Ex.No: 1 **BASIC UNIX COMMANDS**
Date :27.02.2021

Aim
To understand the basic commands used to work with Unix environment.

General Command Structure

Syntax:

Command [-options] arguments

Where,

Command	name of the command
Arguments	file name, user name or some other information that the program needs.
Options	which is a option for a program. So it is put inside square brackets.

BASIC COMMANDS

1. Command : **who**
   Purpose : It is used to get the information about all the users currently working in the system.
   Syntax : who
   Example **:** $ who

2. Command : **who am i**
   Purpose : It is used to know in which terminal the user is currently logged on.
   Syntax : who am i
   Example : $ who am I
3. Command : **date**
   Purpose : It is used to display the system date and time.
   Syntax : date
   Example : $ date
4. Command : **cal**
   Purpose : It prints the calender for the specified year and month.
   Syntax :cal<month><year>
   Example : $ cal 05 2003

5. Command : **id**
   Purpose : It is used to display the login name.
   Syntax : id
   Example : $ id

6. Command : **clear**
   Purpose : It is used to clear the screen.
   Syntax : clear
   Example : $ clear

J.Tavamani raja durai
19CSE060

7. Command : **uname**
   Purpose : It is used to display the details about the OS in which we are working.
   Syntax :uname [options]
   Example : $ uname –n

8. Command : **tty**
   Purpose : It is used to know the terminal name on which we work.
   Syntax :tty
   Example : $ tty
9. Command : **pwd**
   Purpose : It is used to display the absolute pathname of current working directory.
   Syntax :pwd
   Example : $ pwd
10. Command : **bc**
    Purpose : It is used to perform simple mathematical calculations.
    Syntax :bc filename

Example : $ bcpp

11. Command :**echo**
    Purpose : It echoes the argument on the standard output textce.
    Syntax : echo [options] <string>
    Example : $ echo 'BOOM'
12. Command : **man**
    Purpose : It gives details about the unix commands.
    Syntax :man < command name >
    Example : $ man echo

FILE MANIPULATION COMMANDS

1. Command **:cat**
   Purpose **:** It is used to create a new file
   Syntax **:** cat ><file name>
   Example **:** $ cat >
   This is sample File in Unix
   Ctrl – d

   To append the content of already existing file.
   Example : $ cat>>

   It is also used to display the contents of the fileas well as used to create a new file.
   Syntax **:** cat <file name >
   Example **:** $ cat
   This is sample File in Unix

   To display the contents of two or more files, specify the filenames with the cat commands separated by a space in between them.
   Example:

$cat 1 2 3
1 file
2 file
3 file

2. Command : **ls**
   Purpose : It is used to display the files in the current working directory.
   Syntax : ls [options] <arguments>
   Options
   a - to list all directory entries
   d - to list name of directories
   l - to list files in long form
   r - to list file in the reverse order
   t - to list the files sorted by time
   lrt - list content of current directory. Output will be sorted based on modification date & time.

   Example : $ ls –l

3. Command : **tail**
   Purpose :It is used to Prints the last several lines of the specified files.
   Syntax : tail [options] <file name>
   Example :$ tail -5 text

4. Command : **head**
   Purpose :It is used to display the top portion of the file where top portion represents the no'sof lines.
   Syntax : head [options] <file name>
   Example : $ head -5 text
5. Command :**cmp**
   Purpose :Compare two files, and if they differ, tells the first byte and line number where they differ.
   Syntax :cmp file1 file2
   Example : $ cmp a1 a2

6. Command :**diff**
   Purpose :The diff command analyses line by line and displays a list of changes between two files.
   Syntax :diff file1 file2
   Example :$ diff a1 a2

7. Command **: wc**
   Purpose **:** It is used to count the number of lines,words and characters in a file or group of files.
   Syntax :wc [options] <file name >
   Example **:** $ wc .txt

2 15 95 .txt

The options used with **wc** commands are listed.

