Assignment 1

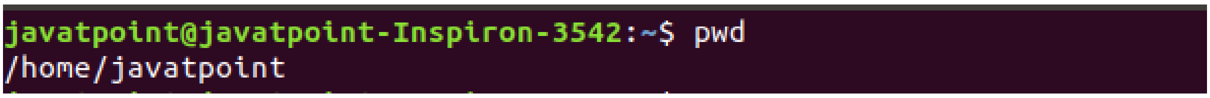
1. pwd Command

The pwd command is used to display the location of the current working directory.

Syntax:

```
pwd
```

Output:



```
javatpoint@javatpoint-Inspiron-3542:~$ pwd
/home/javatpoint
```

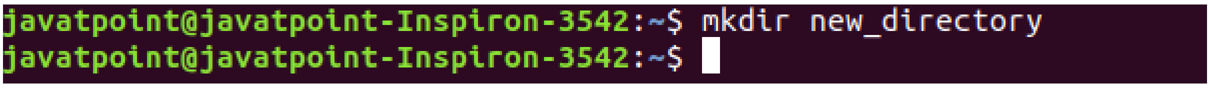
2. mkdir Command

The mkdir command is used to create a new directory under any directory.

Syntax:

```
mkdir <directory name>
```

Output:



```
javatpoint@javatpoint-Inspiron-3542:~$ mkdir new_directory
javatpoint@javatpoint-Inspiron-3542:~$ 
```

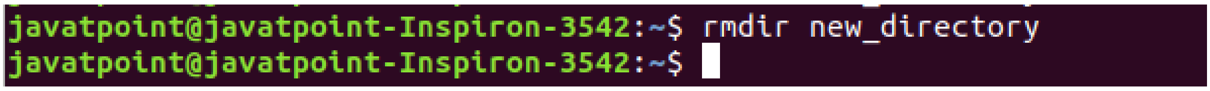
3. rmdir Command

The rmdir command is used to delete a directory.

Syntax:

```
rmdir <directory name>
```

Output:



```
javatpoint@javatpoint-Inspiron-3542:~$ rmdir new_directory
javatpoint@javatpoint-Inspiron-3542:~$ 
```

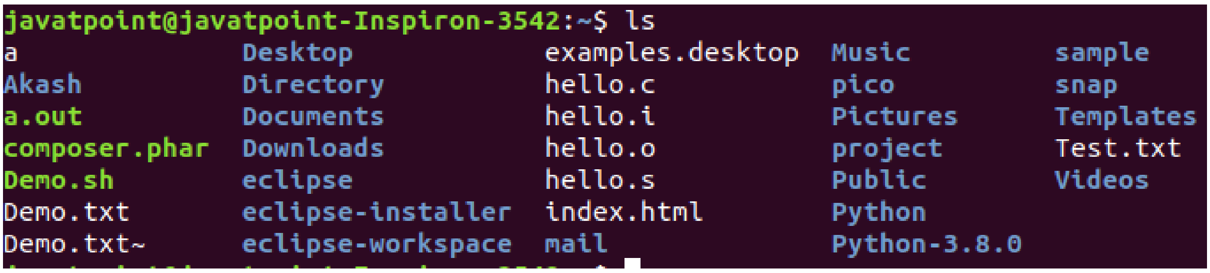
4. ls Command

The ls command is used to display a list of content of a directory.

Syntax:

1. ls

Output:



```
javatpoint@javatpoint-Inspiron-3542:~$ ls
a              Desktop            examples.desktop  Music           sample
Akash          Directory          hello.c           pico            snap
a.out          Documents          hello.i           Pictures        Templates
composer.phar  Downloads          hello.o           project         Test.txt
Demo.sh        eclipse            hello.s           Public          Videos
Demo.txt       eclipse-installer  index.html        Python
Demo.txt~      eclipse-workspace  mail              Python-3.8.0
```

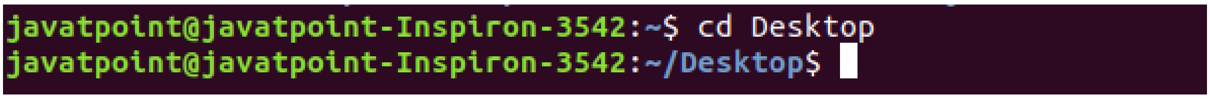
5. cd Command

The cd command is used to change the current directory.

Syntax:

1. cd <directory name>

Output:



```
javatpoint@javatpoint-Inspiron-3542:~$ cd Desktop
javatpoint@javatpoint-Inspiron-3542:~/Desktop$
```

Linux File commands

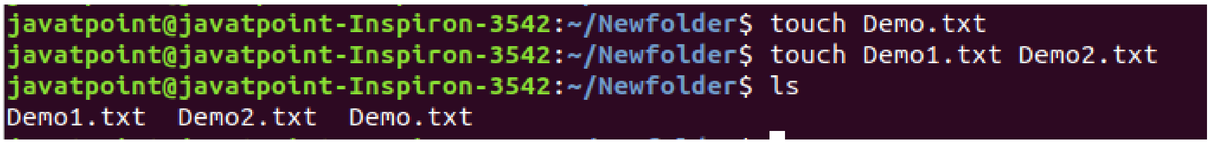
6. touch Command

The touch command is used to create empty files. We can create multiple empty files by executing it once.

Syntax:

1. touch <file name>
2. touch <file1> <file2> ....

Output:



```
javatpoint@javatpoint-Inspiron-3542:~/Newfolder$ touch Demo.txt
javatpoint@javatpoint-Inspiron-3542:~/Newfolder$ touch Demo1.txt Demo2.txt
javatpoint@javatpoint-Inspiron-3542:~/Newfolder$ ls
Demo1.txt  Demo2.txt  Demo.txt
```

7. cat Command

The cat command is a multi-purpose utility in the Linux system. It can be used to create a file, display content of the file, copy the content of one file to another file, and more.

Syntax:

1. cat [OPTION]... [FILE]..

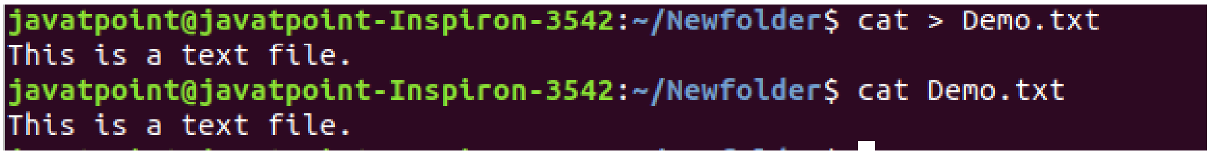
To create a file, execute it as follows:

1. cat > <file name>
2. // Enter file content

Press "**CTRL+ D**" keys to save the file. To display the content of the file, execute it as follows:

1. cat <file name>

Output:



```
javatpoint@javatpoint-Inspiron-3542:~/Newfolder$ cat > Demo.txt
This is a text file.
javatpoint@javatpoint-Inspiron-3542:~/Newfolder$ cat Demo.txt
This is a text file.
```

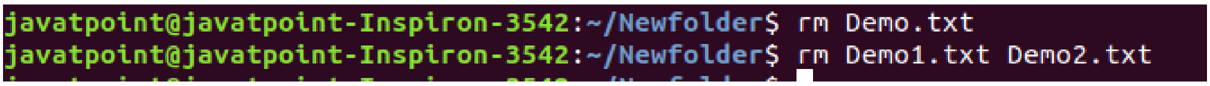
8. rm Command

The rm command is used to remove a file.

