Linux Commands

date Command	
Format Specifiers	
date	Display today's date as **Friday 21 January 2022 09:20:22 AM IST**
date +%D	Display date as **mm/dd/yy**
date +%d	Display the day of the month (**01** to **31**)
date +%a	Displays the abbreviated name for weekdays (**Sun** to **Sat**)
date +%A	Displays full weekdays (**Sunday** to **Saturday**)
date +%m	Displays the month of year (**01** to **12**)
date +%h	Displays abbreviated month name (**Jan** to **Dec**)
date +%b	Displays abbreviated month name (**Jan** to **Dec**)
date +%B	Displays full month name (**January** to **December**)
date +%y	Displays last two digits of the year (**00** to **99**).
date +%Y	Display four-digit year (Ex. **2022**)
date +%T	Display the time in 24 hour format as **HH:MM:SS**
date +%H	Display the hour
date +%M	Display the minute
date +%S	Display the seconds
-s or -set option to set the system date and time	
date --set "Wed Nov 14 15:23:34 PDT 2018" → Error due to user role	
sudo date --set "Wed Nov 14 15:23:34 PDT 2018"	
--date option for displaying future dates	
date --date "Next Sunday" → Date of next Sunday	
date --date "2 Days" → Date after 2 days	
date --date "2 Years" → Date after 2 years	
date --date "13 Months" → Date after 13 months	
--date option for displaying past dates	
date --date "10 Days Ago" → Date of 10 Days Ago	
date --date "10 Months Ago" → Date of 10 Months Ago	
date --date "10 Years Ago" → Date of 10 Years Ago	
date --date "Yesterday" → Date of Yesterday	

time Command

time command is used to execute a command and prints a summary of real-time, user CPU time and system CPU time spent by executing a command when it terminates.

'real' time	It is the elapsed wall clock time taken by a command to get executed.
'user' time	Number of CPU seconds that command uses in user mode.
'sys' time	Number of CPU seconds that command uses in kernel mode.

cal Command

cal command is a calendar command which is used to see the calendar of a specific month or a whole year.

cal	Shows current month calendar on the terminal with the current date highlighted.
cal -y	Shows the calendar of the complete current year with the current date highlighted.
cal 03 2022	Shows the calendar of month March of 2022.
cal 2021	Shows the calendar of year 2021.
cal -3	Shows calendar of previous, current and next month.
cal -A4	Shows calendar of **current month** and **next four month**.
cal -B4	Shows calendar of **previous four months** and **current month**.
cal -y -B2 -A2	Shows calendar of **all month of current year, last two months of previous year** and **two months of next year**. In other words, Last Year's Nov/Dec to Next Year Jan/Feb

echo Command

It is used for displaying lines of text or string which are passed as arguments on the command line.

It can be used for creating file also.

$ x=10

$ y=15

$ echo $x → 10

$ echo "Value of x = $x" → Value of x = 10

$ echo "Sum of $x and $y is $(($x+$y))" → Sum of 10 and 15 is 25

```
sps@SPS:~$ echo "Welcome" >Test
sps@SPS:~$ ls
Desktop    Downloads  Pictures  Templates  Videos
Documents  Music      Public    Test
```

printf Command

It is used for to display the given string, number or any other format specifier on the terminal window.

$ x=10

$ y=15

$ printf “%d” $x → 10

$ printf “Value of x = %d” $x → Value of x = 10

$ printf “Sum of %d and %d is %d” $x $y $(($x+$y))” → Sum of 10 and 15 is 25

bc Command

bc is an acronym for “basic calculator.”

$ bc

3+2

5

3/2

1

Scale=2

3/2

1.50

quit → To Exit

Combination of echo, bc, and pipe:

echo “1<=2” | bc

1 (TRUE)

echo "12+5" | bc

17

echo "10^2" | bc

100

passwd Command

passwd command is used to change the user account passwords. The root user reserves the privilege to change the password for any user on the system, while a normal user can only change the account password for his/her own account.

$ passwd

```
sps@SPS:~$ passwd
Changing password for sps.
Current password:
New password:
Retype new password:
You must choose a longer password
```

who Command

who command is mainly used to find out users' information.

$ who -H

```
sps@SPS:~$ who -H
NAME     LINE         TIME             COMMENT
sps      :0           2022-01-20 18:57 (:0)
```

NAME → User ID

LINE → Device Name

TIME → Time of Login

COMMENT → Machine Name

whoami Command

It displays the username of the current user when this command is invoked.

$ whoami

sps

$SHELL Variable

$SHELL variable represent the current shell of system.

$ echo $SHELL

/bin/bash

$ cat /etc/shells

Lists all shells installed

$ chsh

To change shell

which Command

which command is used to locate the executable file associated with the given command by searching it in the path environment variable.