Option	Description
-c	Count number of characters in the file
-w	Counts the number of words in the file
-l	Counts number of lines in the file

Example:

```
$ wc –c .txt
95  .txt
$ wc –w .txt
15 .txt
$ wc –l .txt
2 .txt
```

8. Command **: sort**
   Purpose **:** Sorts the specified files. The command has many useful arguments..
   Syntax :sort [options] <file name >
   Option-r reverse
   Example **:** $ sort .txt

9. Command : **pr**
   Purpose : It is used to display the contents of the file by separating them into pages and each page begins with the header information.
   Syntax : pr [options] <file name >
   Example : $ pr
10. Command : **cut**
    Purpose : It is used to extract selected fields or columns from each line of one or more files and display them on the standard output textce.
    Syntax : cut [options] <file name >
    Example : $ cut –c5

11. Command **: paste**
    Purpose **:** It concatenates the line from each input file column by column with tab characters in between them.
    Syntax **:** paste [options] <file name >
    Example **:** $ paste f1 f2

12. Command **: join**
    Purpose **:** It is used to extracts common lines from two sorted files and there should be the common field in both file.
    Syntax **:** join [options] <file name1 ><file name 2>
    Example **:** $ join –a1 f1 f2
13. Command **:uniq**
    Purpose **:** It compares adjacent lines in the file and displays the output by eliminating duplicate adjacent lines in it.

J.Tavamani raja durai
19CSE060

Syntax :uniq [options] <file name >
Example :$ uniq -c text

14. Command **:nl**
    Purpose **:** It is used to add the line numbers to the file.
    Syntax :nl [options] [filename]
    Example **:** $ nl text
15. Command **: tr**
    Purpose **:** It is used to translate or delete a character or a string from the standard input to produce the required output.
    Syntax :tr [options] <string1><string2>
    Example : $ tr –s 'a' 'b' < text
16. Command **:tee**
    Purpose **:** It is used to read the contents from standardinput or from output of another command and reproduces the output to both in standardoutput and direct into output to one or more files.
    Syntax :command | tee [options] <file name >
    Example **:** $ date | tee dat.txt
17. Command **:grep**
    Purpose **:** It is used to search the specified pattern from one or more files.
    Syntax :grep [options] <pattern><file name >
    Example **:** $ grep welcometext

DIRECTORY MANUPULATION COMMANDS

1. Command :mkdir
   Purpose **:** It is used to create new directory or more than one directory.
   Syntax :mkdir<directory name >
   Example :$ mkdirsudhan

2. Command :cd
   Purpose **:** It is used to change the control from oneworking directory to another specified directory.
   Syntax :cd <directory name >
   Example **:** $ cd sudhan

3. Command :rmdir
   Purpose **:** It is used to remove the directory if it is empty.
   Syntax :rmdir<directory name >
   Example **:** $ rmdirsudhan

   You can use the –r(recursive) option with the rmdir command so that it deletes the directory even when it is not empty.
   For example, $rmdir –r haran

4. Command :cp
   Purpose **:** It is used to copy one or more files.

Syntax **:**cp<source file name ><destination file name>
Example **:** $ cptexttext1

5. Command **:** more
   Purpose **:** It is used to control the information on the screen from a line to a screen full.
   Syntax **:** more options <file name >
   Example **:**$ more

   This command displays one screen full of information from the file. To display the next screen press enter key or space bar. To quit press Q.

6. Command **:**passwd
   Example : passwd user
   Changes the current user's password, or that of the specified user (requires root privileges). The command prompts for the new password.

7. Command **:** mv
   Purpose **:** It is used to move a file within a directory with different names and also used to move a file to different directory with its original name.
   Syntax **:**mv<source file name ><destination directory name>
   Example **:** $ mv texttext2

8. **chmod command**
   You set the access modes of a directory or file by using the chmod command, which has the following pattern:

   chmod
   nnn directory-or-file

   The argument *nnn* is a three-digit number, which gives the access mode for the owner, group, and other users.
   For example, the argument 751 is equivalent to rwxr-x--x, which gives the owner every possible permission, gives the group read and execute permission, and gives other users execute permission.

9. Command **:** write
   Purpose **:** the write command can be used by any user to write something on someone else's terminal, provided the recipient of the message permits communication
   Syntax **:** write <user name>
   Example **:** $ write user2

   This is sample message.
   Ctrl d

   On executing this command the message would be relayed to the user whose login name is user2. He would hear a beep on his terminal, followed by the message.
   There are two prerequisites for a smooth write operation:

1. The recipient must be logged in; else an error message is inevitable.
2. The recipient must have given permission for messages to reach his or her terminal.

