Discipline

Basic & Advance UNIX command on LINUX OS

CS5003

Assignment 1

Sheikh Muhammed Tadeeb (AU19B1014)

❖ Problem Statement:

Read the Linux Administrative command manual and answer the following question.

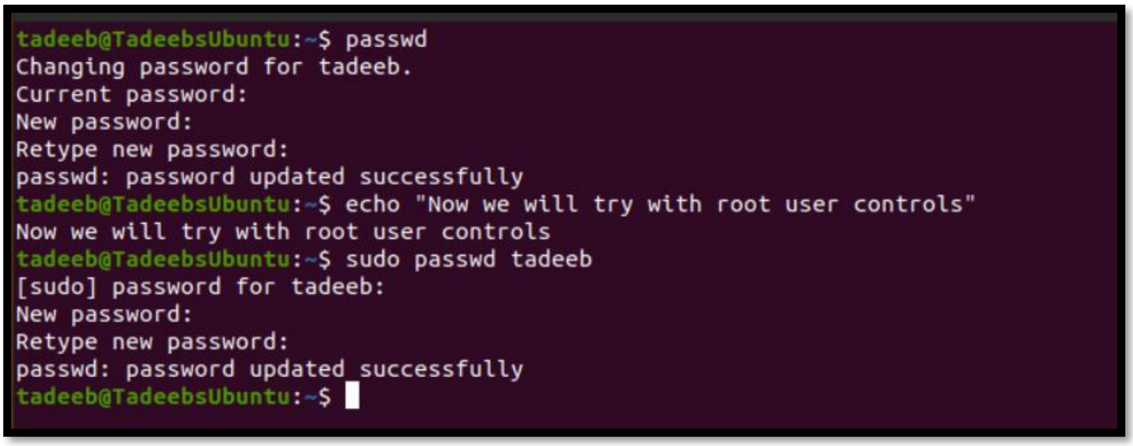
❖ Solution:

1) Change your password to a password you would like to use for the remainder of the semester.

- Brief:

Sno.	Command	Description
1	*passwd*	Prompted to enter your current password. If the password is correct, the command will ask us to enter and confirm the new password.
2	*sudo passwd <username>*	Only the root user and users with sudo access can change the password of another user account.

- Output:



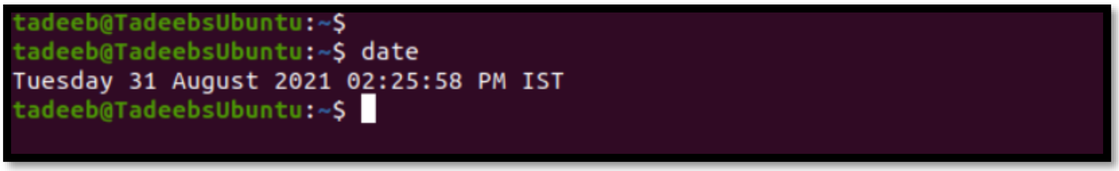
```
tadeeb@TadeebsUbuntu:~$ passwd
Changing password for tadeeb.
Current password:
New password:
Retype new password:
passwd: password updated successfully
tadeeb@TadeebsUbuntu:~$ echo "Now we will try with root user controls"
Now we will try with root user controls
tadeeb@TadeebsUbuntu:~$ sudo passwd tadeeb
[sudo] password for tadeeb:
New password:
Retype new password:
passwd: password updated successfully
tadeeb@TadeebsUbuntu:~$ 
```

2) Display the system's date.

- Brief:

Sno.	Command	Description
1	*date*	See current time and date:

- Output:



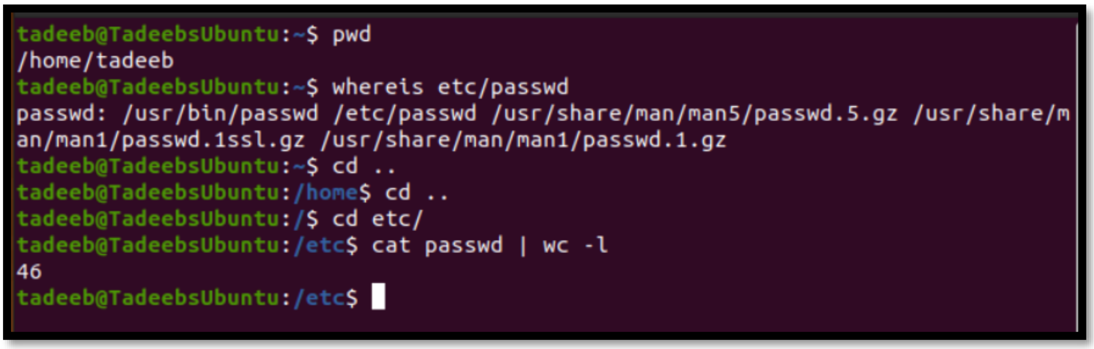
```
tadeeb@TadeebsUbuntu:~$
tadeeb@TadeebsUbuntu:~$ date
Tuesday 31 August 2021 02:25:58 PM IST
tadeeb@TadeebsUbuntu:~$ 
```

3) Count the number of lines in the /etc/passwd file.

- Brief:

Sno.	Command	Description
1	*cat*	cat command reads data from the file and gives their content as output. It helps us to create, view, concatenate files.
2	*wc*	wc stands for word count. As the name implies, it is mainly used for counting purpose. It is used to find out number of lines, word count, byte and characters count in the files specified in the file arguments. Syntax: *wc [option] [File]* *-l* This option prints the number of lines present in a file.
3	*cat /etc/passwd* | *wc -l*	We can directly do this from our pwd by using pipe function (used for making output of one command as input to another).

- Output:



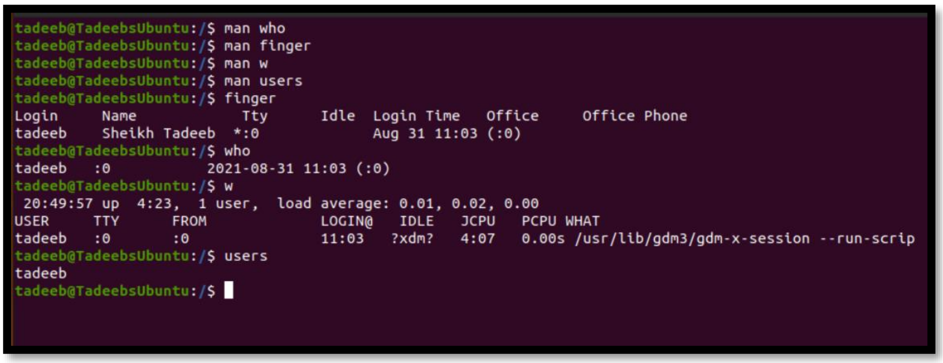
```
tadeeb@TadeebsUbuntu:~$ pwd
/home/tadeeb
tadeeb@TadeebsUbuntu:~$ whereis etc/passwd
passwd: /usr/bin/passwd /etc/passwd /usr/share/man/man5/passwd.5.gz /usr/share/m
an/man1/passwd.1ssl.gz /usr/share/man/man1/passwd.1.gz
tadeeb@TadeebsUbuntu:~$ cd ..
tadeeb@TadeebsUbuntu:/home$ cd ..
tadeeb@TadeebsUbuntu:/$ cd etc/
tadeeb@TadeebsUbuntu:/etc$ cat passwd | wc -l
46
tadeeb@TadeebsUbuntu:/etc$ 
```

4) Find out who else is on the system.

- Brief:

Sno.	Command	Description
1	*who*	Prints information about users who are currently logged in.
2	*finger*	It displays information about the system users.
3	*w*	It shows who is logged on & what they are doing.
4	*users*	Print the username of users currently logged-in.

- Output:



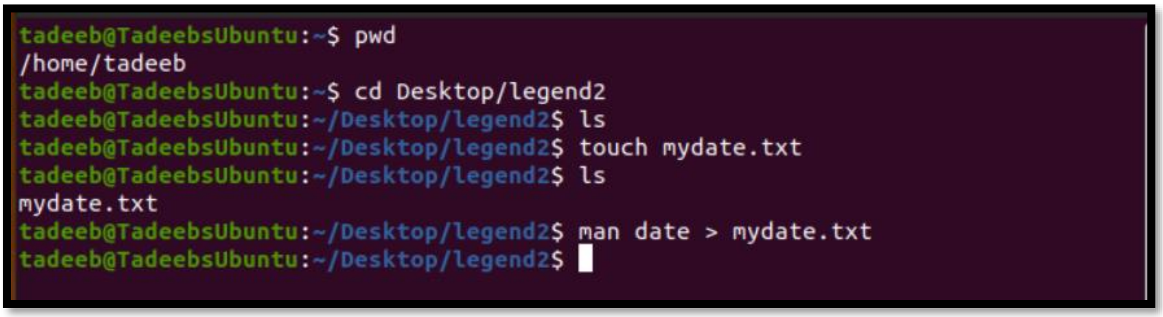
```
tadeeb@TadeebsUbuntu:/$ man who
tadeeb@TadeebsUbuntu:/$ man finger
tadeeb@TadeebsUbuntu:/$ man w
tadeeb@TadeebsUbuntu:/$ man users
tadeeb@TadeebsUbuntu:/$ finger
Login     Name            Tty      Idle  Login Time   Office     Office Phone
tadeeb    Sheikh Tadeeb  *:0             Aug 31 11:03 (:0)
tadeeb@TadeebsUbuntu:/$ who
tadeeb   :0           2021-08-31 11:03 (:0)
tadeeb@TadeebsUbuntu:/$ w
 20:49:57 up  4:23,  1 user,  load average: 0.01, 0.02, 0.00
USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
tadeeb   :0       :0               11:03   ?xdm?   4:07   0.00s /usr/lib/gdm3/gdm-x-session --run-scrip
tadeeb@TadeebsUbuntu:/$ users
tadeeb
tadeeb@TadeebsUbuntu:/$ 
```

5) Direct the output of the man pages for the date command to a file named mydate.

- Brief:

Sno.	Command	Description
1	*touch*	The touch command is a standard command used in UNIX/Linux operating system which is used to create, change and modify timestamps of a file.
2	>	It is used to redirect the data of one file to another.