$ which echo

/user/bin/echo

$ which who

/user/bin/who

$ which passwd

/user/bin/passwd

uname Command

It returns Name of OS, Version of OS, machine name (host name), etc.

```
sps@SPS:~$ uname
Linux
sps@SPS:~$ uname -r
5 Files -43-generic
sps@SPS:~$ uname -n
SPS
```

hostname Command

It returns Host Name (Machine Name).

```
sps@SPS:~$ hostname
SPS
```

tty Command

It returns Tele Type (Terminal Name).

```
sps@SPS:~$ tty
/dev/pts/0
```

ps Command

It lists all the processes which are create by the users.

```
sps@SPS:~$ ps
    PID TTY          TIME CMD
   2651 pts/0    00:00:00 bash
   2741 pts/0    00:00:00 ps
```

pwd Command

It returns the present working directory.

```
sps@SPS:~$ pwd
/home/sps
```

cat Command

The cat command displays the contents of files.

```
sps@SPS:~$ echo "Welcome" >Test
sps@SPS:~$ ls
Desktop    Downloads  Pictures  Templates  Videos
Documents  Music      Public    Test
sps@SPS:~$ cat Test
Welcome
sps@SPS:~$
```

File (Test) created with echo. The contents can be read with cat.

cat /etc/shells → It lists all shells installed.

chsh → It allows to change the shell. After giving password, $ /bin/bash

ls Command

It lists directory contents of files and directories.

```
sps@SPS:~$ pwd
/home/sps
sps@SPS:~$ ls
Desktop    Downloads  Pictures  Templates  Videos
Documents  Music      Public    Test
sps@SPS:~$
```

ls -t

It sorts the files/directories by modification time, showing the last edited file/directory first.

```
sps@SPS:~$ ls -t
Test       Downloads  Pictures  Templates  Desktop
Documents  Music      Public    Videos
sps@SPS:~$
```

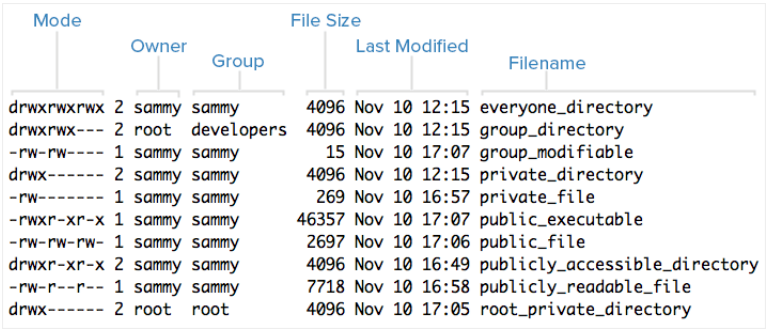
ls -1

Display one file/directory per line.

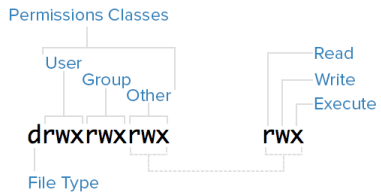
```
sps@SPS:~$ ls -1
Desktop
Documents
Downloads
Music
Pictures
Public
Templates
Test
Videos
sps@SPS:~$
```

ls -l

Display all information about files/directories.



--



ls -l \Directory

It will list all the contents of **given directory name**.

ls -F

Identify directories and executable.

```
sps@SPS:~$ ls -F
Desktop/    Downloads/  Pictures/  Templates/  Videos/
Documents/  Music/      Public/    Test
```

/ –> directory.

nothing –> normal file.

@ –> link file.

*** –> Executable file**

ls ~

Lists contents of home directory.

ls -a	Lists hidden files/directories also.
ls -r	Lists contents in reverse order.

ls ../	Contents of parent directory:
	`sps@SPS:~$ ls ../`
	`sps`

ls -R	Lists contents of sub-directories also.
	`sps@SPS:~$ ls -R`
	`.:`
	`Desktop  Downloads  Pictures  Templates  Videos`
	`Documents  Music  Public  Test`
	`./Desktop:`
	`./Documents:`
	`./Downloads:`
	`./Music:`
	`./Pictures:`
	`./Public:`
	`./Templates:`
	`./Videos:`

ls -ltr	To sort the file names in the last modification time in reverse order. This will be showing the last edited file in the last line which will be handy when the listing goes beyond a page.

cd Command	
It is used to change the current directory.	

cd ..	It changes the current directory to **one step before**.
	`sps@SPS:~$ pwd`
	`/home/sps`
	`sps@SPS:~$ cd ..`
	`sps@SPS:/home$ pwd`
	`/home`
	`sps@SPS:/home$ cd ..`
	`sps@SPS:/$ pwd`
	`/`
	`sps@SPS:/$`

cd ../..	It changes the current directory to **two steps before**.
	`sps@SPS:~$ pwd`
	`/home/sps`
	`sps@SPS:~$ cd ../..`
	`sps@SPS:/$ pwd`
	`/`
	`sps@SPS:/$`

cd	It brings to **home directory**.
	`sps@SPS:/$ cd`
	`sps@SPS:~$ pwd`
	`/home/sps`
	`sps@SPS:~$`

cd ./d1	It is equivalent to **cd d1**. →./ refers to current directory.