Syntax:

rm <file name>

Output:



```
javatpoint@javatpoint-Inspiron-3542:~/Newfolder$ rm Demo.txt
javatpoint@javatpoint-Inspiron-3542:~/Newfolder$ rm Demo1.txt Demo2.txt
```

9. cp Command

The cp command is used to copy a file or directory.

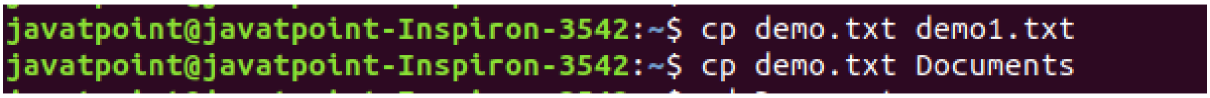
Syntax:

To copy in the same directory:

1. cp <existing file name> <new file name>

To copy in a different directory:

Output:



```
javatpoint@javatpoint-Inspiron-3542:~$ cp demo.txt demo1.txt
javatpoint@javatpoint-Inspiron-3542:~$ cp demo.txt Documents
```

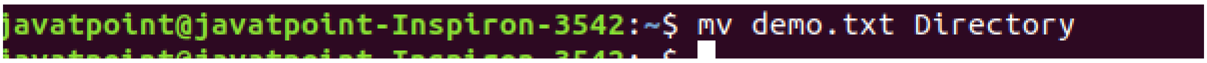
10. mv Command

The mv command is used to move a file or a directory form one location to another location.

Syntax:

1. mv <file name> <directory path>

Output:



```
javatpoint@javatpoint-Inspiron-3542:~$ mv demo.txt Directory
```

11. rename Command

The rename command is used to rename files. It is useful for renaming a large group of files.

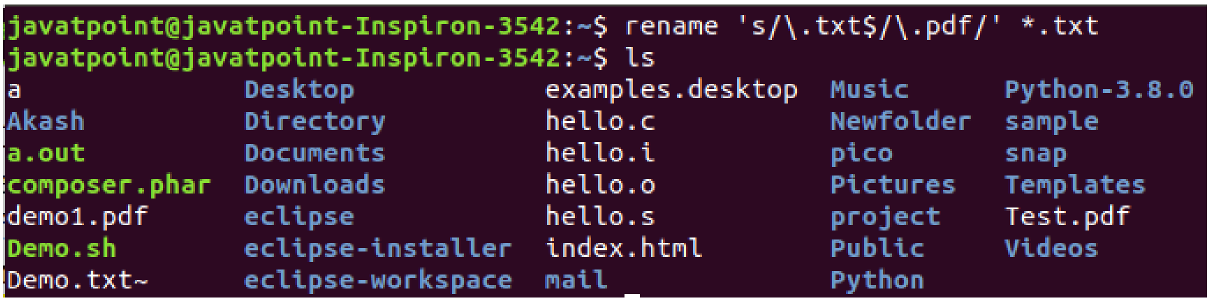
Syntax:

1. rename 's/old-name/new-name/' files

For example, to convert all the text files into pdf files, execute the below command:

1. rename 's/\.txt$/\.pdf/' *.txt

Output:



```
javatpoint@javatpoint-Inspiron-3542:~$ rename 's/\.txt$/\.pdf/' *.txt
javatpoint@javatpoint-Inspiron-3542:~$ ls
a              Desktop            examples.desktop  Music      Python-3.8.0
Akash          Directory          hello.c           Newfolder  sample
a.out          Documents          hello.i           pico       snap
composer.phar  Downloads          hello.o           Pictures   Templates
demo1.pdf      eclipse            hello.s           project    Test.pdf
Demo.sh        eclipse-installer  index.html        Public     Videos
Demo.txt~      eclipse-workspace  mail              Python
```

Linux File Content Commands

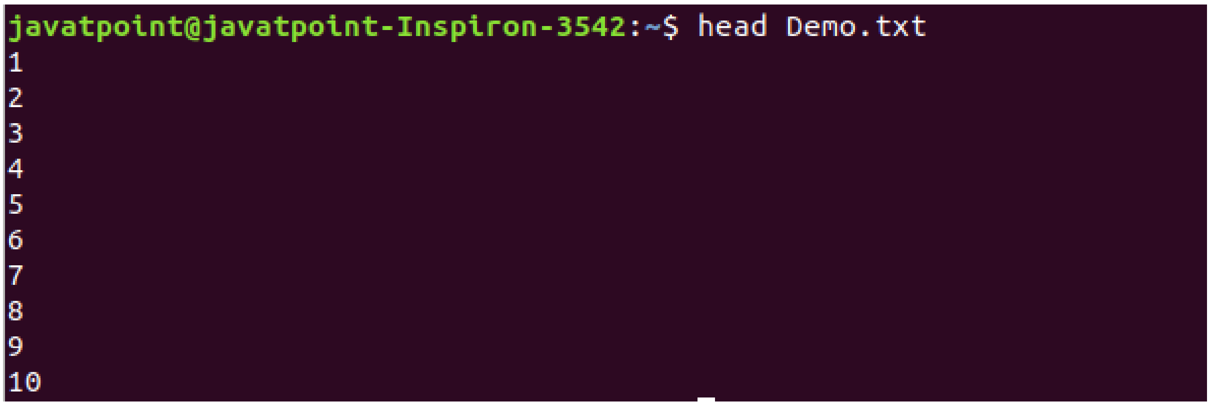
12. head Command

The head command is used to display the content of a file. It displays the first 10 lines of a file.