- Output:



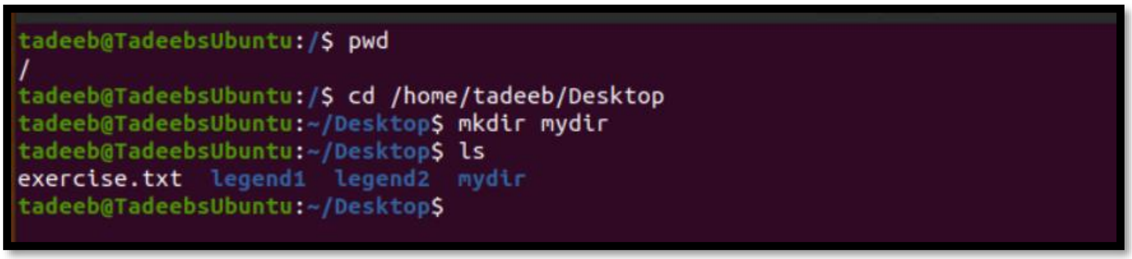
```
tadeeb@TadeebsUbuntu:~$ pwd
/home/tadeeb
tadeeb@TadeebsUbuntu:~$ cd Desktop/legend2
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ ls
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ touch mydate.txt
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ ls
mydate.txt
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ man date > mydate.txt
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ 
```

6) Create a subdirectory called mydir.

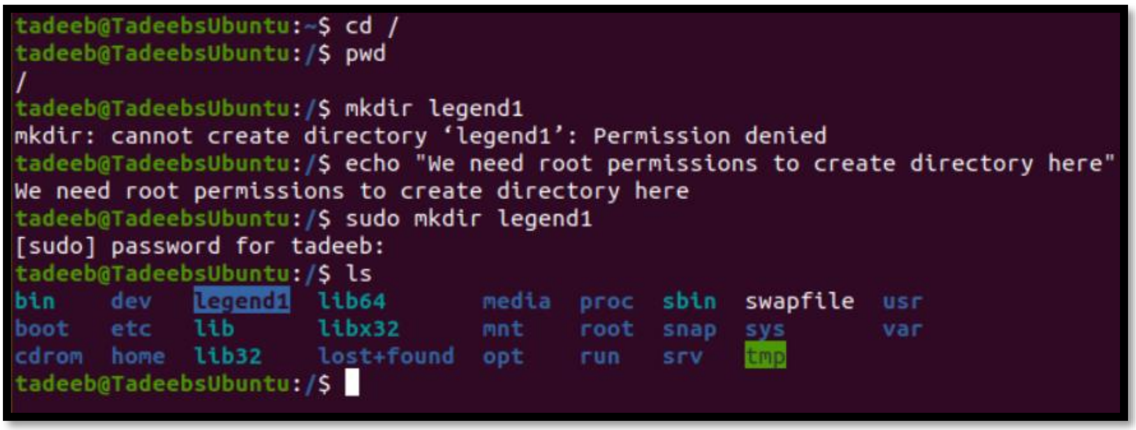
- Brief:

Sno.	Command	Description
1	*mkdir*	*mkdir* command in Linux allows the user to create directories (also referred to as folders in some operating systems). This command can create multiple directories at once as well as set the permissions for the directories. It is important to note that the user executing this command must have enough permissions to create a directory in the parent directory, or he/she may receive a 'permission denied' error.
2	*sudo*	We can use *sudo* if we try to create sub-directory in root directory.

- Output:



```
tadeeb@TadeebsUbuntu:/$ pwd
/
tadeeb@TadeebsUbuntu:/$ cd /home/tadeeb/Desktop
tadeeb@TadeebsUbuntu:~/Desktop$ mkdir mydir
tadeeb@TadeebsUbuntu:~/Desktop$ ls
exercise.txt  legend1  legend2  mydir
tadeeb@TadeebsUbuntu:~/Desktop$
```



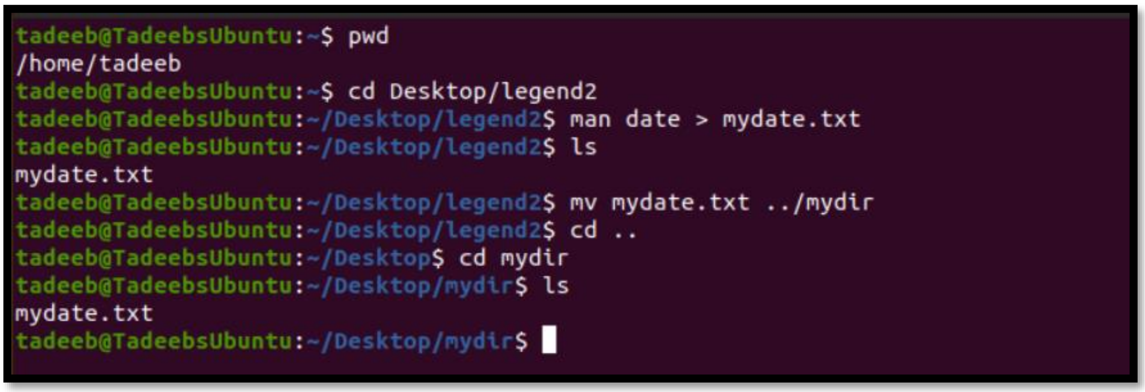
```
tadeeb@TadeebsUbuntu:~$ cd /
tadeeb@TadeebsUbuntu:/$ pwd
/
tadeeb@TadeebsUbuntu:/$ mkdir legend1
mkdir: cannot create directory ‘legend1’: Permission denied
tadeeb@TadeebsUbuntu:/$ echo "We need root permissions to create directory here"
We need root permissions to create directory here
tadeeb@TadeebsUbuntu:/$ sudo mkdir legend1
[sudo] password for tadeeb:
tadeeb@TadeebsUbuntu:/$ ls
bin     dev     legend1  lib64       media  proc  sbin  swapfile  usr
boot    etc     lib      libx32      mnt    root  snap  sys       var
cdrom   home    lib32    lost+found  opt    run   srv   tmp
tadeeb@TadeebsUbuntu:/$
```

7) Move the file mydate into the new subdirectory.

- Brief:

Sno.	Command	Description
1	*mv*	It could be used to rename a file, move a file or can do both.

- Output:



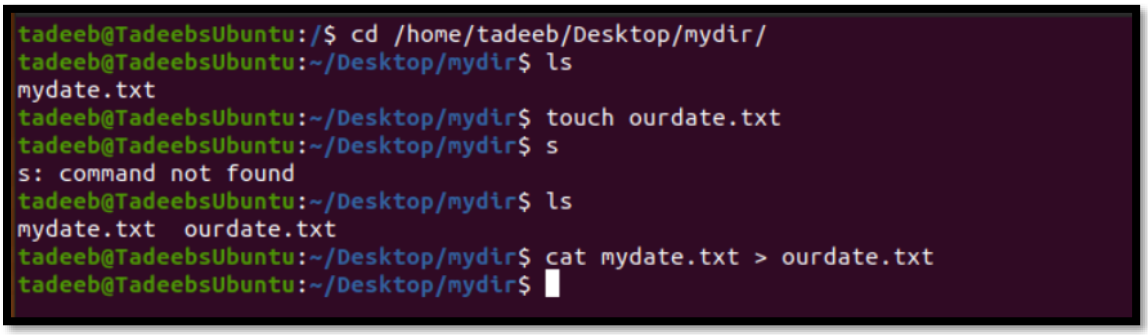
```
tadeeb@TadeebsUbuntu:~$ pwd
/home/tadeeb
tadeeb@TadeebsUbuntu:~$ cd Desktop/legend2
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ man date > mydate.txt
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ ls
mydate.txt
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ mv mydate.txt ../mydir
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ cd ..
tadeeb@TadeebsUbuntu:~/Desktop$ cd mydir
tadeeb@TadeebsUbuntu:~/Desktop/mydir$ ls
mydate.txt
tadeeb@TadeebsUbuntu:~/Desktop/mydir$
```

8) Go to the subdirectory mydir and copy the file mydate to a new file called ourdate.

- Brief:

Sno.	Command	Description
1	*cat*	Copy the contents of one file to another file.
2	>	It is used to redirect the data of one file to another.

- Output:



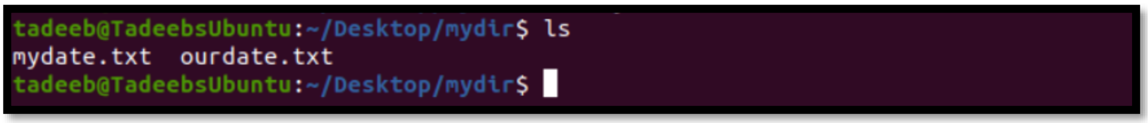
```
tadeeb@TadeebsUbuntu:/$ cd /home/tadeeb/Desktop/mydir/
tadeeb@TadeebsUbuntu:~/Desktop/mydir$ ls
mydate.txt
tadeeb@TadeebsUbuntu:~/Desktop/mydir$ touch ourdate.txt
tadeeb@TadeebsUbuntu:~/Desktop/mydir$ s
s: command not found
tadeeb@TadeebsUbuntu:~/Desktop/mydir$ ls
mydate.txt  ourdate.txt
tadeeb@TadeebsUbuntu:~/Desktop/mydir$ cat mydate.txt > ourdate.txt
tadeeb@TadeebsUbuntu:~/Desktop/mydir$ 
```

9) List the contents of mydir.

- Brief:

Sno.	Command	Description
1	*ls*	The Linux ls command allows you to view a list of the files and folders in a given directory.

- Output:



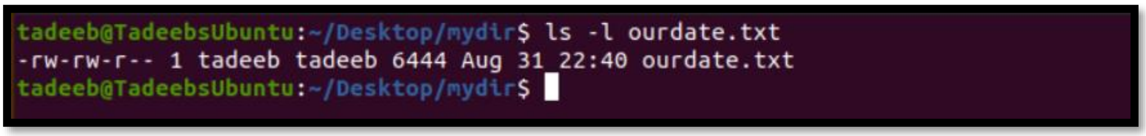
```
tadeeb@TadeebsUbuntu:~/Desktop/mydir$ ls
mydate.txt  ourdate.txt
tadeeb@TadeebsUbuntu:~/Desktop/mydir$ 
```

10) Do a long listing on the file ourdate and note the permissions.

- Brief:

Sno.	Command	Description
1	*ls -l*	Does long listing on specified file.

- Output:



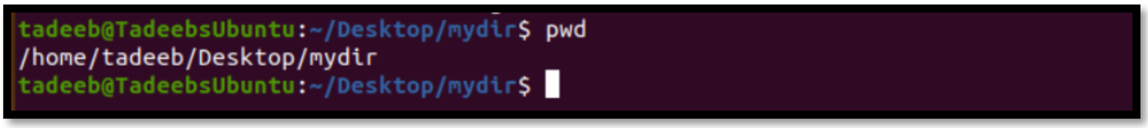
```
tadeeb@TadeebsUbuntu:~/Desktop/mydir$ ls -l ourdate.txt
-rw-rw-r-- 1 tadeeb tadeeb 6444 Aug 31 22:40 ourdate.txt
tadeeb@TadeebsUbuntu:~/Desktop/mydir$ 
```

11) Display the name of the current directory starting from the root.

