Working with files

Working with files in linux involves creation , modification, identification, deletion , renaming and moving using some basic commands.

1. Creation

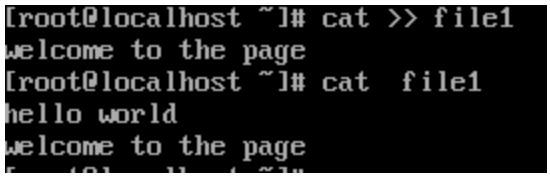
There are four ways to create a file.

- **cat**

cat > filename - it creates a file & if you again execute the same command the content is overwritten .

cat filename -used to read the content from the file.

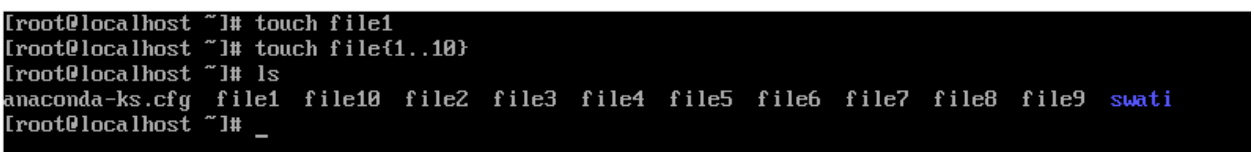
cat >> filename -to append the content in the file .



- **touch**

touch filename -used to create empty files.

touch filename{1..n} OR touch file1,file2,file3 - used to create multiple empty files.



touch .filename — to create hidden files

- **nano**

nano filename

- **vi**

Vi is a editor in which there are four modes

1. **Normal/default/command mode**
2. **Insertion mode**
3. **Visual mode**
4. **Execution mode**

syntax : vi filename

there are some commands to exit from the editor

:q- quit without saving

:q!-quit without saving forcefully

:w- save and stay in the file

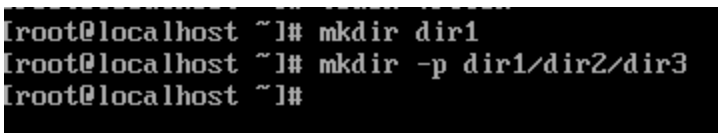
:wq or x -save &quit

:wq! or x!- save & quit forcefully

to create a directory

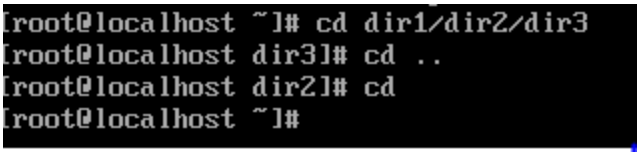
mkdir directoryname - to create a single directory

mkdir -p dir1/dir2/dir3 - to create multiple directories



cd - is used to change the directory

cd .. - is used to come one directory back from the present directory



File reading types

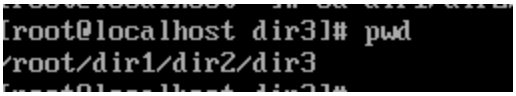
less filename - to read data top - bottom

tail filename - to read data bottom -top

head filename - to read first ten lines

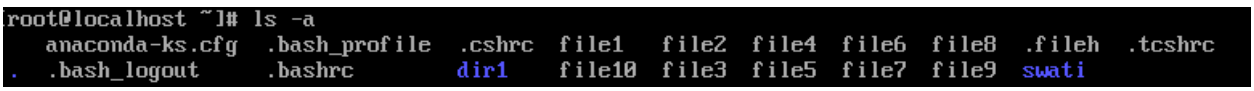
Basic commands:

1. pwd (print working directory) - it shows our current working directory



2. ls - to list the files and directories in the present directory .

3. ls -a to see all the files and directories with hidden files and directories



4. ll - long list - it displays a detailed information of file and directories.

```
[root@localhost ~]# ll
total 4
-rw-------. 1 root root 1425 Apr  6 06:52 anaconda-ks.cfg
drwxr-xr-x. 3 root root   18 Apr  6 11:37 dir1
-rw-r--r--. 1 root root    0 Apr  6 11:34 file1
-rw-r--r--. 1 root root    0 Apr  6 11:34 file10
-rw-r--r--. 1 root root    0 Apr  6 11:34 file2
-rw-r--r--. 1 root root    0 Apr  6 11:34 file3
-rw-r--r--. 1 root root    0 Apr  6 11:34 file4
-rw-r--r--. 1 root root    0 Apr  6 11:34 file5
-rw-r--r--. 1 root root    0 Apr  6 11:34 file6
-rw-r--r--. 1 root root    0 Apr  6 11:34 file7
-rw-r--r--. 1 root root    0 Apr  6 11:34 file8
-rw-r--r--. 1 root root    0 Apr  6 11:34 file9
drwxr-xr-x. 2 root root   21 Apr  6 06:55 swati
```

5. ll -d directoryname — to view a particular directory

```
[root@localhost ~]# ll -d dir1
drwxr-xr-x. 3 root root 18 Apr  6 11:37 dir1
[root@localhost ~]#
```

File Permission

file has read ,write and execute permission .