Syntax:

1. head <file name>

Output:



```
javatpoint@javatpoint-Inspiron-3542:~$ head Demo.txt
1
2
3
4
5
6
7
8
9
10
```

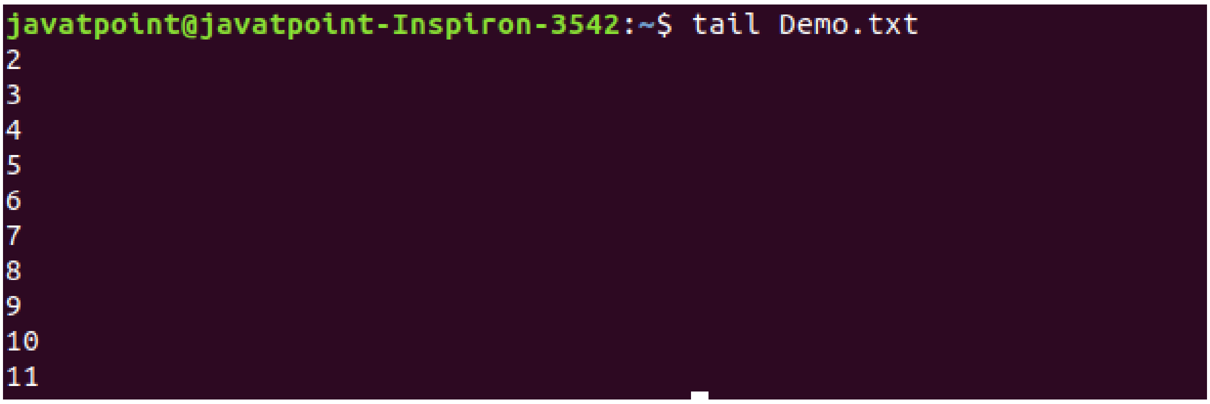
13. tail Command

The tail command is similar to the head command. The difference between both commands is that it displays the last ten lines of the file content. It is useful for reading the error message.

Syntax:

1. tail <file name>

Output:



```
javatpoint@javatpoint-Inspiron-3542:~$ tail Demo.txt
2
3
4
5
6
7
8
9
10
11
```

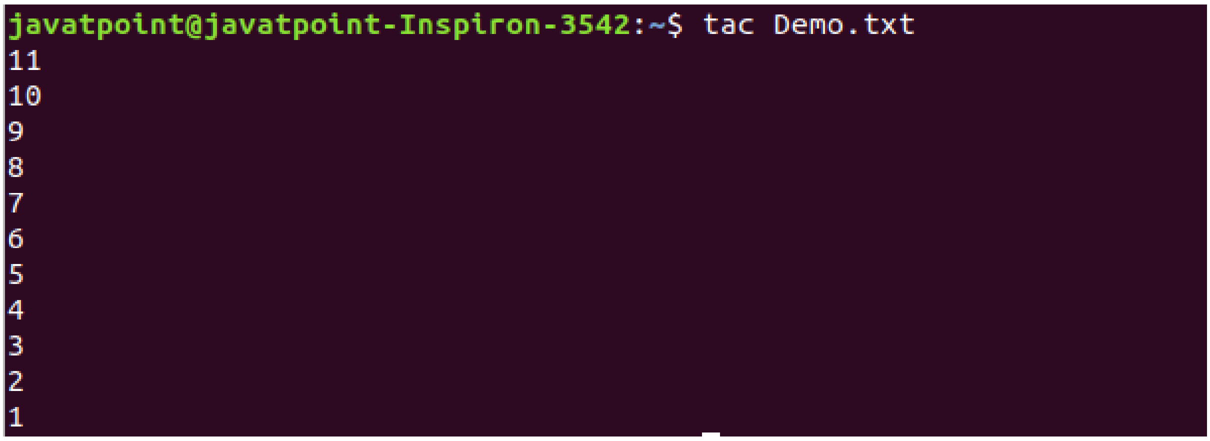
14. tac Command

The tac command is the reverse of cat command, as its name specified. It displays the file content in reverse order (from the last line).

Syntax:

```
1. tac <file name>
```

Output:



```
javatpoint@javatpoint-Inspiron-3542:~$ tac Demo.txt
11
10
9
8
7
6
5
4
3
2
1
```

15. more command

The more command is quite similar to the cat command, as it is used to display the file content in the same way that the cat command does. The only difference between both commands is that, in case of larger files, the more command displays screenful output at a time.

In more command, the following keys are used to scroll the page:

ENTER key: To scroll down page by line.

Space bar: To move to the next page.

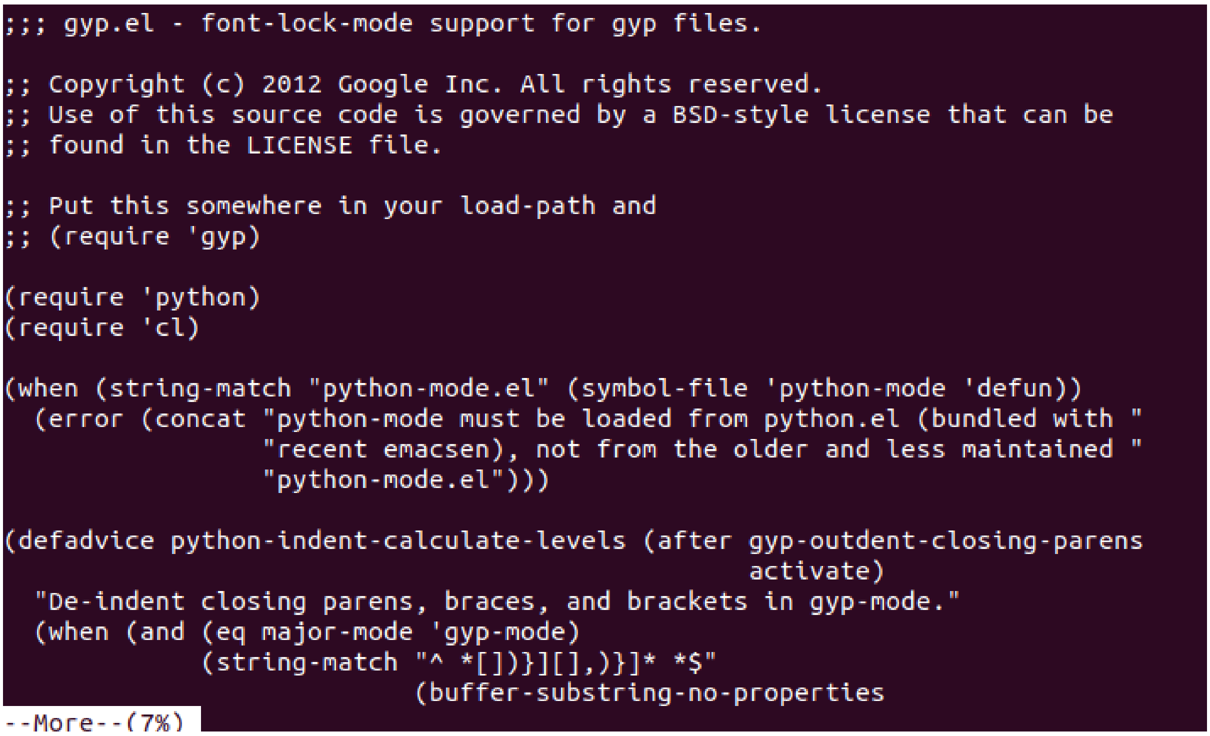
b key: To move to the previous page.