- Brief:

Sno.	Command	Description
1	*pwd*	*pwd* stands for Print Working Directory. It prints the path of the working directory, starting from the root.

- Output:



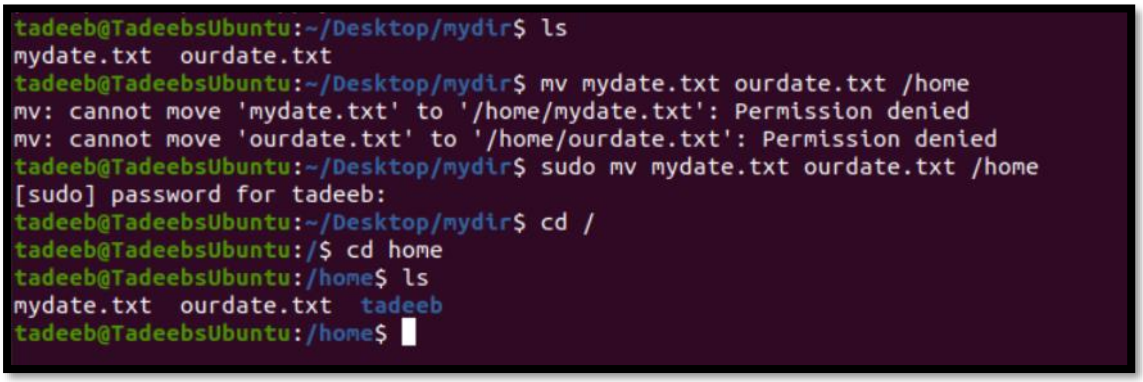
```
tadeeb@TadeebsUbuntu:~/Desktop/mydir$ pwd
/home/tadeeb/Desktop/mydir
tadeeb@TadeebsUbuntu:~/Desktop/mydir$ 
```

12) Move the files in the directory mydir back to your home directory..

- Brief:

Sno.	Command	Description
1	*mv*	It could be used to rename a file, move a file or can do both.
2	*sudo*	We use *sudo* to give root like permissions to some other user.

- Output:



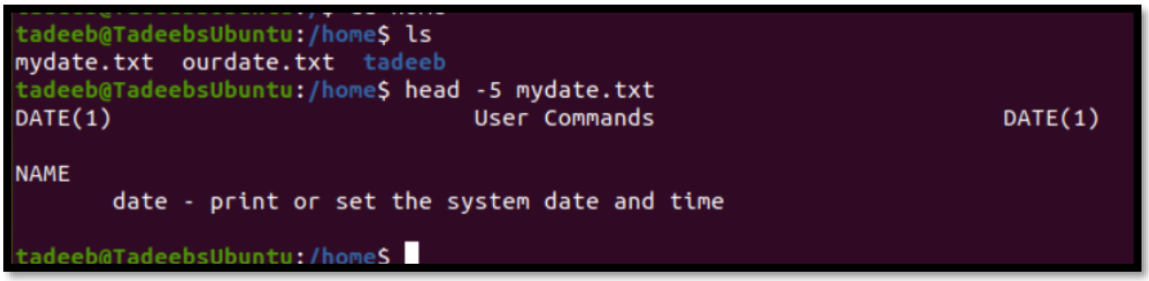
```
tadeeb@TadeebsUbuntu:~/Desktop/mydir$ ls
mydate.txt  ourdate.txt
tadeeb@TadeebsUbuntu:~/Desktop/mydir$ mv mydate.txt ourdate.txt /home
mv: cannot move 'mydate.txt' to '/home/mydate.txt': Permission denied
mv: cannot move 'ourdate.txt' to '/home/ourdate.txt': Permission denied
tadeeb@TadeebsUbuntu:~/Desktop/mydir$ sudo mv mydate.txt ourdate.txt /home
[sudo] password for tadeeb:
tadeeb@TadeebsUbuntu:~/Desktop/mydir$ cd /
tadeeb@TadeebsUbuntu:/$ cd home
tadeeb@TadeebsUbuntu:/home$ ls
mydate.txt  ourdate.txt  tadeeb
tadeeb@TadeebsUbuntu:/home$ 
```

13) Display the first 5 lines of mydate.

- Brief:

Sno.	Command	Description
1	*head*	It is used to print first 10 lines of file by default. With options we can specify the number of lines.

- Output:



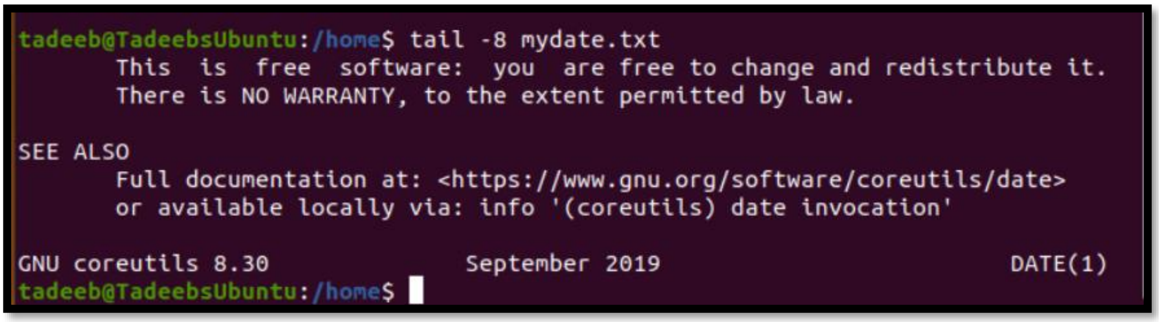
```
tadeeb@TadeebsUbuntu:/home$ ls
mydate.txt  ourdate.txt  tadeeb
tadeeb@TadeebsUbuntu:/home$ head -5 mydate.txt
DATE(1)                          User Commands                          DATE(1)

NAME
       date - print or set the system date and time

tadeeb@TadeebsUbuntu:/home$ 
```

14) Display the last 8 lines of mydate.

- Brief:

Sno.	Command	Description
1	*tail*	It is used to print last 10 lines of file by default. With options we can specify the number of lines.

- Output:



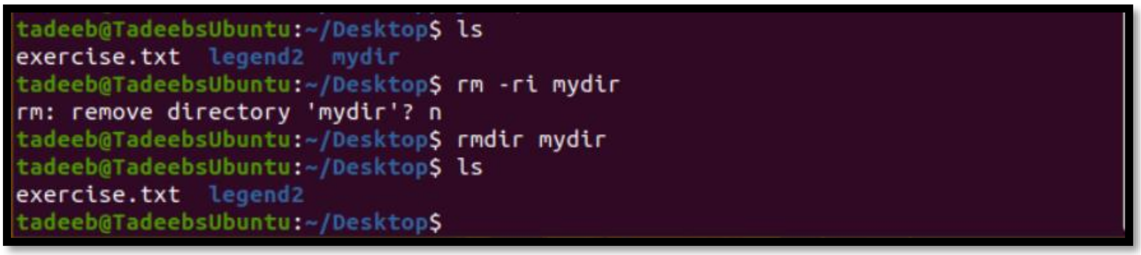
```
tadeeb@TadeebsUbuntu:/home$ tail -8 mydate.txt
       This  is  free  software:  you  are free to change and redistribute it.
       There is NO WARRANTY, to the extent permitted by law.

SEE ALSO
       Full documentation at: <https://www.gnu.org/software/coreutils/date>
       or available locally via: info '(coreutils) date invocation'

GNU coreutils 8.30              September 2019                          DATE(1)
tadeeb@TadeebsUbuntu:/home$
```

15) Remove the directory mydir.

- Brief:

Sno.	Command	Description
1	*rmdir*	It is used to remove a directory if it is empty.
2	*rm -rf*	It is used to remove a directory forcefully.
3	*rm-ri*	It is used to remove a directory interactively.

- Output:



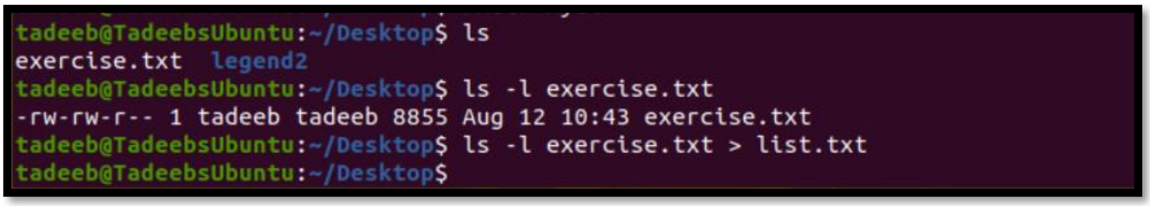
```
tadeeb@TadeebsUbuntu:~/Desktop$ ls
exercise.txt  legend2  mydir
tadeeb@TadeebsUbuntu:~/Desktop$ rm -ri mydir
rm: remove directory 'mydir'? n
tadeeb@TadeebsUbuntu:~/Desktop$ rmdir mydir
tadeeb@TadeebsUbuntu:~/Desktop$ ls
exercise.txt  legend2
tadeeb@TadeebsUbuntu:~/Desktop$
```

16) Redirect the output of the long listing of files to a file named list.

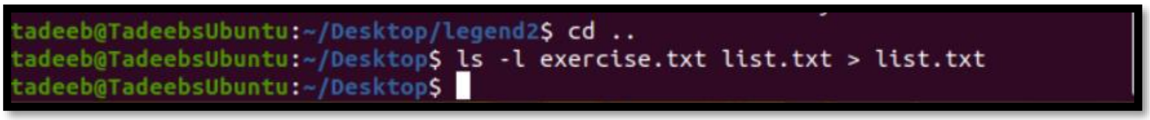
- Brief:

Sno.	Command	Description
1	*ls -l*	Does long-listing on a specified file.
2	>	It is used to redirect the data of one file to another.

