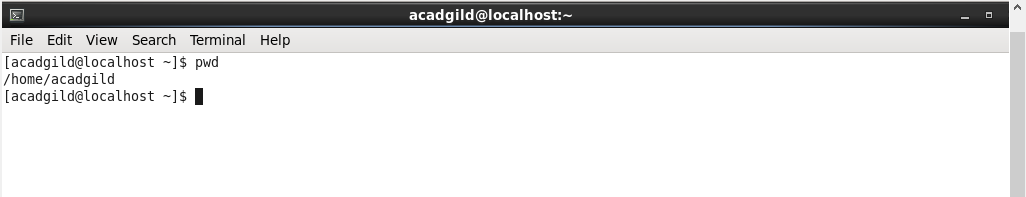
1. Pwd

Syntax: pwd

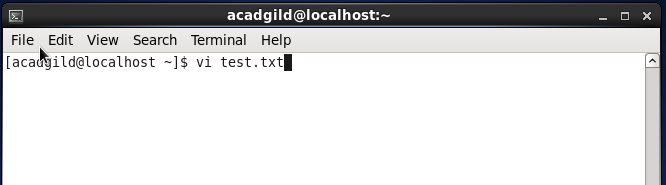
Used to display the path of current working directory



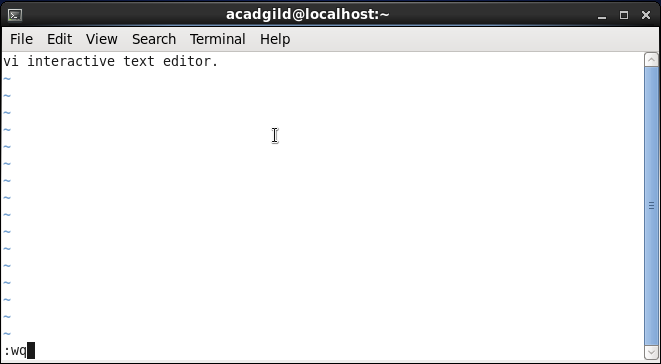
1. Vi

Syntax: vi <filename>

Used to invoke the interactive text editor.



Editor Invoked to enter text



1. Touch

Syntax: touch <filename>

To create an empty file of 0 bytes.



1. Mkdir
2. Syntax: mkdir directoryname

Used to create directory

1. Syntax:- mkdir -p d1/d2/d3/d4/…………

To create a directories & subdirectories recursively. –p is parent directory

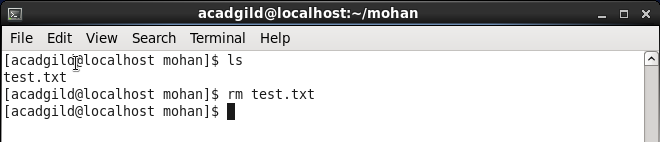


1. rm

a)Syntax:- rm filename

It is used for to remove a file/files

Snapshot:-



Here you can see the file test.txt is being deleted.

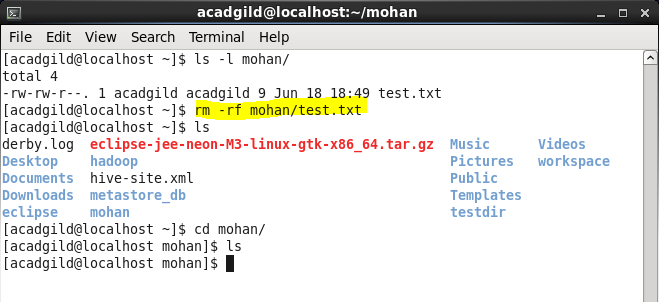
b)Syntax:- rm -rf < filename/directoryname >

-r recursively

-f forcefully

To remove files, directories and subdirectories Recursively or Forcefully

Snapshot:-



Here you can see the the directory which contain data inside will also get deleted by using this -rf along with rm.

c)Syntax:– rmdir < directoryname >

To remove an empty directory

Snapshot:-



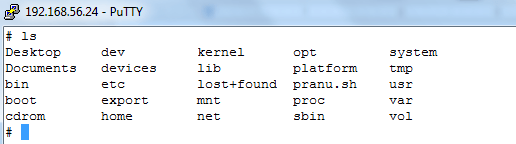
Here you can see directory named sasi is being deleted.

1. Ls

**a)**Syntax:**– ls**

1. This command is used to list all the files and directories.

**Snapshot:**

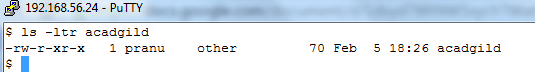


**All the files present in the current directory are shown once the ls command was typed in the terminal.**

**b) Syntax :-  ls -ltr filename**

command is used to list one file with name mentioned in command.

