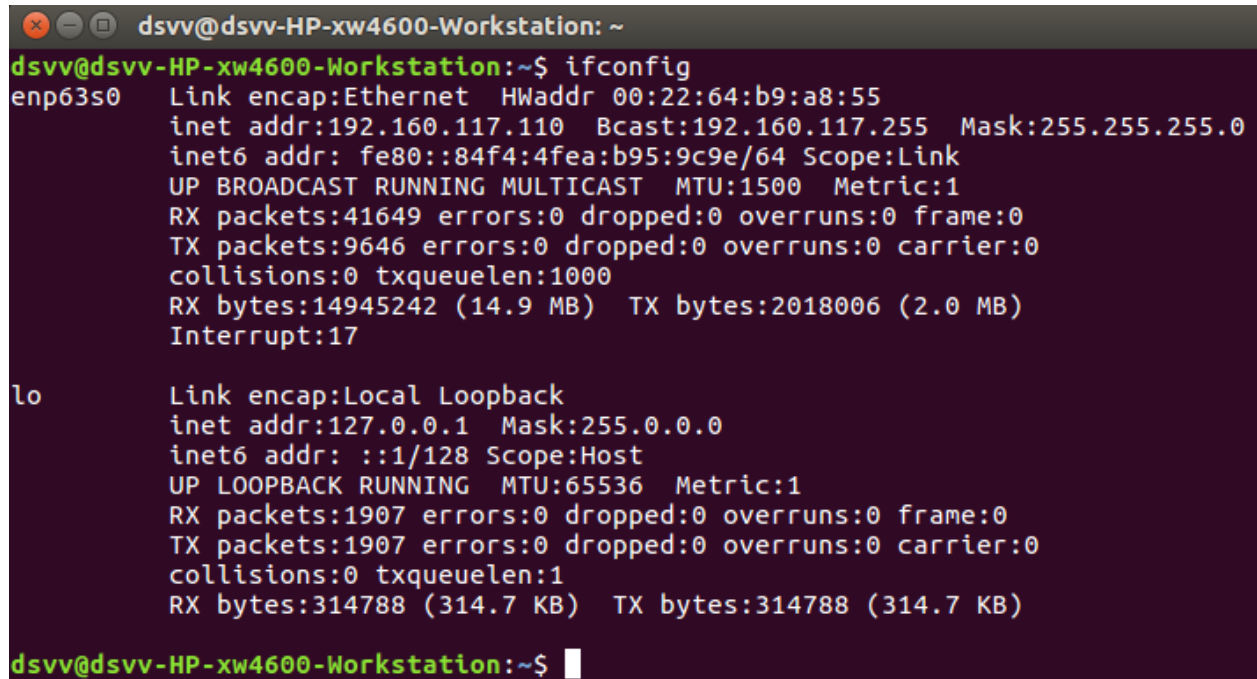
```
dsvv@dsvv-HP-xw4600-Workstation: ~  
dsvv@dsvv-HP-xw4600-Workstation:~$ kill firefox  
bash: kill: firefox: arguments must be process or job IDs  
dsvv@dsvv-HP-xw4600-Workstation:~$
```

## Task #8 : Commands for Communication

(i). **ifconfig** : This command is used to see the addres. (physical as well as logical).

Syntax : `$ ifconfig`

Output:



```
dsvv@dsvv-HP-xw4600-Workstation: ~  
dsvv@dsvv-HP-xw4600-Workstation:~$ ifconfig  
enp63s0  Link encap:Ethernet  HWaddr 00:22:64:b9:a8:55  
         inet addr:192.160.117.110  Bcast:192.160.117.255  Mask:255.255.255.0  
         inet6 addr: fe80::84f4:4fea:b95:9c9e/64  Scope:Link  
         UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1  
         RX packets:41649 errors:0 dropped:0 overruns:0 frame:0  
         TX packets:9646 errors:0 dropped:0 overruns:0 carrier:0  
         collisions:0 txqueuelen:1000  
         RX bytes:14945242 (14.9 MB)  TX bytes:2018006 (2.0 MB)  
         Interrupt:17  
  
lo       Link encap:Local Loopback  
         inet addr:127.0.0.1  Mask:255.0.0.0  
         inet6 addr: ::1/128  Scope:Host  
         UP LOOPBACK RUNNING  MTU:65536  Metric:1  
         RX packets:1907 errors:0 dropped:0 overruns:0 frame:0  
         TX packets:1907 errors:0 dropped:0 overruns:0 carrier:0  
         collisions:0 txqueuelen:1  
         RX bytes:314788 (314.7 KB)  TX bytes:314788 (314.7 KB)  
  
dsvv@dsvv-HP-xw4600-Workstation:~$
```

(ii). **ping** : This command is used to see ping time of any IP address or hostname.