Permissions can be given in two ways

1) Alphabets

2) Numeric

r- read = 4

w-write = 2

x- execute =1

The permissions are given to the users, groups and other users . i.e ugo

Full permission	Default permission
dir - 777	dir -775
file -666	file -644

permission are given using

chmod

syntax : chmod ugo + rwx — filename to give full permission.

chmod 644 filename — to give full permission

```
[root@localhost ~]# chmod ugo+rwx file1
[root@localhost ~]# ll
total 4
-rw-------. 1 root root 1425 Apr  6 06:52 anaconda-ks.cfg
drwxr-xr-x. 3 root root   18 Apr  6 11:37 dir1
-rwxrwxrwx. 1 root root    0 Apr  6 11:34 file1
-rw-r--r--. 1 root root    0 Apr  6 11:34 file10
-rw-r--r--. 1 root root    0 Apr  6 11:34 file2
-rw-r--r--. 1 root root    0 Apr  6 11:34 file3
-rw-r--r--. 1 root root    0 Apr  6 11:34 file4
-rw-r--r--. 1 root root    0 Apr  6 11:34 file5
-rw-r--r--. 1 root root    0 Apr  6 11:34 file6
-rw-r--r--. 1 root root    0 Apr  6 11:34 file7
-rw-r--r--. 1 root root    0 Apr  6 11:34 file8
-rw-r--r--. 1 root root    0 Apr  6 11:34 file9
drwxr-xr-x. 2 root root   21 Apr  6 06:55 swati
[root@localhost ~]# chmod 644 file1
[root@localhost ~]# ll
total 4
-rw-------. 1 root root 1425 Apr  6 06:52 anaconda-ks.cfg
drwxr-xr-x. 3 root root   18 Apr  6 11:37 dir1
-rw-r--r--. 1 root root    0 Apr  6 11:34 file1
-rw-r--r--. 1 root root    0 Apr  6 11:34 file10
-rw-r--r--. 1 root root    0 Apr  6 11:34 file2
-rw-r--r--. 1 root root    0 Apr  6 11:34 file3
-rw-r--r--. 1 root root    0 Apr  6 11:34 file4
-rw-r--r--. 1 root root    0 Apr  6 11:34 file5
-rw-r--r--. 1 root root    0 Apr  6 11:34 file6
-rw-r--r--. 1 root root    0 Apr  6 11:34 file7
-rw-r--r--. 1 root root    0 Apr  6 11:34 file8
-rw-r--r--. 1 root root    0 Apr  6 11:34 file9
drwxr-xr-x. 2 root root   21 Apr  6 06:55 swati
```

we can change the owner of the file by using — chown

syntax : chown uname:gname filename

```
[root@localhost ~]# chown swati:root file1
[root@localhost ~]# ll
total 4
-rw-------. 1 root  root 1425 Apr  6 06:52 anaconda-ks.cfg
drwxr-xr-x. 3 root  root   18 Apr  6 11:37 dir1
-rw-r--r--. 1 swati root    0 Apr  6 11:34 file1
```

Identification

There are two file types

1. User defined — 1. normal (-) 2. directory (d) 3. link (l)

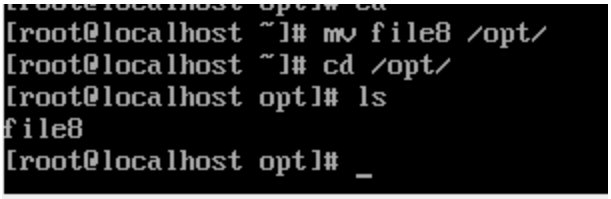
2. System defined — 1. block (b) 2. character (c) 3. socket (s) 4. pipe (p)

Moving and Renaming

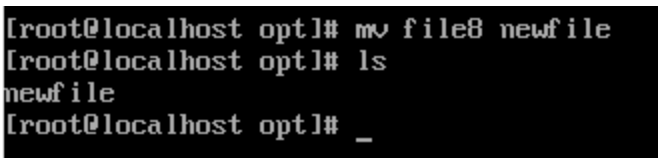
mv — is used to move a file from source to destination .The file is only present at the destination.

syntax : mv filename destination(path)

mv — is also used to rename a file



syntax : mv old_filename ./new_filename



cp — is used to copy the file .The is present at both the location.

syntax : cp filename destination(path)

```
[root@localhost ~]# cd /opt/
[root@localhost opt]# ls
file7  newfile
```

Deletion

rm — to remove a empty file

rm -rf — r- recursive f- forcefully deletes the file

rm -rvf — r-recursive f-forcefully v-verbose deletes the file

rm - -help —to known the rm related information.

```
[root@localhost ~]# rm  -rf dir1
[root@localhost ~]# ls
anaconda-ks.cfg  file10  swati
[root@localhost ~]# rm -r file10
rm: remove regular empty file ‘file10’? y
[root@localhost ~]# ls
anaconda-ks.cfg  swati
[root@localhost ~]#
```