**Snapshot:**



**Here you can see the command gives full detail about the file acadgild.**

**c)**Syntax: –**ls –ltr**

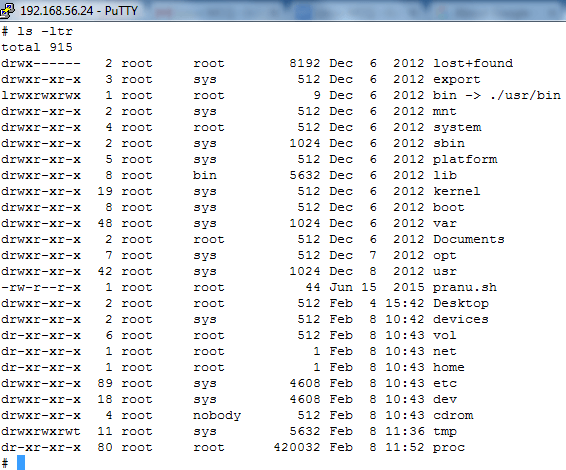
**-l long listing**

**-t** **time**

**r recursive**

Command is used to lists in the order of their creation time . list of files and directories contains permissions, links, ownership, group, date, time and file-name .

**Snapshot:-**



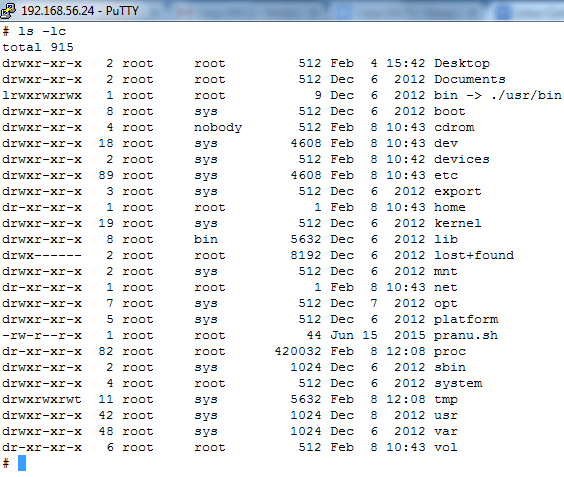
**Here you can see the list of all files present in the directory arranged  in order of creation . Different columns indicate permissions, links, ownership, group, date, time and file-name. For every file all these content are stored in computer by default .**

**d)**Syntax:-**ls –lc filename**

**-l long listing**

**-c** **character specific file**

**Snapshot:-**

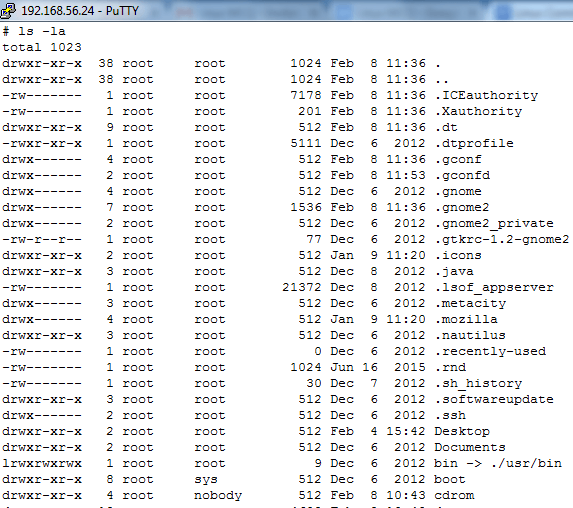


**Here you can see the listing of file in alphabetical order.**

**e)**Syntax:**– ls -la**

1. To display all hidden files. All Hidden files are start with DOT(.)

**Snapshot:-**



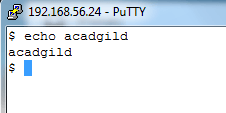
**Here you can see all the hidden files present in the current directory. We can notice all file is prefixed with .(dot) which indicates all file are system files, hidden from users.**

7. echo:

**a)**Syntax:**–**echo

It reflects back the content ,whatever written after echo command.

**Snapshot:-**

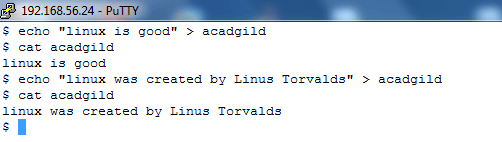


**Here you can see the comment written just after the command echo is printed again i.e anything following the echo command is displayed on the terminal.**

**b)**Syntax:-**echo “some content” > filename**

To insert text inside a file and will delete all previous data of the file.

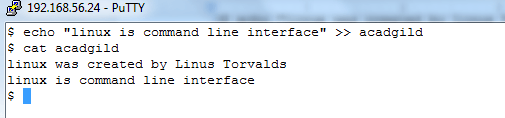
**Snapshot:-**



**Here you can see the statement is redirected to the file acadgild as content and later the content can be printed by cat command.**

**c)**Syntax:-**echo “some content” >> filename**

**Snapshot:-**



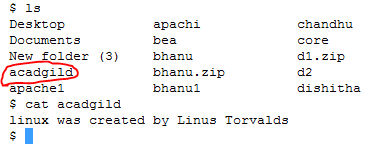
**Here you can see the statement is redirected to the file acadgild along with the previous data as content and later the content is printed by cat command.**

8. cat

 Syntax:-**cat filename**

To read a file

**Snapshot:-**



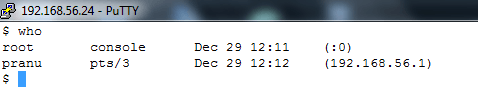
**Here you can see the cat command is used to through all the contents of file to the terminal**.

