

Linux commands - Basics

1. pwd - present working directories

- Path in linux - /users/ravi/Desktop/learning
- Note : here (/) means Root directory
- It shows you in which location you are working in the console.
- The output of pwd command is known as a current location

2. ls - shows all files and directories at current location.

3. mkdir - use to create a directory in current location

- Mkdir directory_name
- Eg : mkdir test

4. rmdir - use to remove the directories

- Syntax → rmdir directory_name
- Eg : rmdir test
- Note : Delete only empty directories

5. cd - Use to change the directory

- Path syntax → cd /root/regex/ravi
- Here : Regex is directory and ravi another directory into regex
- To delete current directory we have to left it and then can be delete
- cd .. → Change the one previous directory
- cd - To exchange with previous directory

6. rm -rf → To delete both files and directories recursively even if they are empty or not.

- `rm -rf ravi`
- `-r` denotes recursively deletion of file and directory
- Here : ravi directory contains ravi and rahul directory so it is not empty

OPTIONS :-

Remove (unlink) the FILE(s).

`-f, --force`

ignore nonexistent files and arguments, never prompt

`-i` prompt before every removal

`-I` prompt once before removing more than three files, or when removing recursively; less intrusive than `-i`, while still giving protection against most mistakes

`--interactive[=WHEN]`

prompt according to WHEN: never, once (`-I`), or always (`-i`); without WHEN, prompt always

`--one-file-system`

when removing a hierarchy recursively, skip any directory that is on a file system different from that of the corresponding command line

argument

`--no-preserve-root`

do not treat `'/'` specially

`--preserve-root[=all]`

do not remove '/' (default); with 'all', reject any command line argument on a separate device from its parent

-r, -R, --recursive

remove directories and their contents recursively

-d, --dir

remove empty directories

-v, --verbose

explain what is being done

7. touch : Use to create the files

8. man command_name : It shows you the description of the given command.

NOTE :- (-v) are use to give description of the files and directories.

Assignment 01

1. Create a folder in the current user location name by regex , Now make another three folders in the regex folder as : learning , assignment . Without using the cd command.

- Make file into learning → data.txt , abc.txt
- From current location delete data.txt

- Now at regex location make folder name by linux into assignment and delete completely it by rm from regex location

2. Copy the abc.txt file into the regex folder

Answers of the following tasks :

TASK 1

```
[root@localhost ~]# mkdir regex
[root@localhost ~]# cd regex
[root@localhost regex]# mkdir learning assignment
[root@localhost regex]# touch /root/regex/learning/data.txt
[root@localhost regex]# touch /root/regex/learning/abc.txt
[root@localhost regex]# ## deleting the file
[root@localhost regex]# rm -rf /root/regex/learning/data.txt
[root@localhost regex]# ## creating folder into assignment
[root@localhost regex]# mkdir /root/regex/assignment/linux
[root@localhost regex]# ## checking the folder
[root@localhost regex]# cd assignment
[root@localhost assignment]# ls
linux
[root@localhost assignment]# cd ..
[root@localhost regex]# ## now deleting this folder from regex
location
[root@localhost regex]# rm -rf assignment
```

TASK 2

Command - (cp) Are use to copy the file

Syntax - cp source_file destination_file

```
[root@localhost regex]# cp /root/regex/learning/abc.txt  
/root/regex
```

```
[root@localhost regex]# ls  
abc.txt assignment learning
```

2. Second session —

1. ll - Long list for any particular directories
2. lh - Human readable format / ls -h

Cp - cp source destination

- a. Eg : cp /root/rahul/pathan.txt /root/ravi
- b. Here : /root/rahul/pathan.txt is a source AND /root/ravi is a destination
- c. Deletion of pathan file from salman is :
- d. rm -rf /root/ravi/pathan.txt
- e. Cp .. → It denotes the present directory location where from we are coping

1. Another way for copy :

- It is also Known as Direct copy procedure
- cp pathan /root/ravi

2. Another way for copy :

- Cp filename ../rahul
- Here :- filename is name of file , (..) denotes file current position , And (/rahul) denotes the destination folder

Path Types :-

1. Relative path is a process of going its child directory to another child directly as :
Cd ../rahul

2. Absolute path - It is a process of indirect move from child to child and to parent by their location.
3. Move command :- It is same as copy command but it move command from one place to another not copy .