Syntax : \$ ping <IP address / hostname>

Example: \$ ping

Output:

```
dsvv@dsvv-HP-xw4600-Workstation: ~  
dsvv@dsvv-HP-xw4600-Workstation:~$ ping  
Usage: ping [-aAbBdDfhLnOqrRUvV] [-c count] [-i interval] [-I interface]  
          [-m mark] [-M pmtudisc_option] [-l preload] [-p pattern] [-Q tos]  
          [-s packetsize] [-S sndbuf] [-t ttl] [-T timestamp_option]  
          [-w deadline] [-W timeout] [hop1 ...] destination  
dsvv@dsvv-HP-xw4600-Workstation:~$
```

(iii). **telnet** : This command is used for connecting to IP address or hostname.

Syntax: \$ telnet <IP address / hostname>

Example: telnet google.com

Output:

```
dsvv@dsvv-HP-xw4600-Workstation: ~  
dsvv@dsvv-HP-xw4600-Workstation:~$ telnet google.com  
Trying 172.217.31.14...  
█
```

- (iv). **ssh** : This command is similar to telnet and is used for connecting to IP address or hostname. It is more secure than telnet.

Syntax : \$ ssh <IP address/host name>

Output:

```
dsvv@dsvv-HP-xw4600-Workstation: ~  
dsvv@dsvv-HP-xw4600-Workstation:~$ ssh  
usage: ssh [-1246AaCfGgKkMnqsTtVvXxYy] [-b bind_address] [-c cipher_spec]  
[-D [bind_address:]port] [-E log_file] [-e escape_char]  
[-F configfile] [-I pkcs11] [-i identity_file] [-L address]  
[-l login_name] [-m mac_spec] [-O ctl_cmd] [-o option] [-p port]  
[-Q query_option] [-R address] [-S ctl_path] [-W host:port]  
[-w local_tun[:remote_tun]] [user@]hostname [command]  
dsvv@dsvv-HP-xw4600-Workstation:~$
```

- (v). **nslookup** : This command is used for identifying IP address of the hostname.

Syntax: \$ nslookup <hostname>

Output:

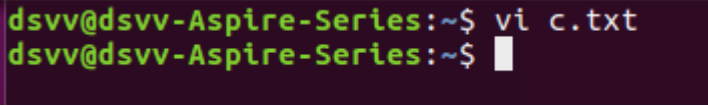
```
dsvv@dsvv-HP-xw4600-Workstation: ~  
dsvv@dsvv-HP-xw4600-Workstation:~$ nslookup  
> ^Z  
[1]+  Stopped                  nslookup  
dsvv@dsvv-HP-xw4600-Workstation:~$ nslookup google.com  
Server:          127.0.1.1  
Address:         127.0.1.1#53  
  
Non-authoritative answer:  
Name:   google.com  
Address: 172.217.31.14  
  
dsvv@dsvv-HP-xw4600-Workstation:~$
```

## Task #9: Commands for Vi editor

Syntax : vi <filename>

Example : vi c.txt

Output:



```
dsvv@dsvv-Aspire-Series:~$ vi c.txt
dsvv@dsvv-Aspire-Series:~$
```

To go into insert mode : i

To go into command mode : **Esc**

To exit from the vi editor : wq (in command mode)

[w is used for saving the file]

### Cursor movement commands :

move the cursor one character left : **h**

move the cursor one character right : **l**

move the cursor one line up : **k**

move the cursor one line down : **j**

**:n** (n=line no.) or **nG** : cursor goes to the specified line.

**\$** :- move the cursor to the end of the current line.

**o** :- It will move the cursor in beginning of the current line.

**w**:- It will move the cursor one word forward.

**b** :- it will move the cursor one word backward.

### Text deletion commands :

**x** : deletion of one character

**dw** : delete one word

**db** : delete one word backward

**yy** : copy content (current line)

**p** : paste downward

**P** : paste upward

**u** : undo

**U** : redo

**a** : append the text following the current cursor position

**A** : append the text to the end of the current line.

vi:

Output:

[illegible]

## Task #10 : List of environment variables in Linux/Unix

(i). **accessing variable values** : This command is used to see the value of a variable.