1. who

**a)**Syntax:-**who**

It shows the information about the users who are logged into the system currently.

**Snapshot:-**



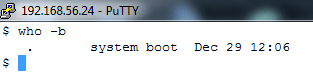
**Here you can see two user logged in named *root* and *pranu* with detail of what date, time and IP they used to log IN.**

**b)**Syntax:**who –b**

 -b  indicates the time and date of the last reboot.

To see when the server/system was started.

**Snapshot:-**



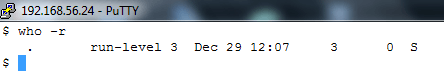
**Here you can see the last time when system started/booted was 29/12 at 12.06 pm.**

**c)**Syntax:-**who -r**

**-r indicates the current run level of the init process**

A run level is a state of init and the whole system that defines what system services are operating.

**Snapshot:-**

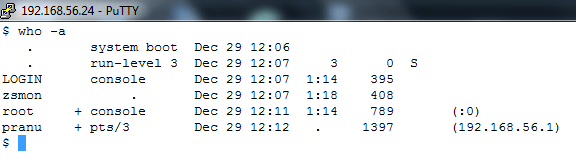


**Here you can see the run level is 3**

**d)**Syntax:-**who -a**

**-a lists processes, variables, users, run-levels**

**Snapshot:-**



**Here you can see all the *who* commands summary at glance.**

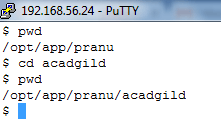
**Can be used in two ways, either way will result change in directory**

1. cd

**a)**Syntax:-**cd <directory name>**

Is  used to change directory

**Snapshot:-**

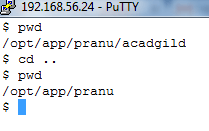


**Here you can see we entered inside acadgild directory.**

**b)**Syntax:-**cd ..**

This command is used to exit from a directory.

**Snapshot**:-



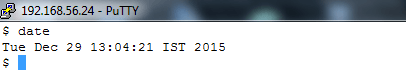
**Here you can see the command brings us one directory prior to where we was.**

1. date

Syntax:-**date**

**To display the current date and time of the system.**

**Snapshot:-**



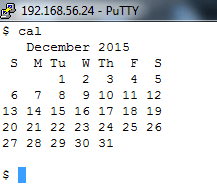
**Here you can see the current day,month,date and time followed by timing zone and year.**

12. cal

**a)**Syntax:-**cal**

It shows the present month calendar

**Snapshot:-**



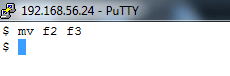
**Here you can see the calendar for a particular month.**

1. mv

Syntax:-**mv <sourcefile> <destinationfile>**

It is used to rename a file or move the file from one location to another location

**Snapshot:-**



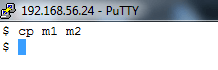
**Here you can see a file f2 is being moved with the new name f3, with all the contents of file f2 preserved in f3.**

14. cp

**a)**Syntax:-**cp <source file> <targetfile>**

To copy a file in same directory.

**Snapshot:-**

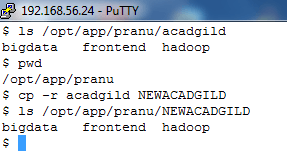


**Here you can see the file m1 content is copied to file m2.**

**b)**Syntax:-**cp -r :**

To copy all files and subdirectories recursively present inside a parent directory.

**Screenshot:-**



**Here you can see directories named big data, frontend and hadoop is copied when the parent directory is copied**

**c)**Syntax:-**cp -f :**

**-f** **Unlink. If a file descriptor for  a  destination  file**

**cannot be obtained, this option attempts to unlink the**

**destination file and proceed.**

**d)**Syntax:-**cp -i** :

**-i         Interactive. cp prompts for confirmation whenever  the**

**copy  would  overwrite  an existing target.**

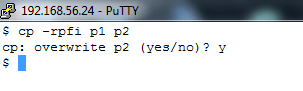
**e)**Syntax:-**cp -p** :

**-p preserve**

it is use to (keep the same detail record of original file)preserve the permission, time, date of source file to target file

**f)**Syntax:-**cp –rpfi <sourcefile> <destinationfile>**

**Snapshot:-**



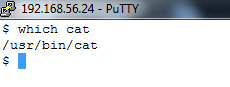
**Here you can see the the contents of p1 is forcefully pushed into p2 by deleting all the previous data of p2.**

15. which

Syntax:-**which <command-name>**

**It display  where the command is residing**

**Snapshot:-**



**Here you can see *cat* location is shown as output i.e, cat is being used from /usr/bin/cat .**