10. wall command

Prints a message to each user except those who've disabled message reception.Type **Ctrl-D** to end the message.

11. Command **:** News

Purpose **:** The system administrator is the sole person whocan make news under the Unix OS. He types the information which he wants everyone on the network to know of in different files in /usr/news directory.

Syntax **:** 1) $news filename
2) $news [-options]

Options : **:** 1) -n option only lists the names of the news items from the /usr/news directory that have not yet been read by you.
2) –s option which provides a count of the unread new items in the /usr/news directory.
3) –a shows all the news.

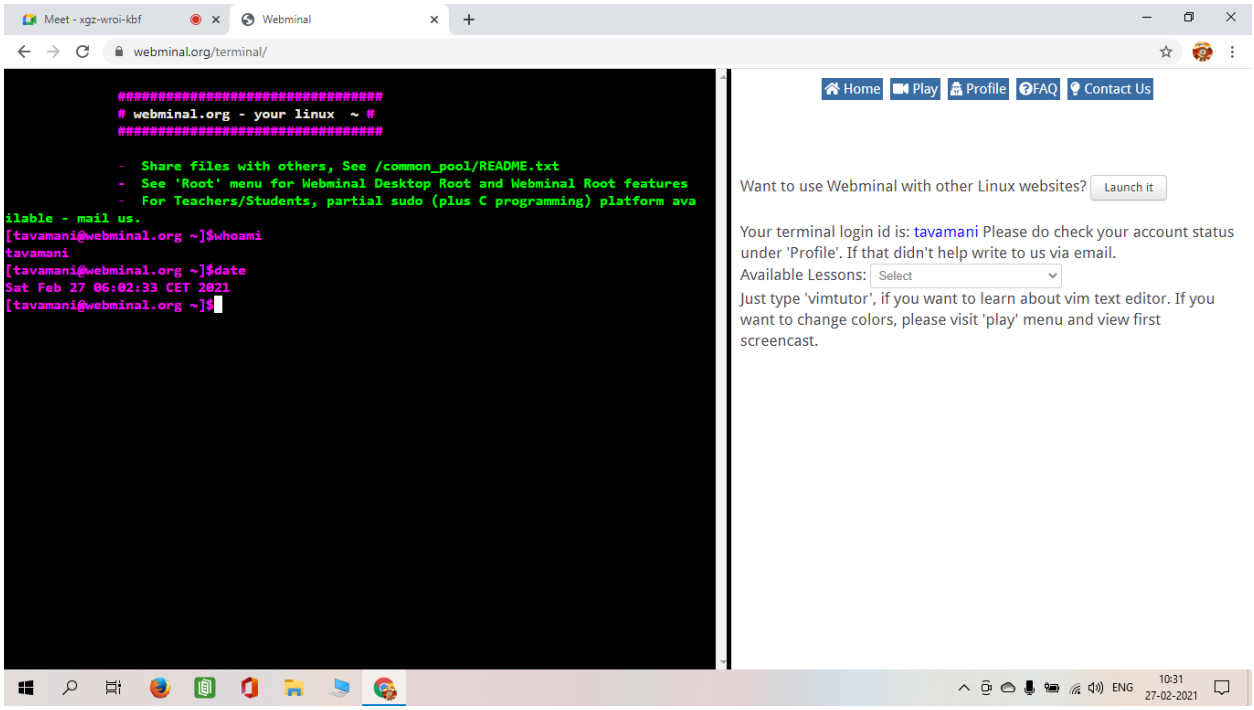
Example 1 **:** $ news sample

Where sample is the file name in which the news items are available.