/ key: To search the string.

Syntax:

1. more <file name>

Output:



```
;;; gyp.el - font-lock-mode support for gyp files.

;; Copyright (c) 2012 Google Inc. All rights reserved.
;; Use of this source code is governed by a BSD-style license that can be
;; found in the LICENSE file.

;; Put this somewhere in your load-path and
;; (require 'gyp)

(require 'python)
(require 'cl)

(when (string-match "python-mode.el" (symbol-file 'python-mode 'defun))
  (error (concat "python-mode must be loaded from python.el (bundled with "
                 "recent emacsen), not from the older and less maintained "
                 "python-mode.el")))

(defadvice python-indent-calculate-levels (after gyp-outdent-closing-parens
                                                 activate)
  "De-indent closing parens, braces, and brackets in gyp-mode."
  (when (and (eq major-mode 'gyp-mode)
             (string-match "^ *[])}][],)}]* *$"
                           (buffer-substring-no-properties
--More--(7%)
```

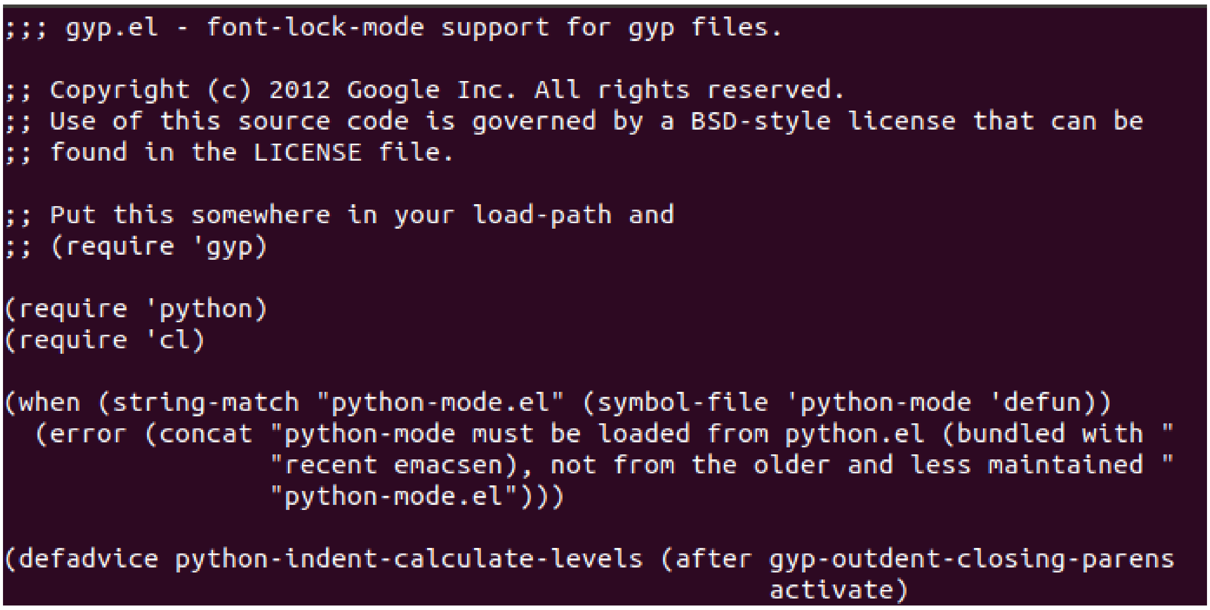
16. less Command

The less command is similar to the more command. It also includes some extra features such as 'adjustment in width and height of the terminal.' Comparatively, the more command cuts the output in the width of the terminal.

Syntax:

1. less <file name>

Output:



```
;;; gyp.el - font-lock-mode support for gyp files.

;; Copyright (c) 2012 Google Inc. All rights reserved.
;; Use of this source code is governed by a BSD-style license that can be
;; found in the LICENSE file.

;; Put this somewhere in your load-path and
;; (require 'gyp)

(require 'python)
(require 'cl)

(when (string-match "python-mode.el" (symbol-file 'python-mode 'defun))
  (error (concat "python-mode must be loaded from python.el (bundled with "
                 "recent emacsen), not from the older and less maintained "
                 "python-mode.el")))

(defadvice python-indent-calculate-levels (after gyp-outdent-closing-parens
                                                 activate)
```

Linux User Commands

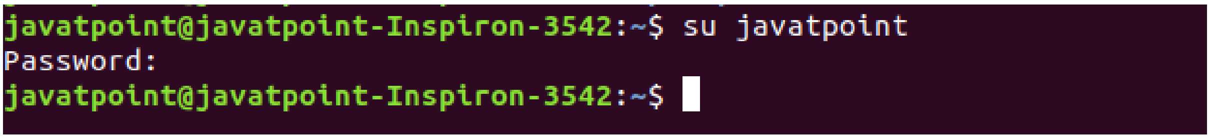
17. su Command

The su command provides administrative access to another user. In other words, it allows access of the Linux shell to another user.

Syntax:

1. su <**user** name>

Output:



```
javatpoint@javatpoint-Inspiron-3542:~$ su javatpoint
Password:
javatpoint@javatpoint-Inspiron-3542:~$ 
```

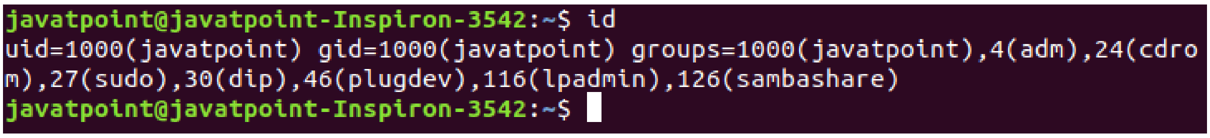
18. id Command

The id command is used to display the user ID (UID) and group ID (GID).

Syntax:

1. id

Output:



```
javatpoint@javatpoint-Inspiron-3542:~$ id
uid=1000(javatpoint) gid=1000(javatpoint) groups=1000(javatpoint),4(adm),24(cdro
m),27(sudo),30(dip),46(plugdev),116(lpadmin),126(sambashare)
javatpoint@javatpoint-Inspiron-3542:~$ 
```

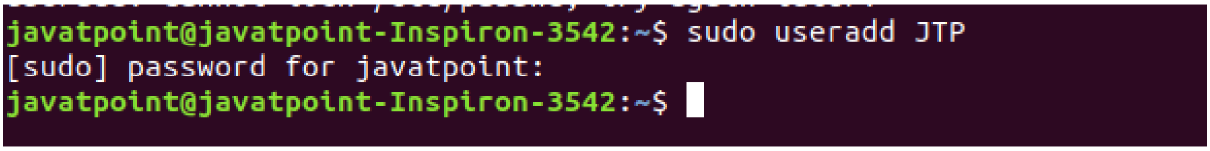
19. useradd Command

The useradd command is used to add or remove a user on a Linux server.