- Output:



```
tadeeb@TadeebsUbuntu:~/Desktop$ ls
exercise.txt  legend2
tadeeb@TadeebsUbuntu:~/Desktop$ ls -l exercise.txt
-rw-rw-r-- 1 tadeeb tadeeb 8855 Aug 12 10:43 exercise.txt
tadeeb@TadeebsUbuntu:~/Desktop$ ls -l exercise.txt > list.txt
tadeeb@TadeebsUbuntu:~/Desktop$
```



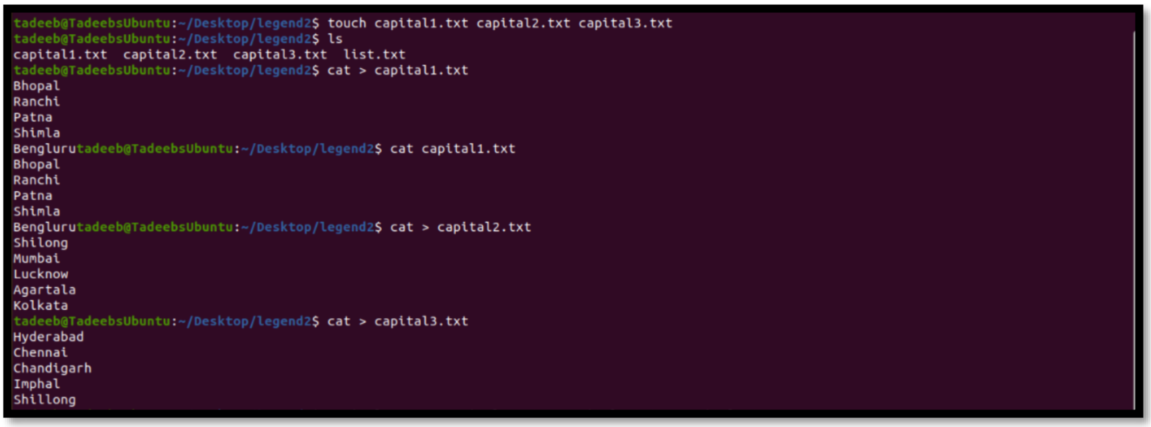
```
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ cd ..
tadeeb@TadeebsUbuntu:~/Desktop$ ls -l exercise.txt list.txt > list.txt
tadeeb@TadeebsUbuntu:~/Desktop$
```

17) Select any 5 capitals of states in India and enter them in a file named capitals1. Choose 5 more capitals and enter them in a file named capitals2. Choose 5 more capitals and enter them in a file named capitals3. Concatenate all 3 files and redirect the output to a file named capitals.

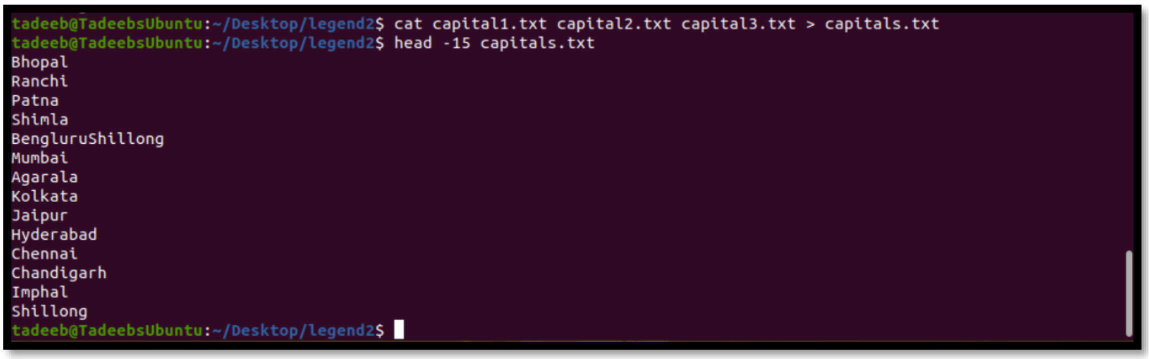
- Brief:

Sno.	Command	Description
1	*cat*	*cat* command reads data from the file and gives their content as output. It helps us to create, view, concatenate files.
2	*touch*	The touch command is a standard command used in UNIX/Linux operating system which is used to create, change and modify timestamps of a file.

- Output:



```
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ touch capital1.txt capital2.txt capital3.txt
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ ls
capital1.txt  capital2.txt  capital3.txt  list.txt
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ cat > capital1.txt
Bhopal
Ranchi
Patna
Shimla
Benglurutadeeb@TadeebsUbuntu:~/Desktop/legend2$ cat capital1.txt
Bhopal
Ranchi
Patna
Shimla
Benglurutadeeb@TadeebsUbuntu:~/Desktop/legend2$ cat > capital2.txt
Shilong
Mumbai
Lucknow
Agartala
Kolkata
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ cat > capital3.txt
Hyderabad
Chennai
Chandigarh
Imphal
Shillong
```



```
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ cat capital1.txt capital2.txt capital3.txt > capitals.txt
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ head -15 capitals.txt
Bhopal
Ranchi
Patna
Shimla
BengluruShillong
Mumbai
Agarala
Kolkata
Jaipur
Hyderabad
Chennai
Chandigarh
Imphal
Shillong
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ 
```

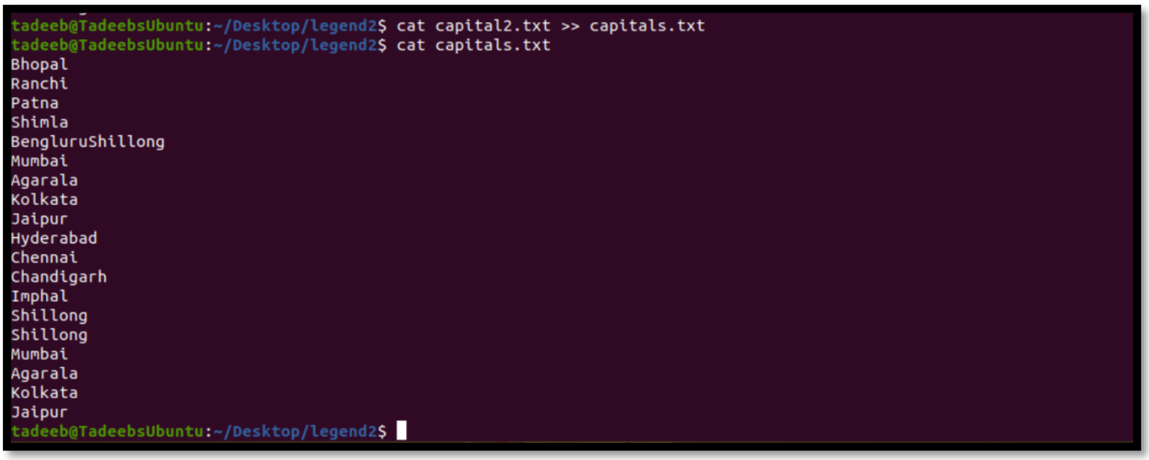
18) Concatenate the file capitals2 at the end of file capitals.

- Brief:

Sno.	Command	Description
1	*cat*	*cat* command reads data from the file and gives their content as output. It helps us to create, view, concatenate files.

2	*cat >>*	Cat command to write in an already existing file.

- Output:



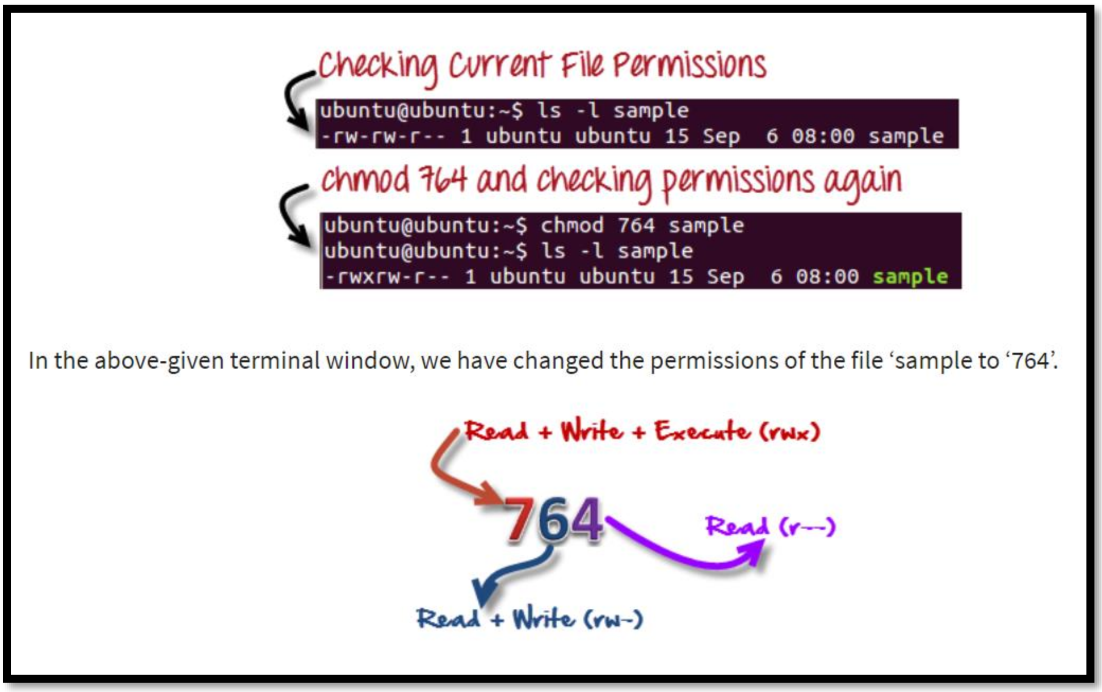
```
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ cat capital2.txt >> capitals.txt
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ cat capitals.txt
Bhopal
Ranchi
Patna
Shimla
BengluruShillong
Mumbai
Agarala
Kolkata
Jaipur
Hyderabad
Chennai
Chandigarh
Imphal
Shillong
Shillong
Mumbai
Agarala
Kolkata
Jaipur
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ 
```

19) Give a read and write permissions to all users for the file capitals.