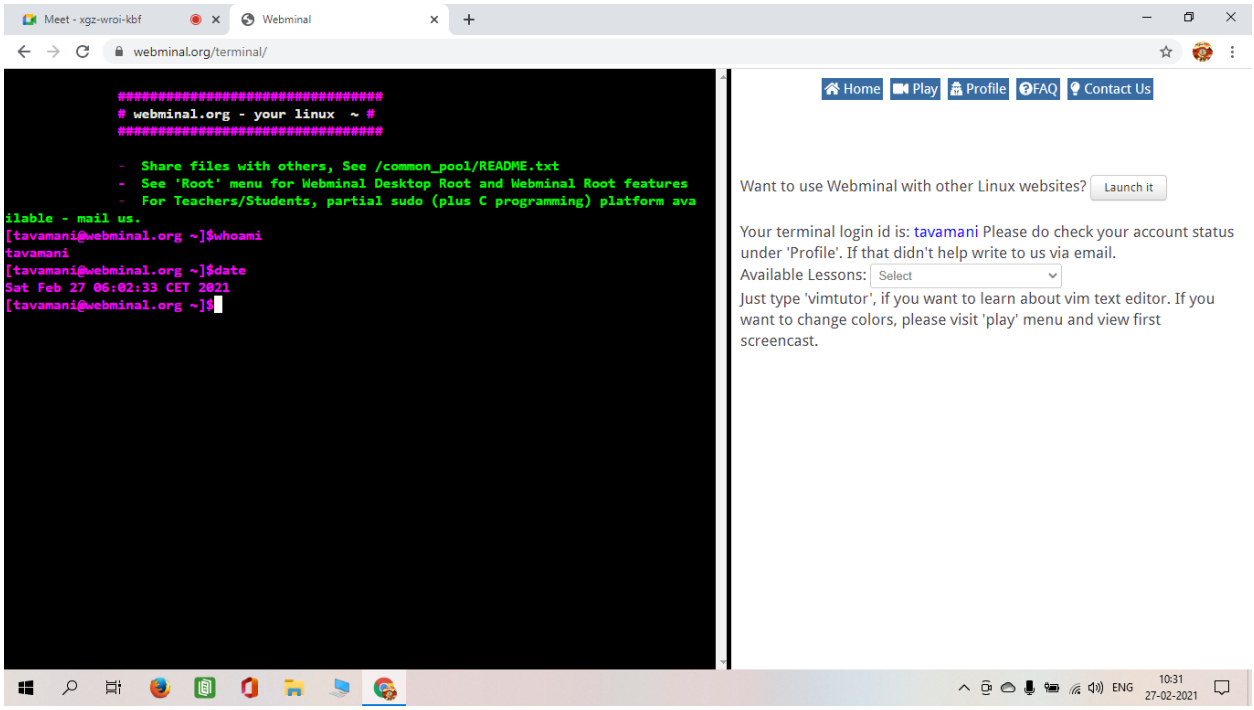
Example 2 : $news –s

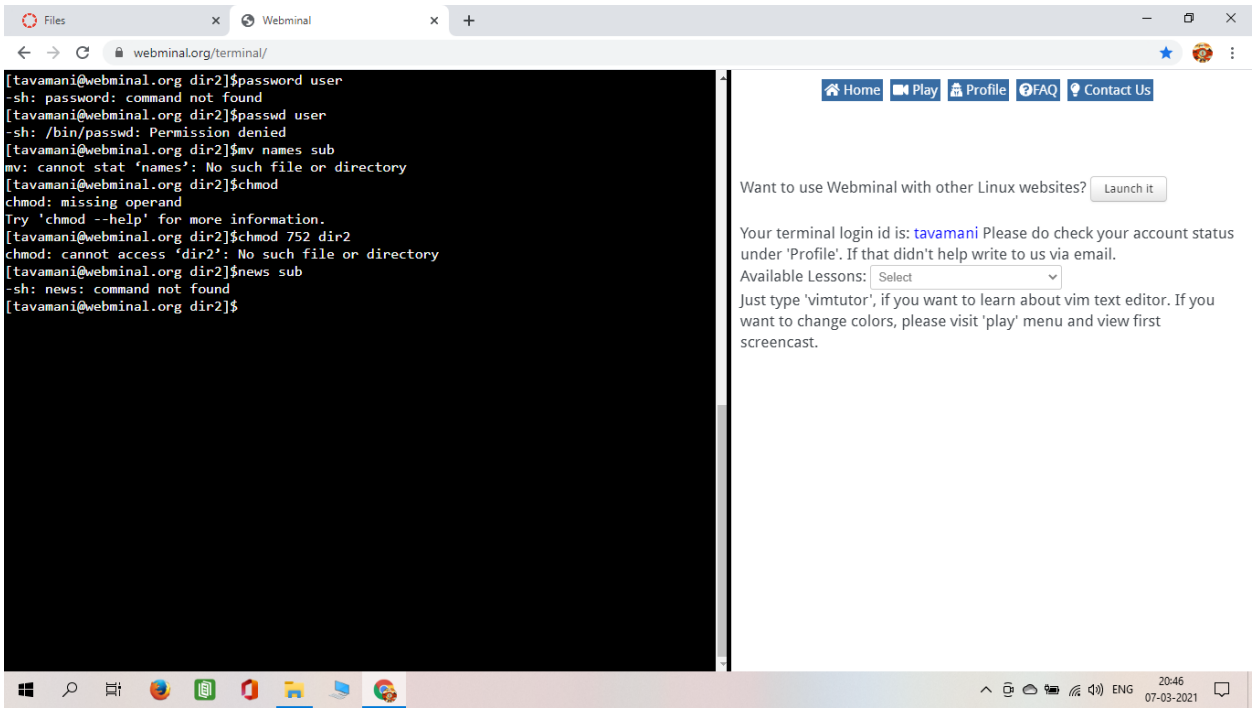
It provides the count of the unread new items.