Syntax:

1. useradd username

Output:



```
javatpoint@javatpoint-Inspiron-3542:~$ sudo useradd JTP
[sudo] password for javatpoint:
javatpoint@javatpoint-Inspiron-3542:~$ 
```

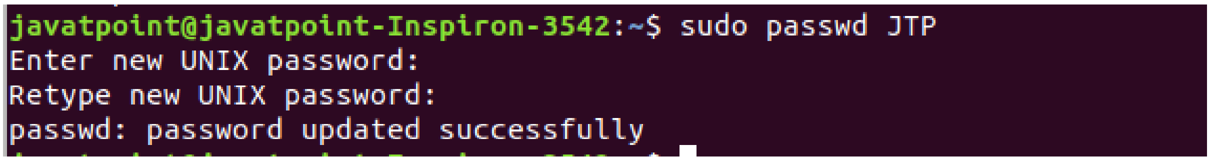
20. passwd Command

The passwd command is used to create and change the password for a user.

Syntax:

1. passwd **<username>**

Output:



```
javatpoint@javatpoint-Inspiron-3542:~$ sudo passwd JTP
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
```

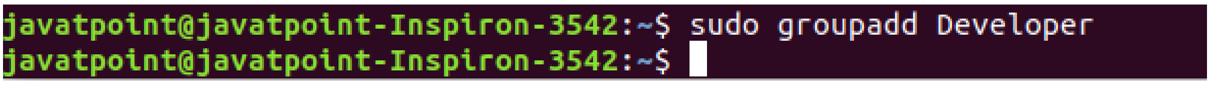
21. groupadd Command

The groupadd command is used to create a user group.

Syntax:

1. groupadd **<group** name**>**

Output:



```
javatpoint@javatpoint-Inspiron-3542:~$ sudo groupadd Developer
javatpoint@javatpoint-Inspiron-3542:~$ 
```

Linux Filter Commands

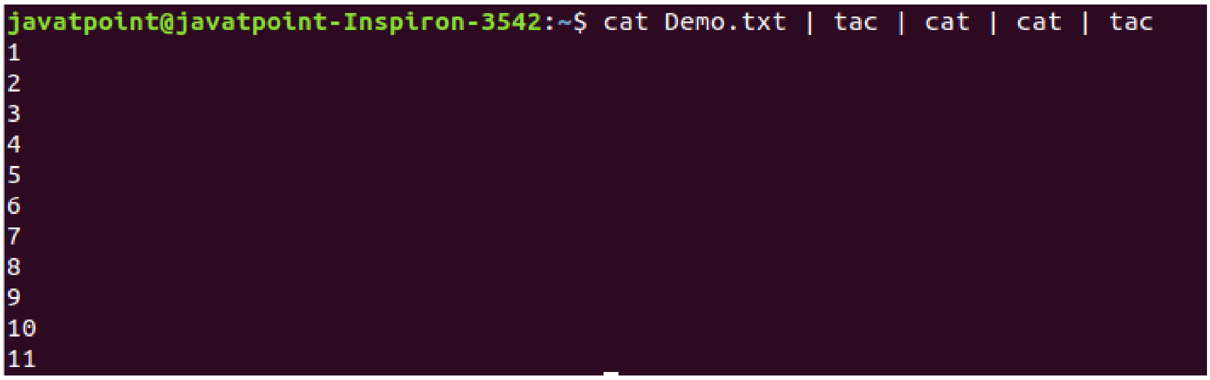
22. cat Command

The cat command is also used as a filter. To filter a file, it is used inside pipes.

Syntax:

1. cat **<fileName>** | cat or tac | cat or tac |. . .

Output:



```
javatpoint@javatpoint-Inspiron-3542:~$ cat Demo.txt | tac | cat | cat | tac
1
2
3
4
5
6
7
8
9
10
11
```

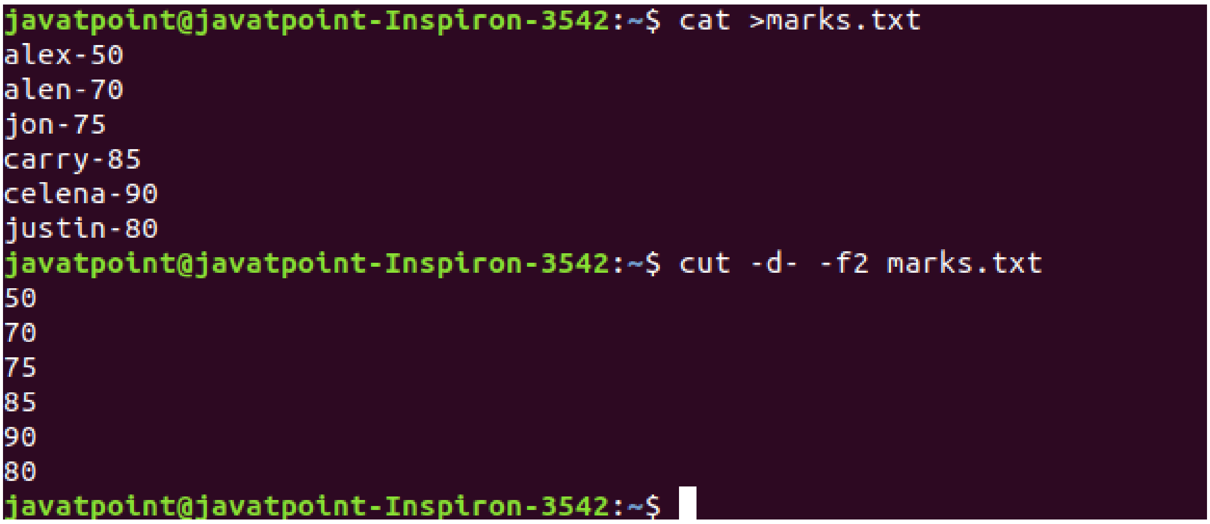
23. cut Command

The cut command is used to select a specific column of a file. The '-d' option is used as a delimiter, and it can be a space (' '), a slash (/), a hyphen (-), or anything else. And, the '-f' option is used to specify a column number.