- Brief:

Sno.	Command	Description
1	*chmod*	We can use the *'chmod'* command which stands for 'change mode'. Using the command, we can set permissions (read, write, execute) on a file/directory for the owner, group and the world. Syntax: `chmod permissions filename` There are 2 ways to use the command – 1) Absolute mode: File permissions are represented as three-digit octal number. 2) Symbolic mode:

- Absolute Mode Example:



Checking Current File Permissions

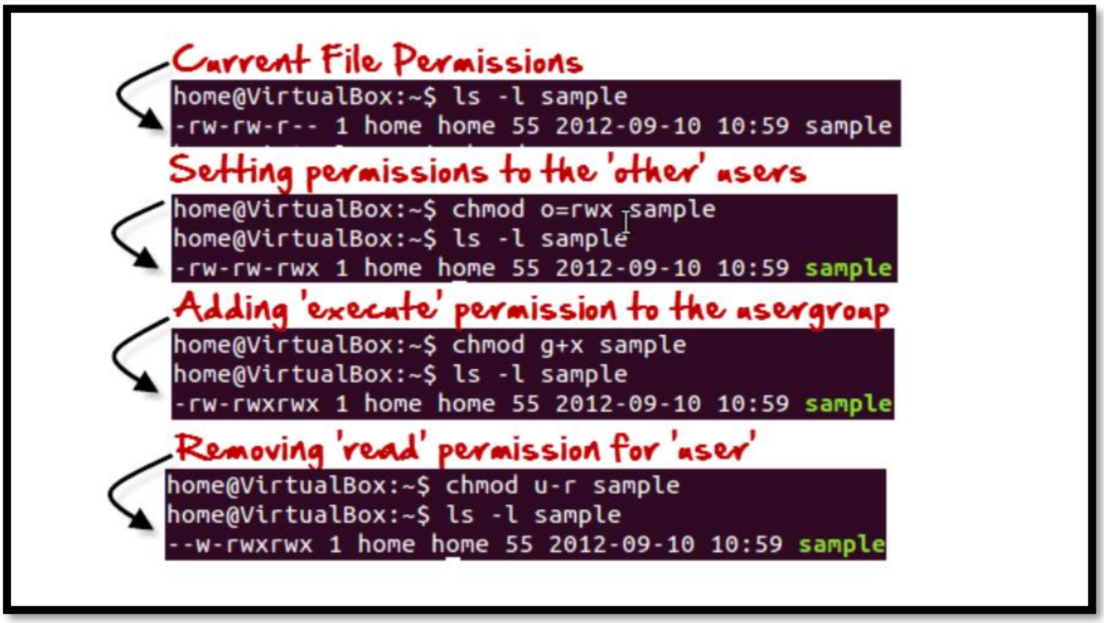
```
ubuntu@ubuntu:~$ ls -l sample
-rw-rw-r-- 1 ubuntu ubuntu 15 Sep  6 08:00 sample
```

chmod 764 and checking permissions again

```
ubuntu@ubuntu:~$ chmod 764 sample
ubuntu@ubuntu:~$ ls -l sample
-rwxrw-r-- 1 ubuntu ubuntu 15 Sep  6 08:00 sample
```

In the above-given terminal window, we have changed the permissions of the file ‘sample to ‘764’.

- Symbolic Mode Example:



Current File Permissions

```
home@VirtualBox:~$ ls -l sample
-rw-rw-r-- 1 home home 55 2012-09-10 10:59 sample
```

Setting permissions to the 'other' users

```
home@VirtualBox:~$ chmod o=rwx sample
home@VirtualBox:~$ ls -l sample
-rw-rw-rwx 1 home home 55 2012-09-10 10:59 sample
```

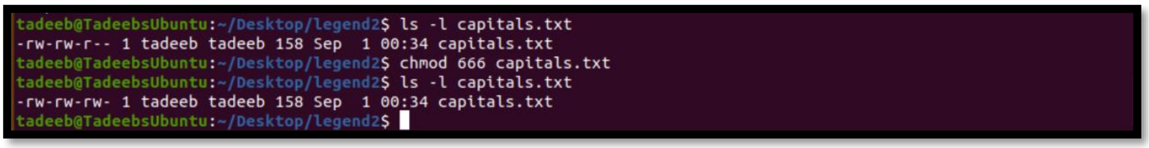
Adding 'execute' permission to the usergroup

```
home@VirtualBox:~$ chmod g+x sample
home@VirtualBox:~$ ls -l sample
-rw-rwxrwx 1 home home 55 2012-09-10 10:59 sample
```

Removing 'read' permission for 'user'

```
home@VirtualBox:~$ chmod u-r sample
home@VirtualBox:~$ ls -l sample
--w-rwxrwx 1 home home 55 2012-09-10 10:59 sample
```

- Output:



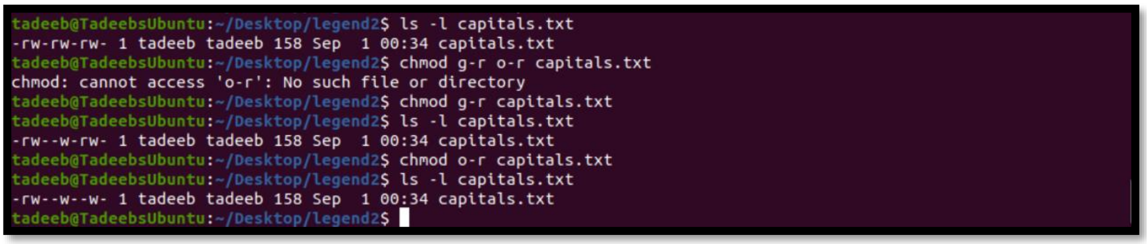
```
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ ls -l capitals.txt
-rw-rw-r-- 1 tadeeb tadeeb 158 Sep  1 00:34 capitals.txt
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ chmod 666 capitals.txt
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ ls -l capitals.txt
-rw-rw-rw- 1 tadeeb tadeeb 158 Sep  1 00:34 capitals.txt
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ 
```

20) Give read permissions only to the owner of the file capitals. Open the file, make some changes, and try to save it. What happens?

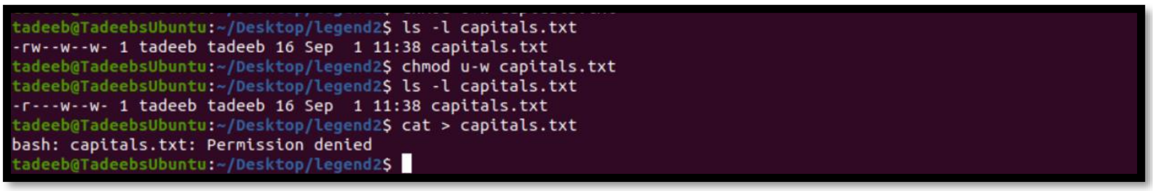
- Brief:

Sno.	Command	Description
1	*chmod*	We can use the *‘chmod’* command which stands for ‘change mode’. Using the command, we can set permissions (read, write, execute) on a file/directory for the owner, group and the world. Syntax: `chmod permissions filename` There are 2 ways to use the command – 3) Absolute mode: File permissions are represented as three-digit octal number. 4) Symbolic mode:

- Output:



```
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ ls -l capitals.txt
-rw-rw-rw- 1 tadeeb tadeeb 158 Sep  1 00:34 capitals.txt
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ chmod g-r o-r capitals.txt
chmod: cannot access 'o-r': No such file or directory
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ chmod g-r capitals.txt
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ ls -l capitals.txt
-rw--w-rw- 1 tadeeb tadeeb 158 Sep  1 00:34 capitals.txt
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ chmod o-r capitals.txt
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ ls -l capitals.txt
-rw--w--w- 1 tadeeb tadeeb 158 Sep  1 00:34 capitals.txt
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ 
```



```
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ ls -l capitals.txt
-rw--w--w- 1 tadeeb tadeeb 16 Sep  1 11:38 capitals.txt
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ chmod u-w capitals.txt
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ ls -l capitals.txt
-r---w--w- 1 tadeeb tadeeb 16 Sep  1 11:38 capitals.txt
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ cat > capitals.txt
bash: capitals.txt: Permission denied
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ 
```

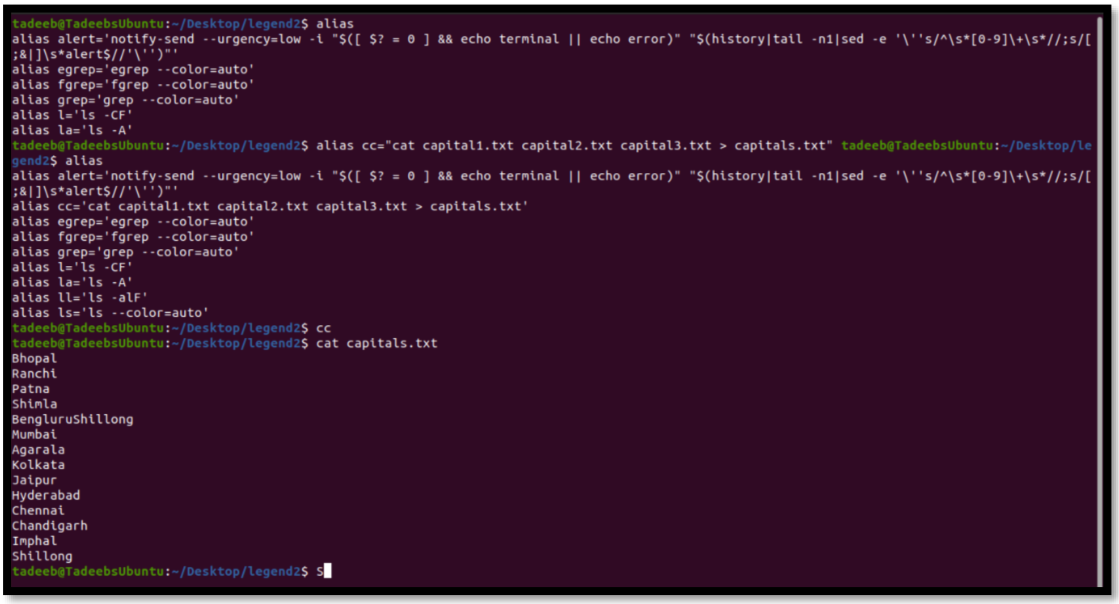
It doesn’t allow us to make any changes as we have given only read permission to the user.

21) Create an alias to concatenate the 3 files capitals1, capitals2, capitals3 and redirect the output to a file named capitals. Activate the alias and make it run.