Output:

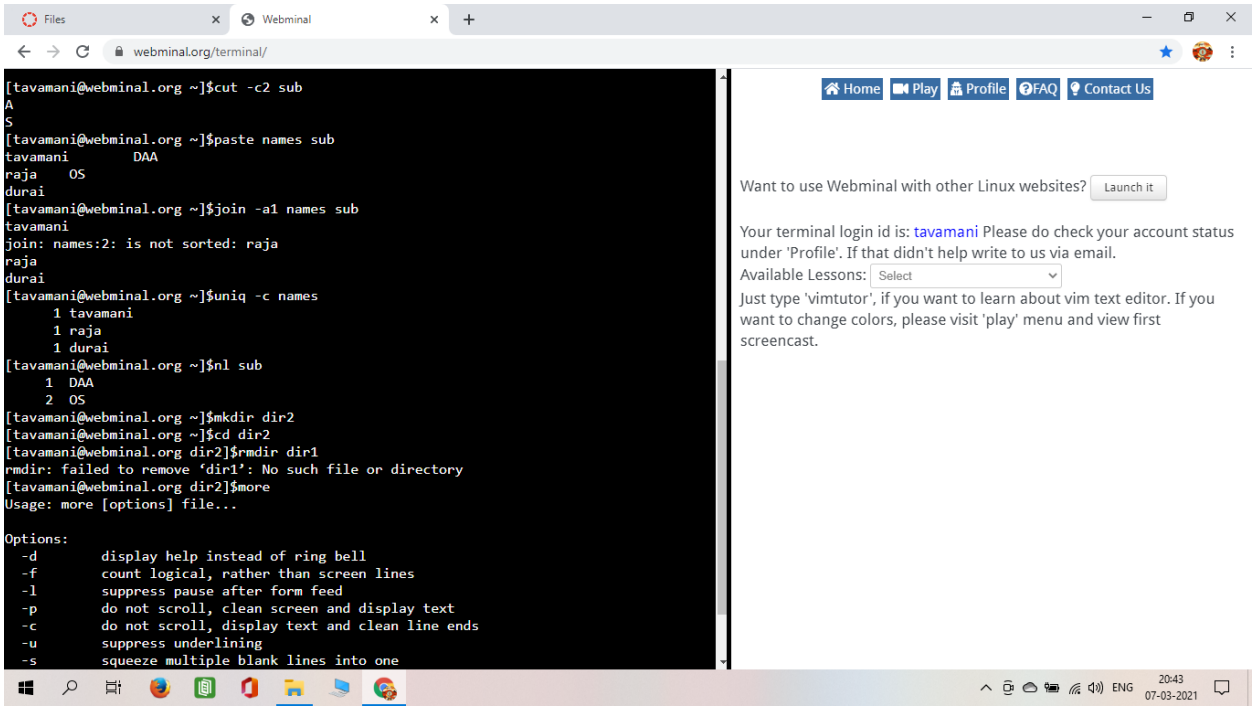


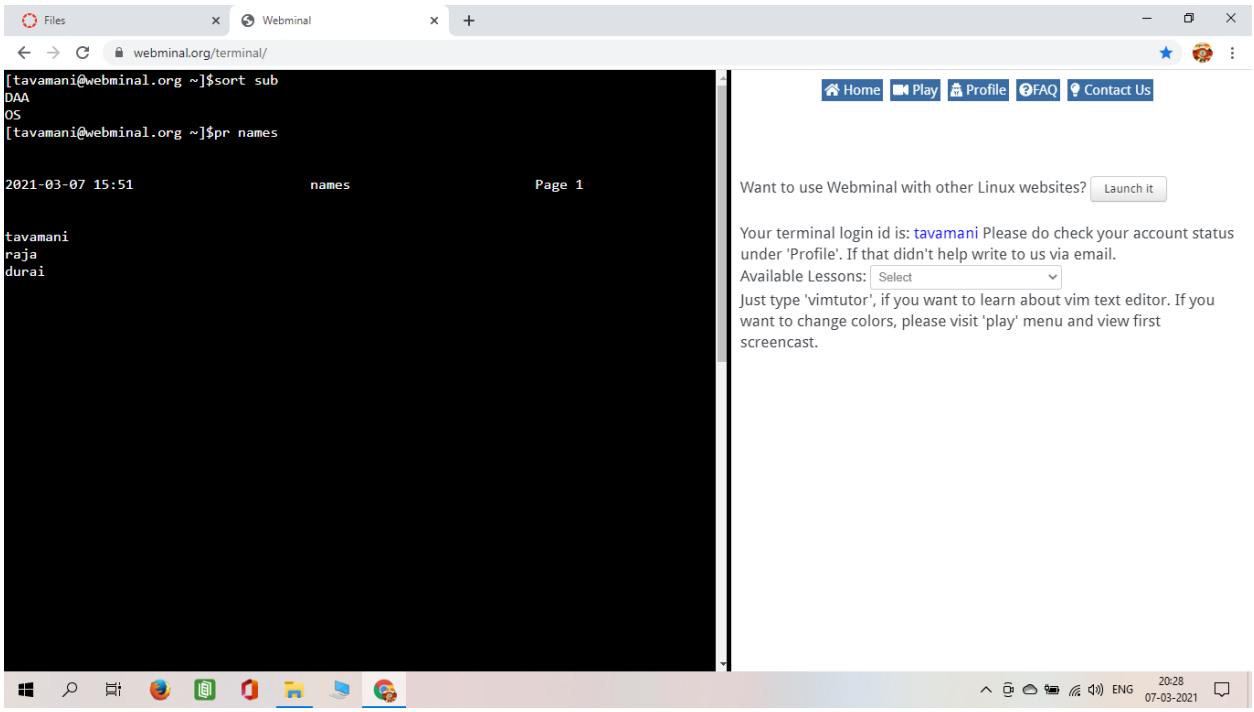
J.Tavamani raja durai
19CSE060