Syntax : `$ echo $variable_name`

Example : `$ echo $var`

(ii). **Create Variable** : This command is used for creating a variable.

Syntax : `$ variable_name=value`

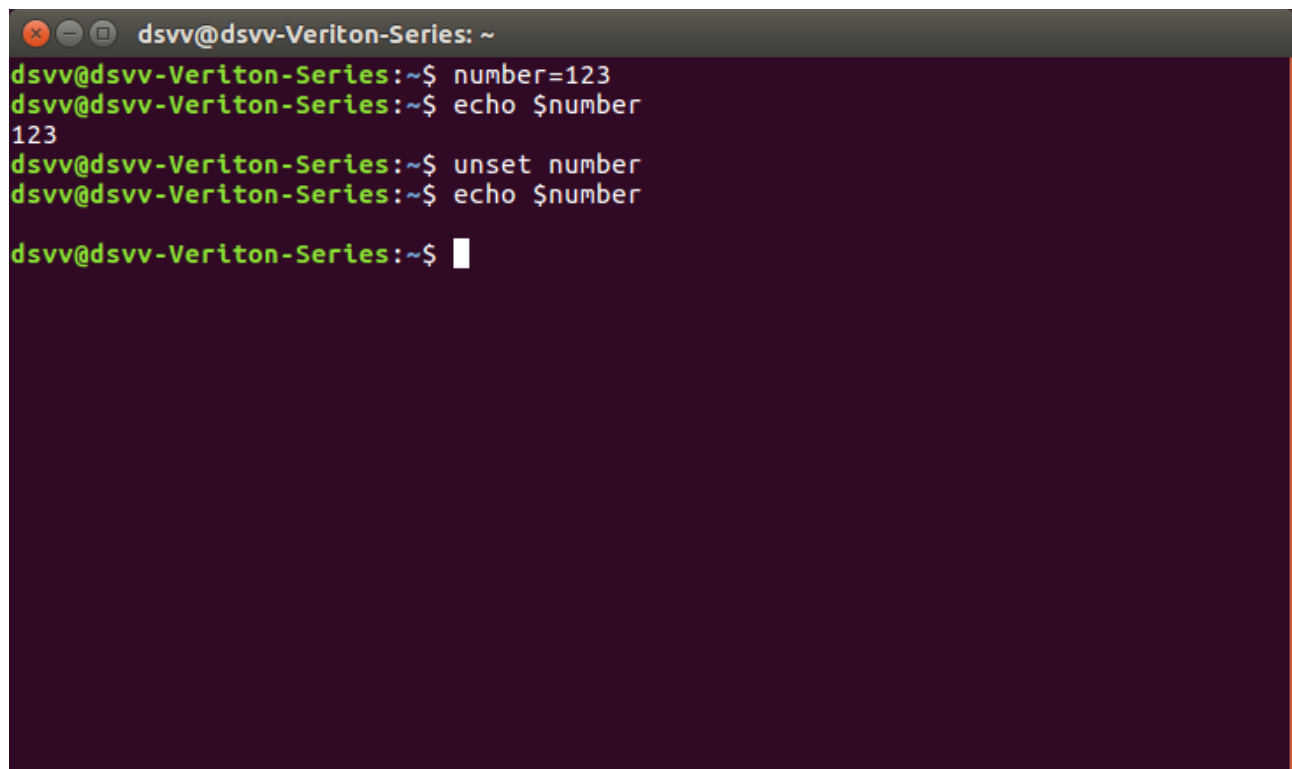
Example : `$ var = 123`

(iii). **Delete Variable** : This command is used for deleting a variable.

Syntax : `$ unset variable_name`

Example : `$ unset var`

Output:

A terminal window titled 'dsvv@dsvv-Veriton-Series: ~' with a dark purple background. It shows a sequence of commands and their outputs: 'number=123' is entered, followed by 'echo \$number' which outputs '123'. Then 'unset number' is entered, followed by 'echo \$number' which produces no output. The prompt returns to the shell.

```
dsvv@dsvv-Veriton-Series: ~$ number=123
dsvv@dsvv-Veriton-Series:~$ echo $number
123
dsvv@dsvv-Veriton-Series:~$ unset number
dsvv@dsvv-Veriton-Series:~$ echo $number
dsvv@dsvv-Veriton-Series:~$
```