Syntax:

1. cut -d(delimiter) -f(columnNumber) **<fileName>**

Output:



```
javatpoint@javatpoint-Inspiron-3542:~$ cat >marks.txt
alex-50
alen-70
jon-75
carry-85
celena-90
justin-80
javatpoint@javatpoint-Inspiron-3542:~$ cut -d- -f2 marks.txt
50
70
75
85
90
80
javatpoint@javatpoint-Inspiron-3542:~$
```

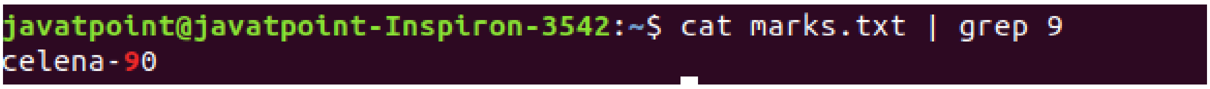
24. grep Command

The grep is the most powerful and used filter in a Linux system. The 'grep' stands for "**global regular expression print**." It is useful for searching the content from a file. Generally, it is used with the pipe.

Syntax:

1. command | grep **<searchWord>**

Output:



```
javatpoint@javatpoint-Inspiron-3542:~$ cat marks.txt | grep 9
celena-90
```

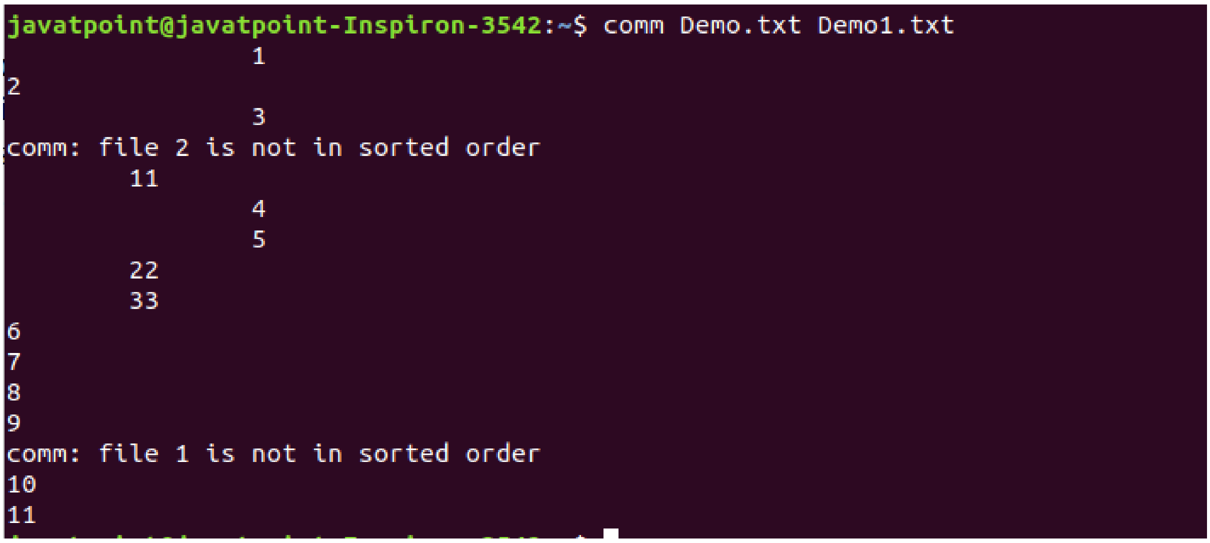
25. comm Command

The 'comm' command is used to compare two files or streams. By default, it displays three columns, first displays non-matching items of the first file, second indicates the non-matching item of the second file, and the third column displays the matching items of both files.

Syntax:

1. comm **<file1> <file2>**

Output:



```
javatpoint@javatpoint-Inspiron-3542:~$ comm Demo.txt Demo1.txt
                1
2
                3
comm: file 2 is not in sorted order
        11
                4
                5
        22
        33
6
7
8
9
comm: file 1 is not in sorted order
10
11
```

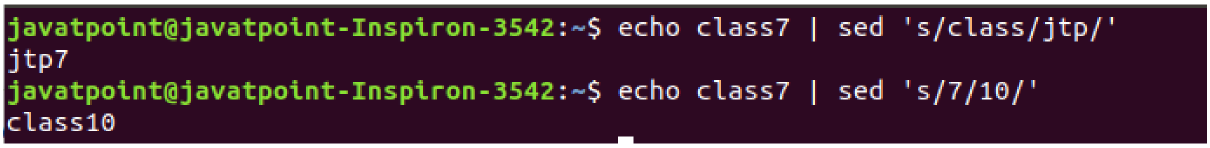
26. sed command

The sed command is also known as **stream editor**. It is used to edit files using a regular expression. It does not permanently edit files; instead, the edited content remains only on display. It does not affect the actual file.

Syntax:

```
1.  command | sed 's/<oldWord>/<newWord>/'
```

Output:



```
javatpoint@javatpoint-Inspiron-3542:~$ echo class7 | sed 's/class/jtp/'
jtp7
javatpoint@javatpoint-Inspiron-3542:~$ echo class7 | sed 's/7/10/'
class10
```

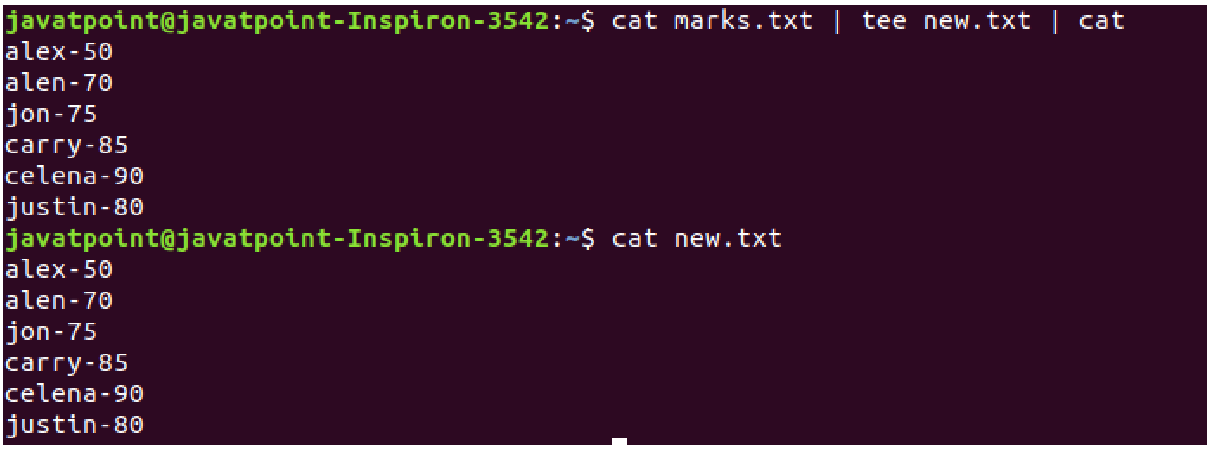
27. tee command

The tee command is quite similar to the cat command. The only difference between both filters is that it puts standard input on standard output and also write them into a file.