- Brief:

Sno.	Command	Description
1	*cat*	*cat* command reads data from the file and gives their content as output. It helps us to create, view, concatenate files.
2	*alias*	*alias* command instructs the shell to replace one string with another string while executing the commands. When we often have to use a single big command multiple times, in those cases, we create something called as alias for that command. Alias is like a shortcut command which will have same functionality as if we are writing the whole command. We can save alias in two ways: 1) Temporary (saved for that session) 2) Permanent (saved in configuration file)

- Output:



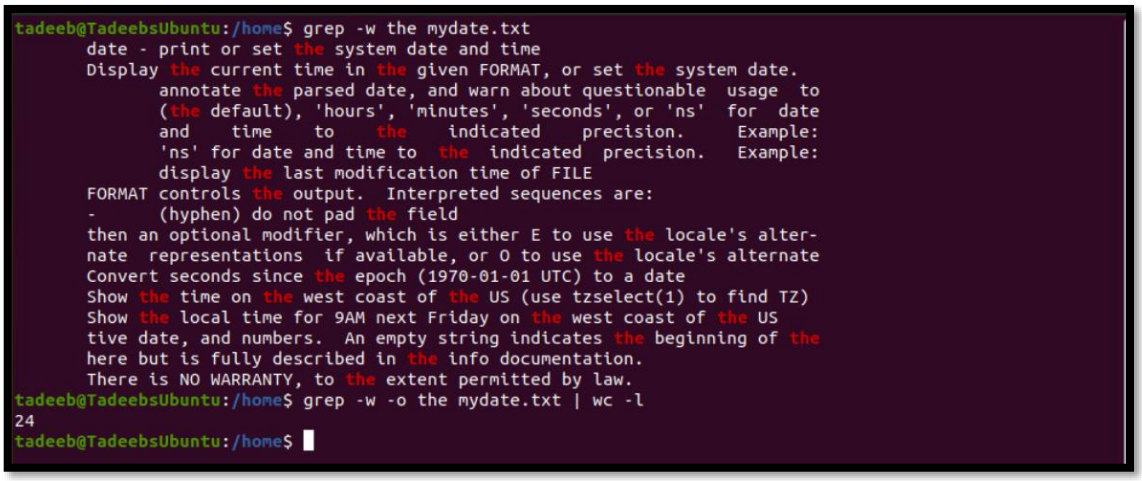
```
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ alias
alias alert='notify-send --urgency=low -i "$([ $? = 0 ] && echo terminal || echo error)" "$(history|tail -n1|sed -e '\''s/^\s*[0-9]\+\s*//;s/[
;&|]\s*alert$//'\'')"'
alias egrep='egrep --color=auto'
alias fgrep='fgrep --color=auto'
alias grep='grep --color=auto'
alias l='ls -CF'
alias la='ls -A'
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ alias cc="cat capital1.txt capital2.txt capital3.txt > capitals.txt" tadeeb@TadeebsUbuntu:~/Desktop/le
gend2$ alias
alias alert='notify-send --urgency=low -i "$([ $? = 0 ] && echo terminal || echo error)" "$(history|tail -n1|sed -e '\''s/^\s*[0-9]\+\s*//;s/[
;&|]\s*alert$//'\'')"'
alias cc='cat capital1.txt capital2.txt capital3.txt > capitals.txt'
alias egrep='egrep --color=auto'
alias fgrep='fgrep --color=auto'
alias grep='grep --color=auto'
alias l='ls -CF'
alias la='ls -A'
alias ll='ls -alF'
alias ls='ls --color=auto'
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ cc
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ cat capitals.txt
Bhopal
Ranchi
Patna
Shimla
BengluruShillong
Mumbai
Agarala
Kolkata
Jaipur
Hyderabad
Chennai
Chandigarh
Imphal
Shillong
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ S
```

22) Find out the number of times the string “the” appears in the file mydate.

- Brief:

Sno.	Command	Description
1	*grep*	The grep filter searches a file for a particular pattern of characters, and displays all lines that contain that pattern. The pattern that is searched in the file is referred to as the regular expression (grep stands for globally search for regular expression and print out). -w By default, grep matches the given string/pattern even if it found as a substring in a file. The -w option to grep makes it match only the whole words and give line numbers where that word is present. -o By default, grep displays the entire line which has the matched string. We can make the grep to display only the matched string by using the -o option. It counts from substrings too.

- Output:



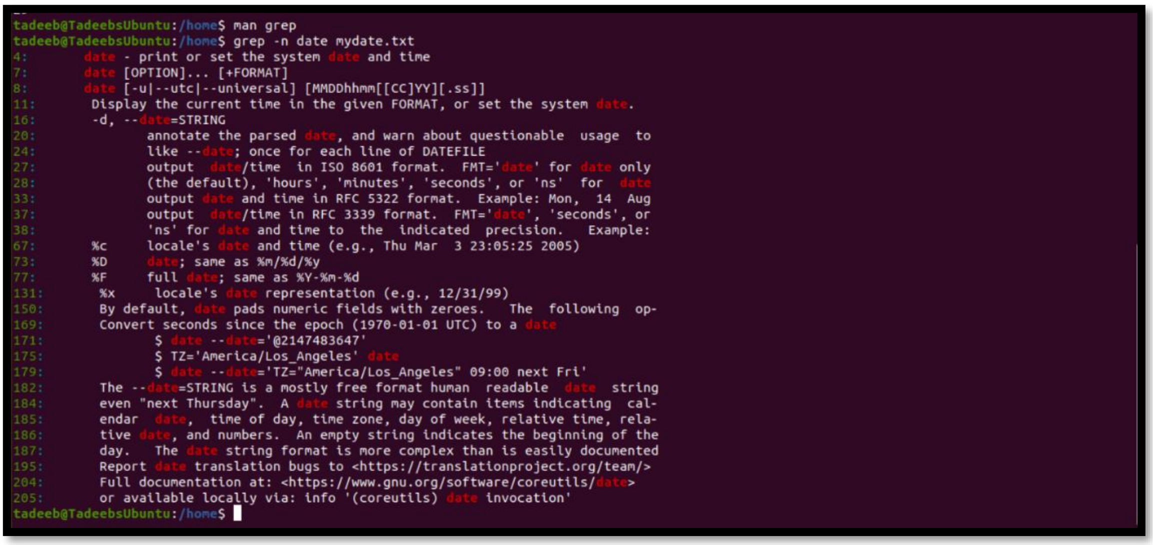
```
tadeeb@TadeebsUbuntu:/home$ grep -w the mydate.txt
       date - print or set the system date and time
       Display the current time in the given FORMAT, or set the system date.
              annotate the parsed date, and warn about questionable  usage  to
              (the default), 'hours', 'minutes', 'seconds', or 'ns'  for  date
              and    time    to    the    indicated    precision.    Example:
              'ns' for date and time to  the  indicated  precision.   Example:
              display the last modification time of FILE
       FORMAT controls the output.  Interpreted sequences are:
       -      (hyphen) do not pad the field
       then an optional modifier, which is either E to use the locale's alter-
       nate  representations  if available, or O to use the locale's alternate
       Convert seconds since the epoch (1970-01-01 UTC) to a date
       Show the time on the west coast of the US (use tzselect(1) to find TZ)
       Show the local time for 9AM next Friday on the west coast of the US
       tive date, and numbers.  An empty string indicates the beginning of the
       here but is fully described in the info documentation.
       There is NO WARRANTY, to the extent permitted by law.
tadeeb@TadeebsUbuntu:/home$ grep -w -o the mydate.txt | wc -l
24
tadeeb@TadeebsUbuntu:/home$ 
```

23) Find out the line numbers on which the string “date” exists in mydate.

- Brief:

Sno.	Command	Description
1	*grep*	It prints the line that matches patterns. *-n (line number)* Display the matched lines and their line numbers.

- Output:



```
tadeeb@TadeebsUbuntu:/home$ man grep
tadeeb@TadeebsUbuntu:/home$ grep -n date mydate.txt
4:       date - print or set the system date and time
7:       date [OPTION]... [+FORMAT]
8:       date [-u|--utc|--universal] [MMDDhhmm[[CC]YY][.ss]]
11:       Display the current time in the given FORMAT, or set the system date.
16:       -d, --date=STRING
20:              annotate the parsed date, and warn about questionable  usage  to
24:              like --date; once for each line of DATEFILE
27:              output  date/time  in ISO 8601 format.  FMT='date' for date only
28:              (the default), 'hours', 'minutes', 'seconds', or 'ns'  for  date
33:              output date and time in RFC 5322 format.  Example: Mon,  14  Aug
37:              output  date/time in RFC 3339 format.  FMT='date', 'seconds', or
38:              'ns' for date and time to  the  indicated  precision.   Example:
67:       %c     locale's date and time (e.g., Thu Mar  3 23:05:25 2005)
73:       %D     date; same as %m/%d/%y
77:       %F     full date; same as %Y-%m-%d
131:       %x     locale's date representation (e.g., 12/31/99)
150:       By default, date pads numeric fields with zeroes.   The  following  op-
169:       Convert seconds since the epoch (1970-01-01 UTC) to a date
171:              $ date --date='@2147483647'
175:              $ TZ='America/Los_Angeles' date
179:              $ date --date='TZ="America/Los_Angeles" 09:00 next Fri'
182:       The --date=STRING is a mostly free format human  readable  date  string
184:       even "next Thursday".  A date string may contain items indicating  cal-
185:       endar  date,  time of day, time zone, day of week, relative time, rela-
186:       tive date, and numbers.  An empty string indicates the beginning of the
187:       day.   The date string format is more complex than is easily documented
195:       Report date translation bugs to <https://translationproject.org/team/>
204:       Full documentation at: <https://www.gnu.org/software/coreutils/date>
205:       or available locally via: info '(coreutils) date invocation'
tadeeb@TadeebsUbuntu:/home$ 
```

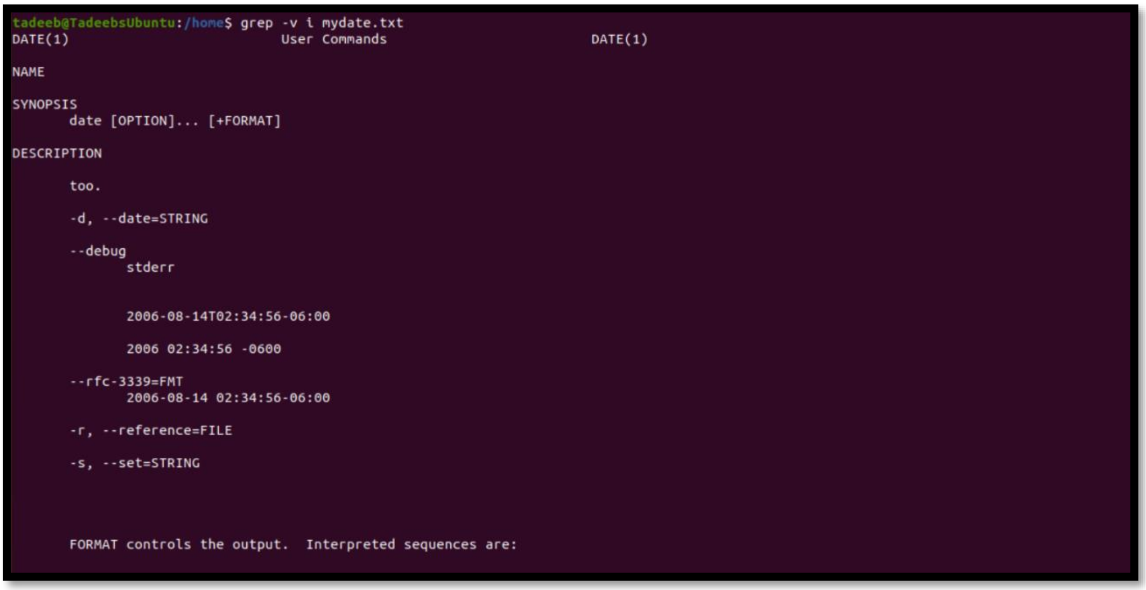
24) Print all lines of mydate except those that have the letter “i” in them.