Assignment 2 :-

1. Create two folders in current user learning and assignment . Now make 50 files in the learning name by data1.txt , data2.txt and so on.
 - Now remove the file from data1.txt to data10.txt , and while deleting it should give prompt (warning)
 - Now copy data11.txt to data.25 from the learning folder to the assignment folder.
 - Now copy the learning folder completely and assign it into the assignment folder.
 - Now copy the data11.txt to learn and while copying the file name should be changed to user11.txt . (One command for these all)

- Now we have a file on /home location name by tushar.txt . Now move this file from /home to current user , while the name should be changed to file as tushargoyal.txt .
- Now rename tushargoyal.txt to tushardata.txt

Answer :-

1. Creating folder learning and assignment and then creating 50 files as :

```
[root@172 ~]# mkdir learning assignment  
[root@172 ~]# cd learning  
[root@172 learning]# touch data{1..50}.txt
```

2. To give warning while deleting file we use here (-i) option of the rm command as :

```
[root@172 learning]# rm -i data{1..10}.txt  
rm: remove regular empty file 'data1.txt'?  
rm: remove regular empty file 'data2.txt'?  
rm: remove regular empty file 'data3.txt'?  
rm: remove regular empty file 'data4.txt'?  
rm: remove regular empty file 'data5.txt'?  
rm: remove regular empty file 'data6.txt'?  
rm: remove regular empty file 'data7.txt'?  
rm: remove regular empty file 'data8.txt'?
```


rm: remove regular empty file 'data9.txt'?
rm: remove regular empty file 'data10.txt'?

3. Now copying files from 11 to 25 into the assignment directory.

```
[root@172 learning]# cp data{11..25}.txt /root/assignment
[root@172 learning]# ll /root/assignment
total 0
-rw-r--r--. 1 root root 0 May 23 13:04 data11.txt
-rw-r--r--. 1 root root 0 May 23 13:04 data12.txt
-rw-r--r--. 1 root root 0 May 23 13:04 data13.txt
-rw-r--r--. 1 root root 0 May 23 13:04 data14.txt
-rw-r--r--. 1 root root 0 May 23 13:04 data15.txt
-rw-r--r--. 1 root root 0 May 23 13:04 data16.txt
-rw-r--r--. 1 root root 0 May 23 13:04 data17.txt
-rw-r--r--. 1 root root 0 May 23 13:04 data18.txt
-rw-r--r--. 1 root root 0 May 23 13:04 data19.txt
-rw-r--r--. 1 root root 0 May 23 13:04 data20.txt
-rw-r--r--. 1 root root 0 May 23 13:04 data21.txt
-rw-r--r--. 1 root root 0 May 23 13:04 data22.txt
-rw-r--r--. 1 root root 0 May 23 13:04 data23.txt
-rw-r--r--. 1 root root 0 May 23 13:04 data24.txt
-rw-r--r--. 1 root root 0 May 23 13:04 data25.txt
```

4. Now copying the learning folder completely into the assignment folder.

```
[root@172 ~]# cp -r learning assignment
```

```
[root@172 assignment]# ls
```

- data11.txt data14.txt data17.txt data20.txt data23.txt **learning**
- data12.txt data15.txt data18.txt data21.txt data24.txt
- data13.txt data16.txt data19.txt data22.txt data25.txt

Here : learning is copied directory inside the assignment directory. And learning contains all their files.