Syntax:

```
1.  cat <fileName> | tee <newFile> |  cat or tac |.....
```

Output:



```
javatpoint@javatpoint-Inspiron-3542:~$ cat marks.txt | tee new.txt | cat
alex-50
alen-70
jon-75
carry-85
celena-90
justin-80
javatpoint@javatpoint-Inspiron-3542:~$ cat new.txt
alex-50
alen-70
jon-75
carry-85
celena-90
justin-80
```

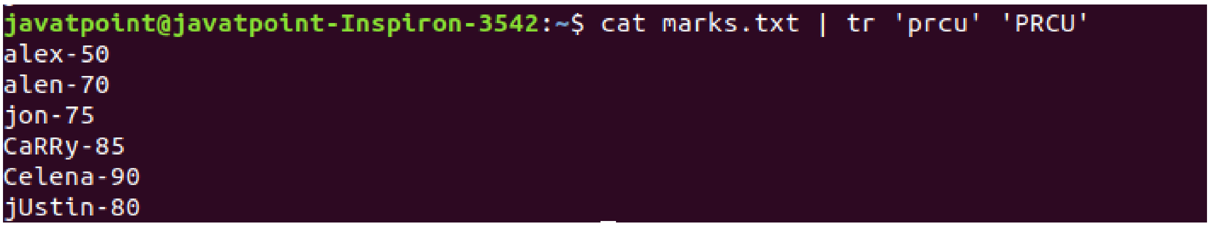
28. tr Command

The tr command is used to translate the file content like from lower case to upper case.

Syntax:

1. command | tr <'old'> <'new'>

Output:



```
javatpoint@javatpoint-Inspiron-3542:~$ cat marks.txt | tr 'prcu' 'PRCU'
alex-50
alen-70
jon-75
CaRRy-85
Celena-90
jUstin-80
```

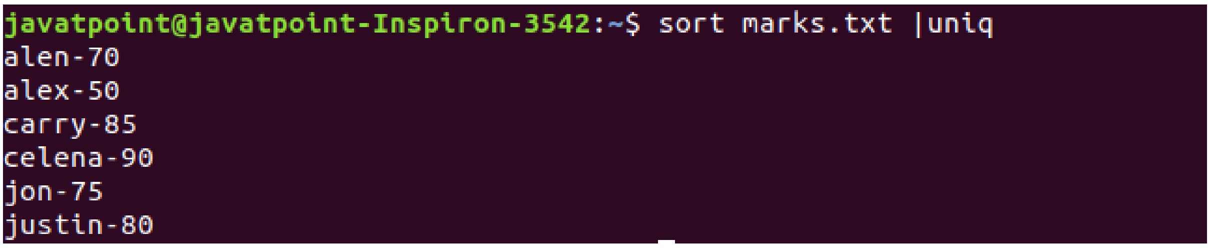
29. uniq Command

The uniq command is used to form a sorted list in which every word will occur only once.

Syntax:

1. command <fileName> | uniq

Output:



```
javatpoint@javatpoint-Inspiron-3542:~$ sort marks.txt |uniq
alen-70
alex-50
carry-85
celena-90
jon-75
justin-80
```

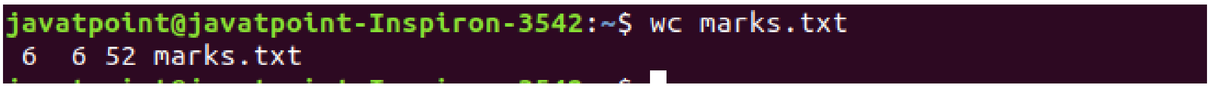
30. wc Command

The wc command is used to count the lines, words, and characters in a file.

Syntax:

1. wc <file name>

Output:



```
javatpoint@javatpoint-Inspiron-3542:~$ wc marks.txt
 6  6 52 marks.txt
```

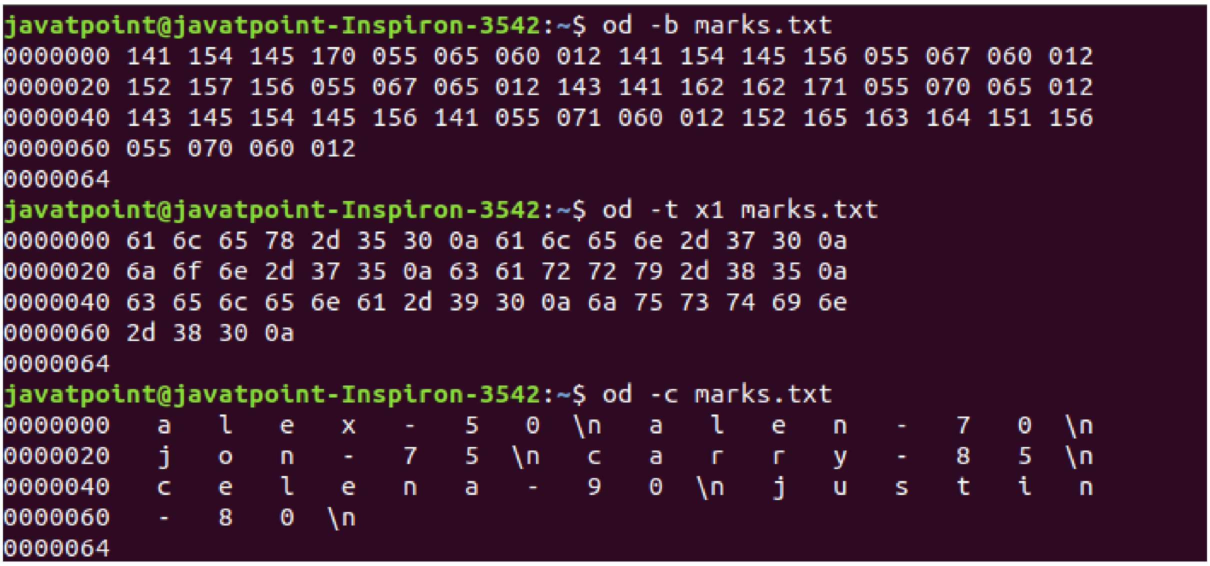
31. od Command

The od command is used to display the content of a file in different s, such as hexadecimal, octal, and ASCII characters.