- Brief:

Sno.	Command	Description
1	*grep*	It prints the line that matches patterns.

		-v (Inverting the pattern match) You can display the lines that are not matched with the specified search sting pattern using the -v option.

- Output:



```
tadeeb@TadeebsUbuntu:/home$ grep -v i mydate.txt
DATE(1)                          User Commands                         DATE(1)

NAME

SYNOPSIS
       date [OPTION]... [+FORMAT]

DESCRIPTION

       too.

       -d, --date=STRING

       --debug
              stderr


              2006-08-14T02:34:56-06:00

              2006 02:34:56 -0600

       --rfc-3339=FMT
              2006-08-14 02:34:56-06:00

       -r, --reference=FILE

       -s, --set=STRING



       FORMAT controls the output.  Interpreted sequences are:
```

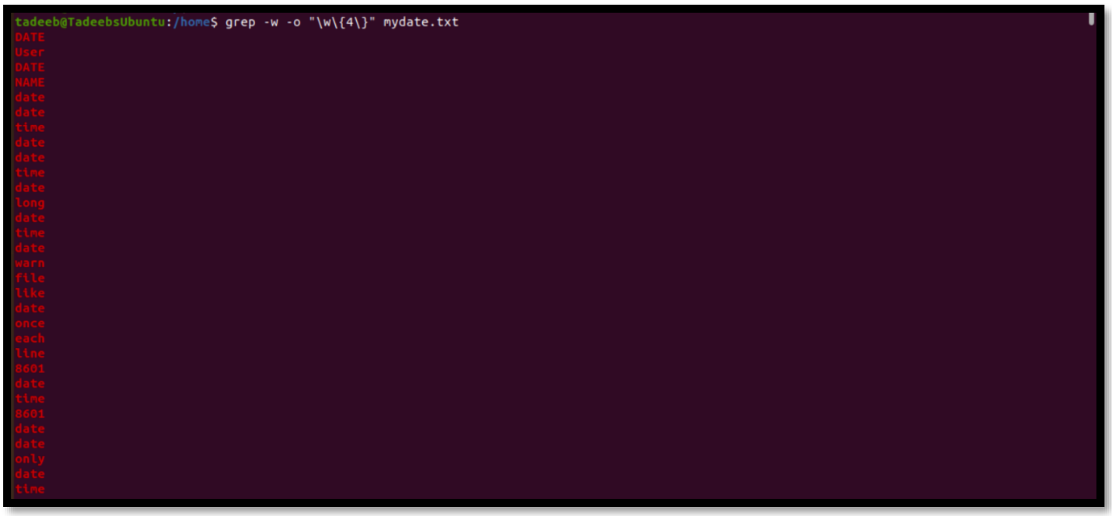
25) List the words of 4 letters from the file mydate.

- Brief:

Sno.	Command	Description
1	*grep*	It prints the line that matches patterns. -w By default, grep matches the given string/pattern even if it found as a substring in a file. The -w option to grep makes it match only the whole words and give line numbers where that word is present.

		-o By default, grep displays the entire line which has the matched string. We can make the grep to display only the matched string by using the -o option. It counts from substrings too. ***grep -w -o “\w\{4\}” filename***

- Output:



```
tadeeb@TadeebsUbuntu:/home$ grep -w -o "\w\{4\}" mydate.txt
DATE
User
DATE
NAME
date
date
time
date
date
time
date
long
date
time
date
warn
file
like
date
once
each
line
8601
date
time
8601
date
date
only
date
time
```

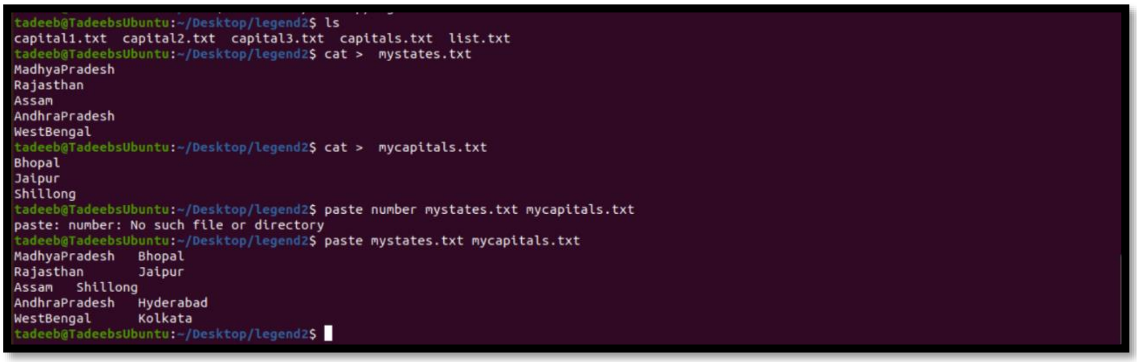
26) List 5 states in north east India in a file mystates. List their corresponding capitals in a file mycapitals. Use the paste command to join the 2 files.

- Brief:

Sno.	Command	Description
1	*paste*	Paste command is one of the useful commands in Unix or Linux operating system. It is used to join files horizontally (parallel merging) by outputting lines consisting of lines from each file specified, separated by tab as delimiter, to the standard output. When no file is specified, or put dash (“- “) instead of file name, paste

		reads from standard input and gives output as it is until a interrupt command **[Ctrl-c]** is given.

- Output:



```
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ ls
capital1.txt  capital2.txt  capital3.txt  capitals.txt  list.txt
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ cat >  mystates.txt
MadhyaPradesh
Rajasthan
Assam
AndhraPradesh
WestBengal
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ cat >  mycapitals.txt
Bhopal
Jaipur
Shillong
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ paste number mystates.txt mycapitals.txt
paste: number: No such file or directory
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ paste mystates.txt mycapitals.txt
MadhyaPradesh   Bhopal
Rajasthan       Jaipur
Assam   Shillong
AndhraPradesh   Hyderabad
WestBengal      Kolkata
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ 
```

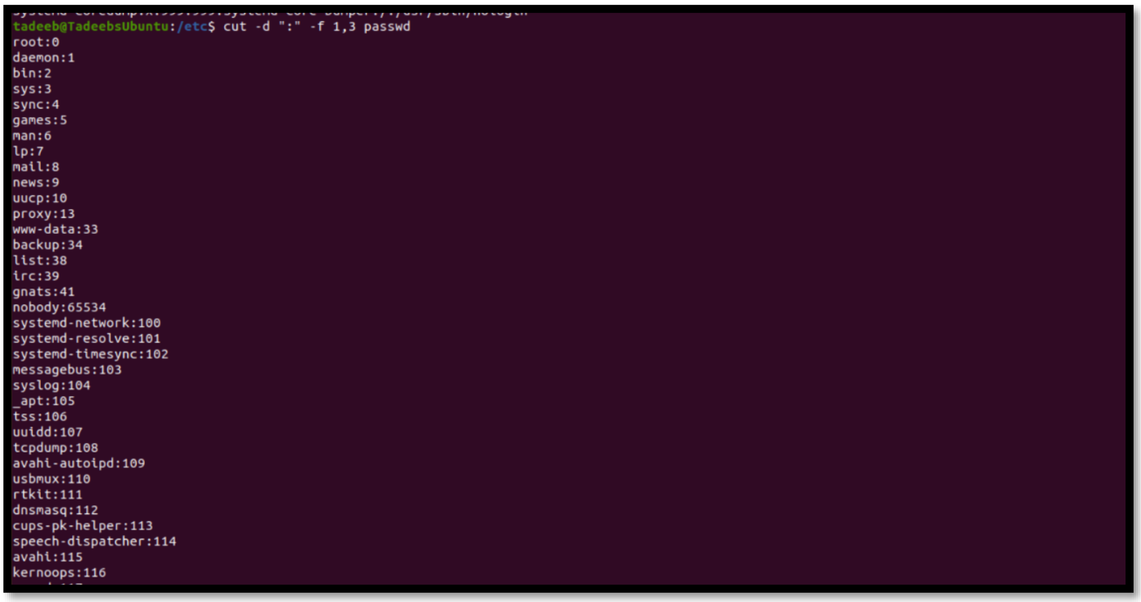
27) Use the cut command to print the 1st and 3rd columns of the /etc/passwd file for all students in this class.

- Brief:

Sno.	Command	Description
1	cut	The cut command in UNIX is a command for cutting out the sections from each line of files and writing the result to standard output. It can be used to cut parts of a line by byte position, character and field. Basically, the cut command slices a line and extracts the text. It is necessary to specify option with command otherwise it gives error. If more than one file name is provided then data from each file is not precedes by its file name. *-f* (field): -c option is useful for fixed-length lines. Most Unix files doesn't have fixed-length lines. To extract the useful information, you need to cut by fields rather than columns. List of the fields number specified must be separated by comma. *Ranges are not described with -f*

		option. cut uses tab as a default field delimiter but can also work with other delimiter by using -d option. Note: Space is not considered as delimiter in U Command: ***cut -d “:” -f 1,3 passwd***

- Output:



```
tadeeb@TadeebsUbuntu:/etc$ cut -d ":" -f 1,3 passwd
root:0
daemon:1
bin:2
sys:3
sync:4
games:5
man:6
lp:7
mail:8
news:9
uucp:10
proxy:13
www-data:33
backup:34
list:38
irc:39
gnats:41
nobody:65534
systemd-network:100
systemd-resolve:101
systemd-timesync:102
messagebus:103
syslog:104
_apt:105
tss:106
uuidd:107
tcpdump:108
avahi-autoipd:109
usbmux:110
rtkit:111
dnsmasq:112
cups-pk-helper:113
speech-dispatcher:114
avahi:115
kernoops:116
```

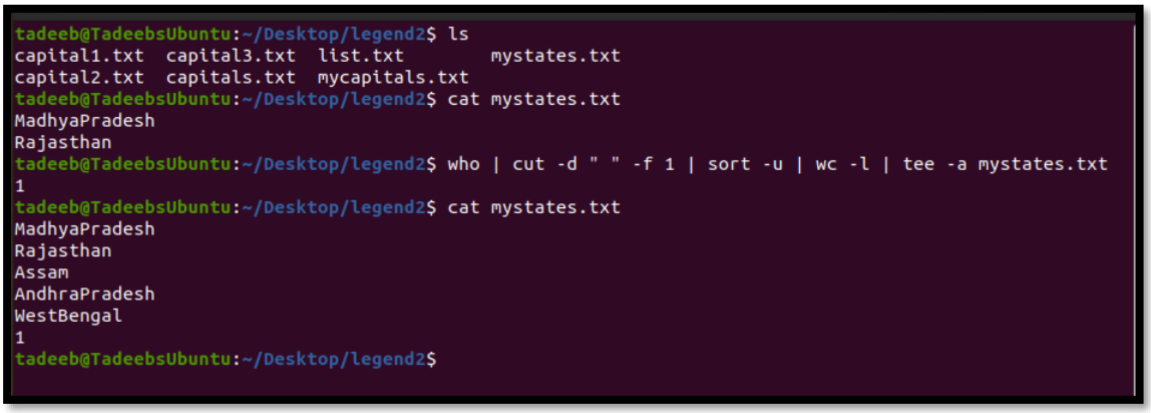
28) Count the number of people logged in and trap the users in a file using the tee command.