5. Now copy the data11.txt to learn and while copying the file name should be changed to user11.txt .

```
[root@172 assignment]# cp data11.txt
```

```
/root/learning/user11.txt
```

```
[root@172 assignment]# ll /root/learning
```

```
-rw-r--r--. 1 root root 0 May 23 13:02 data4.txt  
-rw-r--r--. 1 root root 0 May 23 13:02 data50.txt  
-rw-r--r--. 1 root root 0 May 23 13:02 data5.txt  
-rw-r--r--. 1 root root 0 May 23 13:02 data6.txt  
-rw-r--r--. 1 root root 0 May 23 13:02 data7.txt  
-rw-r--r--. 1 root root 0 May 23 13:02 data8.txt  
-rw-r--r--. 1 root root 0 May 23 13:02 data9.txt
```

`-rw-r--r--. 1 root root 0 May 23 13:23 user11.txt`

Note :- Here you can see that the Red color line denotes that the name of the copying file changed and all files are there in the learning folder.

6. Now we have a file on /home location name by tushar.txt . Now move this file from /home to current user , while the name should be changed to file as tushargoyal.txt .

- Firstly creating a user name by Ravi as :
[root@172 home]# useradd ravi
- Now moving file into ravi user
[root@172 home]# ls
admin ravi
[root@172 home]# touch ravi.txt
[root@172 home]# ls
admin ravi.txt ravi
[root@172 home]# mv /home/ravi.txt
ravi/ravi_chandra.txt
[root@172 home]# cd ravi
[root@172 ravi]# ls
ravi_chandra.txt

7. Now rename tushargoyal.txt to tushardata.t

```
[root@172 royal]# touch tushargoyal.txt  
[root@172 royal]# ls  
tushargoyal.txt
```

```
[root@172 royal]# mv tushargoyal.txt
tushardata.txt
[root@172 royal]# ls
tushargoyal.txt
[root@172 royal]#
```

Wildcard character :-

Wildcard	Meaning
*	Matches any characters
?	Matches any single character
[characters]	Matches any character that is a member of the set characters
[!characters]	Matches any character that is not a member of the set characters
[[:class:]]	Matches any character that is a member of the specified class

Here :- star (*) → Zero or more characters

? : only 1 characters

Eg: ls a*

aman amazon.txt attir.txt azure.doc

Eg : ls a?

It shows that a name file which contains any two characters starting should be a then b.....

Eg : ls a?*

```
[root@172 rahul]# ls a?*
```

```
abc aman amazon.txt attir.txt azure.doc
```

Here : star takes all character after character a

Eg : Now finding those files which taked extension .txt then :

```
[root@172 rahul]# ls *.txt
```

```
amazon.txt attir.txt zebra.txt
```

Eg : Now range the Character files

```
[root@172 rahul]# ls [a-c]*
```

```
aabc abc aman amazon.txt attir.txt azure.doc
```

```
babluji chintu
```

Questions :-

1. Search for a file which has second letter is B
2. Search for those files which has at least three characters And file should end with doc
3. Find those files in which the second letter should be A and last is S.
4. Find those files which has letter (am) in sequence.
5. Find out those files which have 5 characters and the third character should be a or t .
6. Find those files which in range as :
 - C-z
 - E-t
 - 2,4,6,8
 - 2.txt

- But not as abc.txt

Answers :-

1. Search for a file which has second letter is b :

```
[root@172 royal]# ls
b babu baby bbaby bikaner tushargoyal.txt
[root@172 royal]# ls ?b*
bbaby
[root@172 royal]#
```

2. Search for those files which has at least three

```
[root@172 royal]# touch aaa.doc
[root@172 royal]# ls ???*.doc
aaa.doc
[root@172 royal]# touch abc.doc
[root@172 royal]# ls ???*.doc
aaa.doc abc.doc
[root@172 royal]#
```

3. Find those files in which the second letter should be A and last is S.

```
[root@172 royal]# ls
alls blas
[root@172 royal]# ls a?*s
alls
```

4. Find those files which have letter (am) in sequence.

```
[root@172 royal]# touch rvamc
```

```
[root@172 royal]# touch llamll
```

```
[root@172 royal]# ls *am*
```

```
llamll  rvamc
```

```
[root@172 royal]#
```

5. Find out those files which have 5 characters and the third character should be a or t .