Syntax:

1. od -b <fileName>   // Octal format
2. od -t x1 <fileName>   // Hexa decimal format
3. od -c <fileName>   // ASCII character format

Output:



```
javatpoint@javatpoint-Inspiron-3542:~$ od -b marks.txt
0000000 141 154 145 170 055 065 060 012 141 154 145 156 055 067 060 012
0000020 152 157 156 055 067 065 012 143 141 162 162 171 055 070 065 012
0000040 143 145 154 145 156 141 055 071 060 012 152 165 163 164 151 156
0000060 055 070 060 012
0000064
javatpoint@javatpoint-Inspiron-3542:~$ od -t x1 marks.txt
0000000 61 6c 65 78 2d 35 30 0a 61 6c 65 6e 2d 37 30 0a
0000020 6a 6f 6e 2d 37 35 0a 63 61 72 72 79 2d 38 35 0a
0000040 63 65 6c 65 6e 61 2d 39 30 0a 6a 75 73 74 69 6e
0000060 2d 38 30 0a
0000064
javatpoint@javatpoint-Inspiron-3542:~$ od -c marks.txt
0000000   a   l   e   x   -   5   0  \n   a   l   e   n   -   7   0  \n
0000020   j   o   n   -   7   5  \n   c   a   r   r   y   -   8   5  \n
0000040   c   e   l   e   n   a   -   9   0  \n   j   u   s   t   i   n
0000060   -   8   0  \n
0000064
```

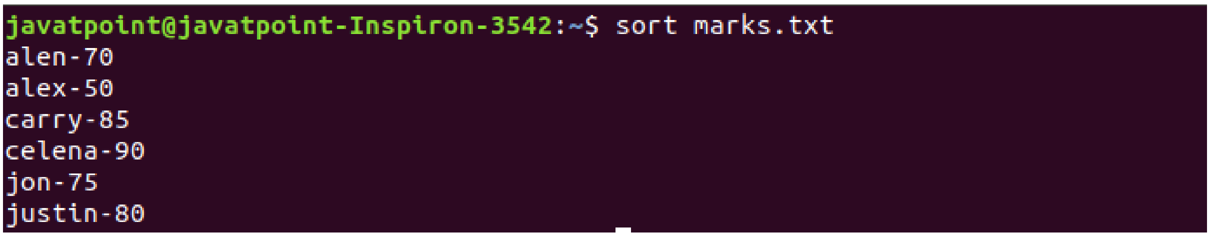
32. sort Command

The sort command is used to sort files in alphabetical order.

Syntax:

1. sort <file name>

Output:



```
javatpoint@javatpoint-Inspiron-3542:~$ sort marks.txt
alen-70
alex-50
carry-85
celena-90
jon-75
justin-80
```

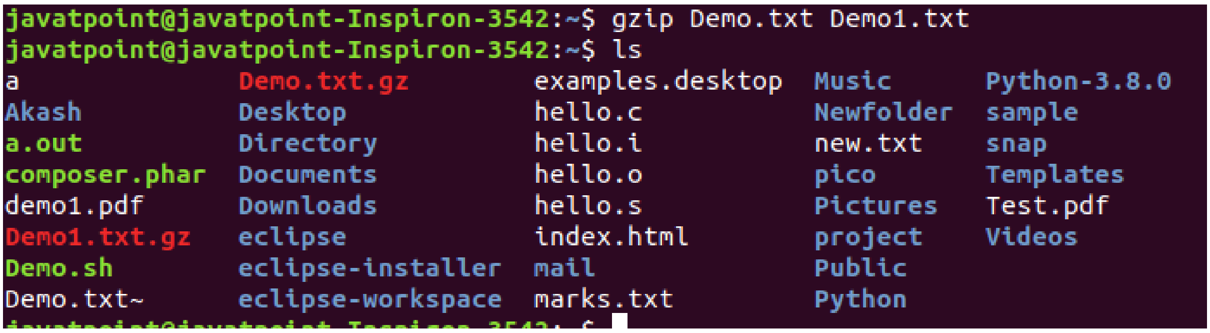
33. gzip Command

The gzip command is used to truncate the file size. It is a compressing tool. It replaces the original file by the compressed file having '.gz' extension.

Syntax:

1. gzip <file1> <file2> <file3>...

Output:



```
javatpoint@javatpoint-Inspiron-3542:~$ gzip Demo.txt Demo1.txt
javatpoint@javatpoint-Inspiron-3542:~$ ls
a              Demo.txt.gz        examples.desktop  Music      Python-3.8.0
Akash          Desktop            hello.c           Newfolder  sample
a.out          Directory          hello.i           new.txt    snap
composer.phar  Documents          hello.o           pico       Templates
demo1.pdf      Downloads          hello.s           Pictures   Test.pdf
Demo1.txt.gz   eclipse            index.html        project    Videos
Demo.sh        eclipse-installer  mail              Public
Demo.txt~      eclipse-workspace  marks.txt         Python
javatpoint@javatpoint-Inspiron-3542:~$
```

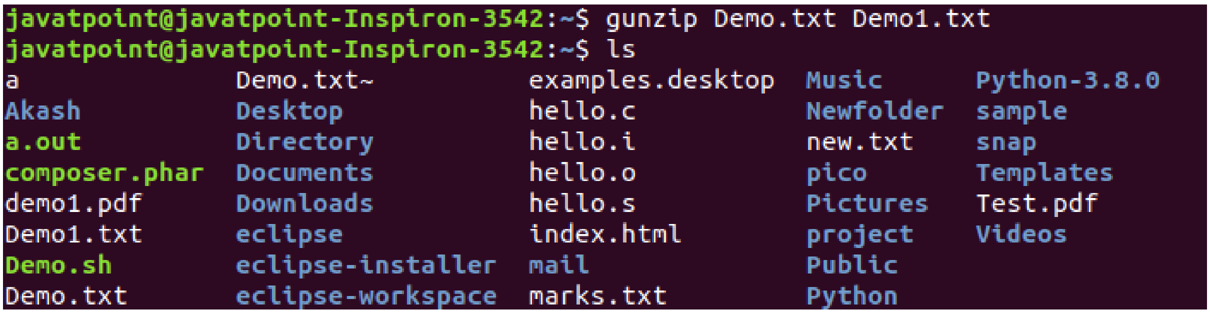
34. gunzip Command

The gunzip command is used to decompress a file. It is a reverse operation of gzip command.

Syntax:

1. gunzip **<file1> <file2> <file3>**. .

Output:



```
javatpoint@javatpoint-Inspiron-3542:~$ gunzip Demo.txt Demo1.txt
javatpoint@javatpoint-Inspiron-3542:~$ ls
a              Demo.txt~          examples.desktop  Music      Python-3.8.0
Akash          Desktop            hello.c           Newfolder  sample
a.out          Directory          hello.i           new.txt    snap
composer.phar  Documents          hello.o           pico       Templates
demo1.pdf      Downloads          hello.s           Pictures   Test.pdf
Demo1.txt      eclipse            index.html        project    Videos
Demo.sh        eclipse-installer  mail              Public
Demo.txt       eclipse-workspace  marks.txt         Python
```