- Brief:

Sno.	Command	Description
1	*tee*	tee command reads the standard input and writes it to both the standard output and one or more files. The command is named after the T-splitter used in plumbing. It basically breaks the output of a program so that it can be both displayed and saved in a file. It does both the tasks simultaneously, copies the result into the specified files or variables and also display the result.

2	|	The output of one program in the pipeline is passed as input to the next.
3	*who*	We can count the total number of open sessions by counting the lines in the output of *who* or *w* with the *-h* option. (The *-h* option omits header lines, which we don't want to count.)
4	*cut*	*who* or *w* counts login sessions, but if a user has more than one login session open, they are counted more than once. To count unique users, we can use the cut command to strip all information except for the user's name

- Output:



```
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ ls
capital1.txt  capital3.txt  list.txt        mystates.txt
capital2.txt  capitals.txt  mycapitals.txt
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ cat mystates.txt
MadhyaPradesh
Rajasthan
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ who | cut -d " " -f 1 | sort -u | wc -l | tee -a mystates.txt
1
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ cat mystates.txt
MadhyaPradesh
Rajasthan
Assam
AndhraPradesh
WestBengal
1
tadeeb@TadeebsUbuntu:~/Desktop/legend2$
```

Taking the output of *who*, and display only the first field of information, which is delimited by a space. It gives us a list of only the usernames, but we still need to filter out repeated names.

To do this, we can add the *sort -u* command. This sorts the names alphabetically and filters out any lines that are not unique

And to count these unique users, we add *wc -l* at the end of our command pipeline.

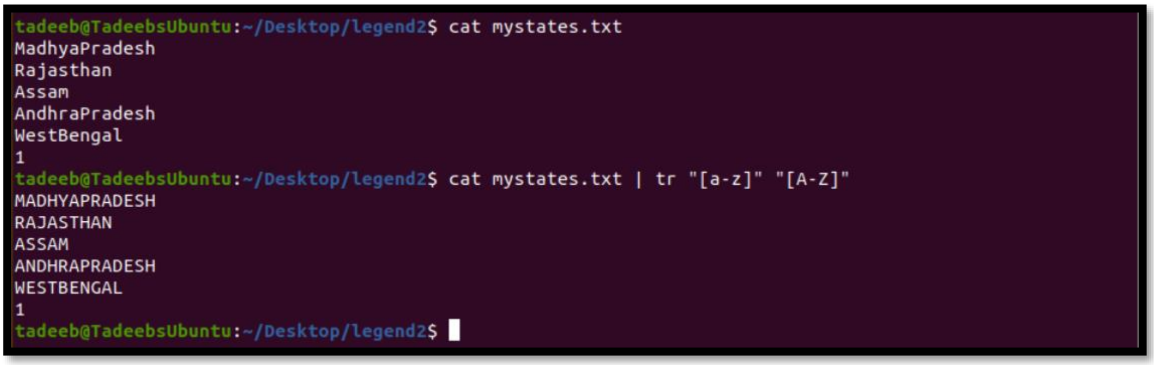
And finally, using *tee* command with option *-a* we appended the count in another file.

29) Convert the contents of mystates into uppercase.

- Brief:

Sno.	Command	Description
1	*tr*	The *tr* command in LINUX/UNIX is a command line utility for translating or deleting characters. It supports a range of transformations including uppercase to lowercase, squeezing repeating characters, deleting specific characters and basic find and replace. It can be used with LINUX/UNIX pipes to support more complex translation. tr stands for translate.

- Output:



```
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ cat mystates.txt
MadhyaPradesh
Rajasthan
Assam
AndhraPradesh
WestBengal
1
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ cat mystates.txt | tr "[a-z]" "[A-Z]"
MADHYAPRADESH
RAJASTHAN
ASSAM
ANDHRAPRADESH
WESTBENGAL
1
tadeeb@TadeebsUbuntu:~/Desktop/legend2$
```

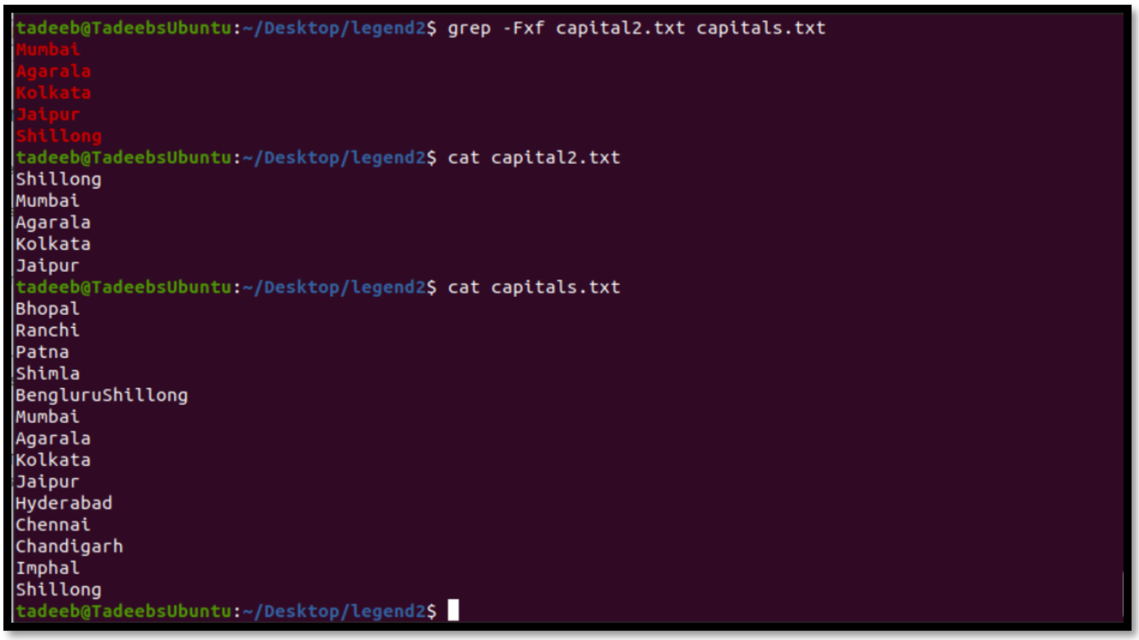
30) Create any two files & display the common values between them.

- Brief:

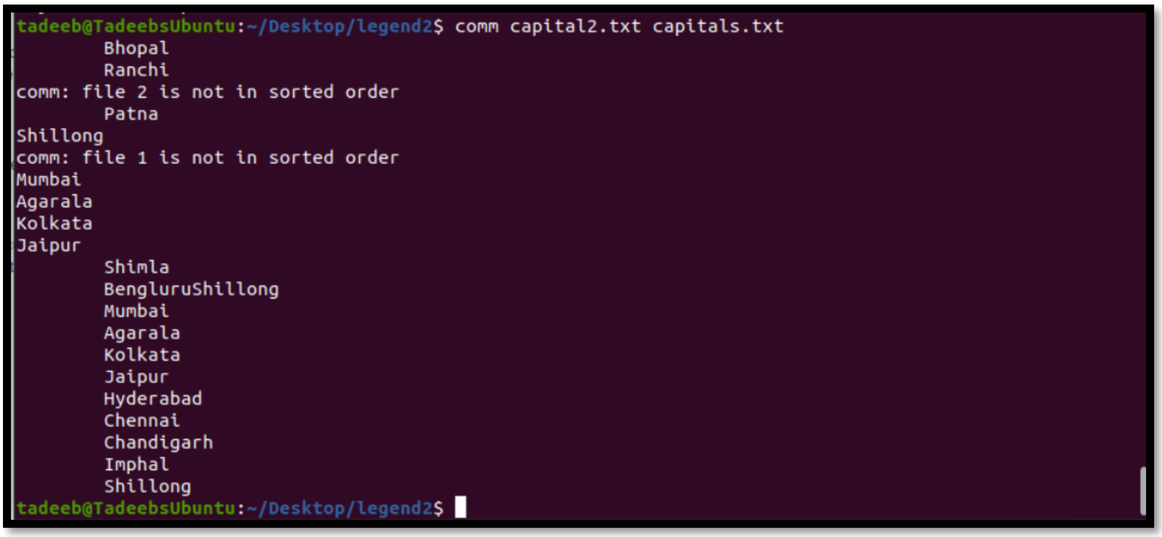
Sno.	Command	Description
1	*sort*	*sort* command is used to sort a file, arranging the records in a particular order. By default, the sort command sorts file assuming the contents are ASCII. Using options in sort command, it can also be used to sort numerically.

2	*comm*	*comm* compare two sorted files line by line and write to standard output; the lines that are common and the lines that are unique.
		• As using comm, we are trying to compare two files therefore the syntax of comm command needs two filenames as arguments. • With no OPTION used, comm produces three-column output where first column contains lines unique to FILE1, second column contains lines unique to FILE2 and third and last column contains lines common to both the files. • *comm* command only works right if you are comparing two files which are already sorted.

- Output:



```
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ grep -Fxf capital2.txt capitals.txt
Mumbai
Agarala
Kolkata
Jaipur
Shillong
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ cat capital2.txt
Shillong
Mumbai
Agarala
Kolkata
Jaipur
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ cat capitals.txt
Bhopal
Ranchi
Patna
Shimla
BengluruShillong
Mumbai
Agarala
Kolkata
Jaipur
Hyderabad
Chennai
Chandigarh
Imphal
Shillong
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ 
```



```
tadeeb@TadeebsUbuntu:~/Desktop/legend2$ comm capital2.txt capitals.txt
        Bhopal
        Ranchi
comm: file 2 is not in sorted order
        Patna
Shillong
comm: file 1 is not in sorted order
Mumbai
Agarala
Kolkata
Jaipur
        Shimla
        BengluruShillong
        Mumbai
        Agarala
        Kolkata
        Jaipur
        Hyderabad
        Chennai
        Chandigarh
        Imphal
        Shillong
tadeeb@TadeebsUbuntu:~/Desktop/legend2$
```