mkdir Command

It is used to create directory.

mkdir Test	It creates a directory with name **Test**.
	`sps@SPS:~$ pwd`
	`/home/sps`
	`sps@SPS:~$ mkdir Test`
	`sps@SPS:~$ ls`
	`Desktop  Downloads  Pictures  Templates  Videos`
	`Documents  Music  Public  Test`
	`sps@SPS:~$`

mkdir d1 d2 d3	It creates three directories named as **d1**, **d2** and **d3**.
	`sps@SPS:~$ cd Test`
	`sps@SPS:~/Test$ mkdir d1 d2 d3`
	`sps@SPS:~/Test$ ls`
	`d1  d2  d3`
	`sps@SPS:~/Test$`

mkdir d4 d4/d5	It creates directory **d4** and subdirectory **d5** under **d4**.
mkdir d7/d8 d7	It gives error because **d7** is not created before creation of **d8**.

rmdir Command

It is used to delete the directory.

rmdir d2	It deletes a directory with name **d2**.
	`sps@SPS:~/Test$ rmdir d2`
	`sps@SPS:~/Test$`

rmdir d1/d4 d1	It deletes sub-directory named as **d4** and then it deletes **d1**.
	`sps@SPS:~/Test$ rmdir d1/d4 d1`
	`sps@SPS:~/Test$`

mkdir d4 d4/d5	It creates directory **d4** and subdirectory **d5** under **d4**.
mkdir d7/d8 d7	It gives error because **d7** is not created before creation of **d8**.

mv Command

It is used to rename a directory or file.

mv oldname newname
mv Test TEST
	It change the directory name **Test** to **TEST**.
	`sps@SPS:~$ ls`
	`Desktop  Downloads  Pictures  Templates  Videos`
	`Documents  Music  Public  Test`
	`sps@SPS:~$ mv Test TEST`
	`sps@SPS:~$ ls`
	`Desktop  Downloads  Pictures  Templates  Videos`
	`Documents  Music  Public  TEST`
	`sps@SPS:~$ echo "Test File" >Test`
	`sps@SPS:~$ ls`
	`Desktop  Downloads  Pictures  Templates  TEST`
	`Documents  Music  Public  Test  Videos`
	`sps@SPS:~$ mv Test TEst`
	`sps@SPS:~$ ls`
	`Desktop  Downloads  Pictures  Templates  TEST`
	`Documents  Music  Public  TEst  Videos`

cp Command	
It is used to copy a file/directory to a new directory. It is also used to copy the contents of a file to other file.	
cp Test TEST	It copies the **Test** file into **TEST** directory, and maintains a copy of **Test** file in PWD.
cp -r dir Dir	It copies the **dir** directory into **Dir** directory.
cp A B	It copies the contents of **file A** to **file B**.

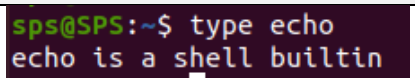
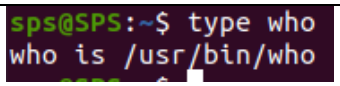
rm Command	
It is used to remove/delete a file.	
rm Test	It deletes a file named as **Test**.

cat Command	
It is used to create file, display contents of file and concatenation of two files.	
cat Test	It lists the contents of file **Test**.

cat A B	It concatenates the output of file **A** and **B**.
	`sps@SPS:~$ echo "Hello" >A`
	`sps@SPS:~$ echo "Welcome" >B`
	`sps@SPS:~$ cat A`
	`Hello`
	`sps@SPS:~$ cat B`
	`Welcome`
	`sps@SPS:~$ cat A B`
	`Hello`
	`Welcome`
	`sps@SPS:~$`

cat >A	It creates a file named as **A**. Aftering entering the text, we need to press: **Ctrl + D** to exit from input mode.
	`sps@SPS:~$ cat >A`
	`Hello`
	`How are you Sir.`
	`I am good.`
	`sps@SPS:~$`
	`sps@SPS:~$ cat A`
	`Hello`
	`How are you Sir.`
	`I am good.`
	`sps@SPS:~$`

cat file1 > file2	It will copy the contents of **file1** to **file2** using the (>) operator. If **file2** does not exist, it will create the **file2**.
cat file1 >> file2	It will append the contents of **file1** to **file2**.

type Command	
It is used to identify whether the command is INTERNAL or EXTERNAL?	
type echo	
type who	

man (MANUAL) Command	
man command is used to display the user manual of any command that we can run on the terminal.	
man who	It will provide the manual of **who**.