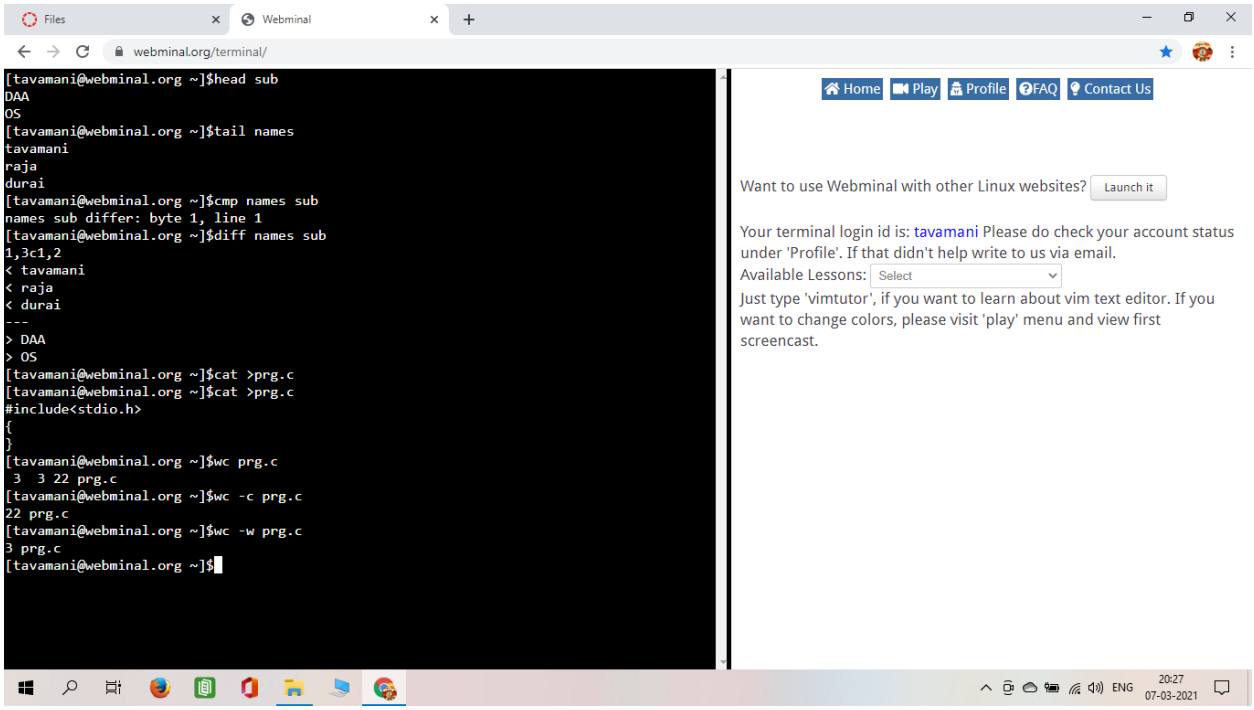


J.Tavamani raja durai
19CSE060

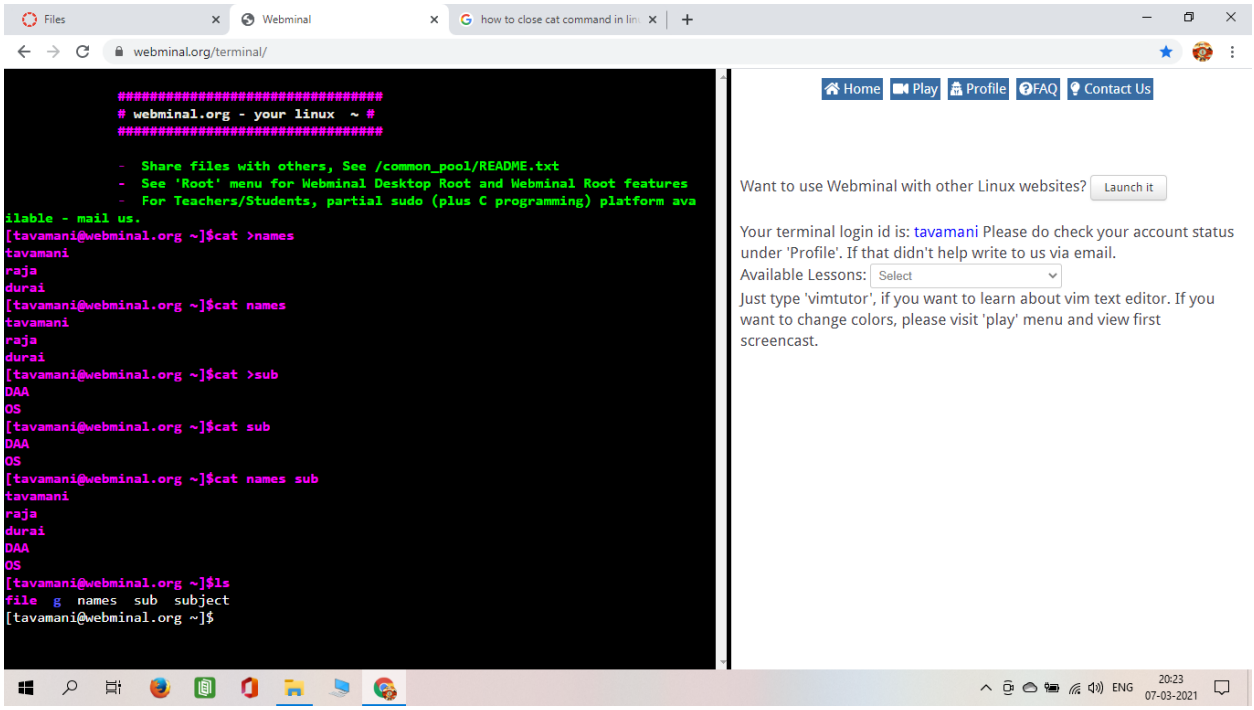


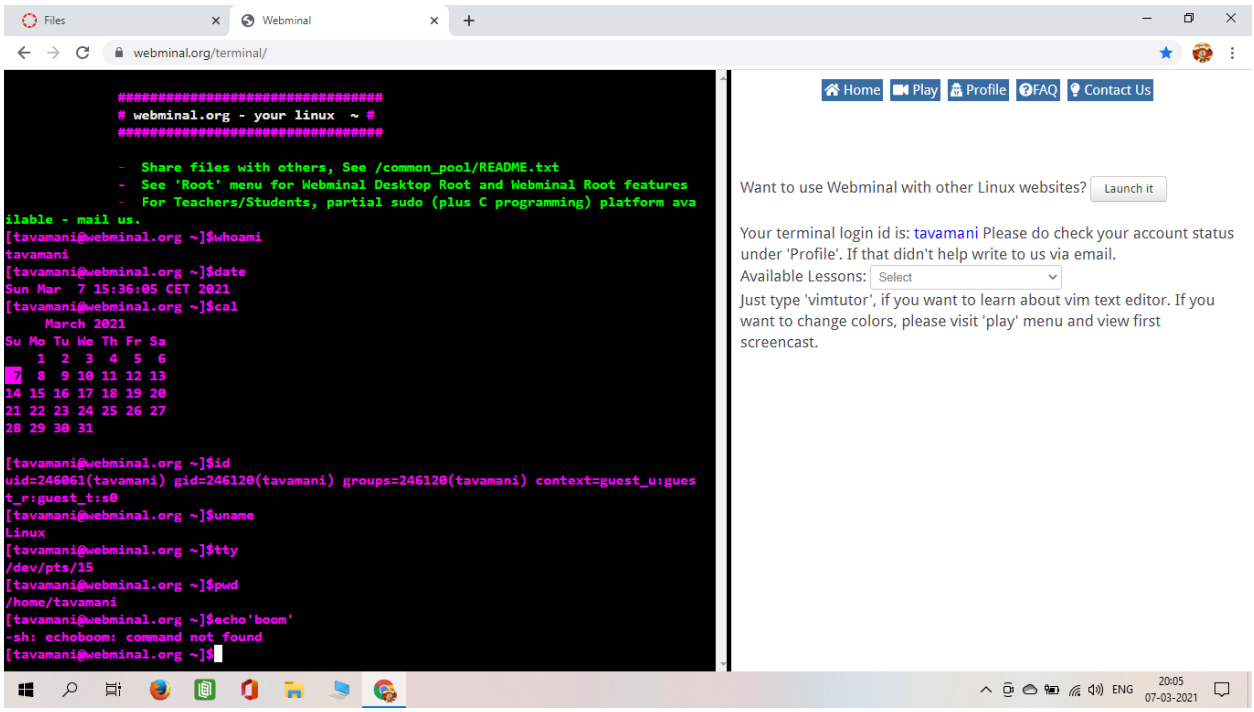


J.Tavamani raja durai
19CSE060



J.Tavamani raja durai
19CSE060





J.Tavamani raja durai
19CSE060

Observation(20)	
Record(5)	
Total(25)	
Initial	

Result:

Thus the basic Unix commands were executed and outputs were noted.

J.Tavamani raja durai
19CSE060