**Day 03**

* **What is inter process communication(IPC)?**

Processes can coordinate and interact with one another using a method called **inter-process communication (IPC)**.

* **Head command**

The **head** command is used to listthe top few lines of file in linux.

A screenshot of a computer

Description automatically generatedHere a file list.txt is created and it contains the data of 10 lines as shown in the A screenshot of a computer program

Description automatically generatedimage.

Here in the given example :

-n 10:specifies the number of lines to display from top.

(so top 10 lines from the file got displayed)

-n 5:specifies the number of lines to display from top.

(hence top 5 lines from the file got displayed)

**Different modes of vi editor :**

**1.)**input

**2.)**command

**3.)**ESCmode

* **A -**appends test at the end of the line
* i - Inserts text at beginning of line
* Append text to right of cursor
* O-opens line above
* o-opens line below
* rch-replaces single character at cursor
* R-replace text from cursor to right
* S-replaces entire line
* s-replaces single character at cursor with any number of characters
* in vi editor deleting can be done in escape mode and by clicking dd
* u -undo
* shift+$ -moves to the end of file
* in the text hello world if you want to delete the word world keep the cursor at w and press dw.In escape mode keep undo then the world comes back
* yy -yanking means to copy
* p- for paste
* y and 3 : to copy multiple lines here 3 indicates 3 lines
* H – cursor moves upper left corner
* J – down a line
* K- up a line
* L-lower left corner
* **history** command lists the command that have been used in the terminal session

**Changing file permissions:**

User group and others can access the file and perform various operations like read write and execute .

Ch mod is the command that is used to change the permissions of file or directory

Owner (or)super user can change the permission of file

u-user

g-group

o-user

a-all

r- read

w-write

x-execute

+means add permissions

-means remove permissions

* In the below image the user got execution permission for the file1.txt as

**chmod u+x [file name]**

* In the next image the execute permissions for the user is removed by using

**chmod u-x[file name]**

A screen shot of a computer

Description automatically generated

A screen shot of a computer

Description automatically generated

This can also be done by using numbers like

Execute-1

Write-2

Read-4

Ex: chmod 444 file1.txt

This means all the user group and others have the permission to read only they don’t have access to write and execute.

Similarly,

**chown** is used to change the ownership

ex: chown sam file1.txt

**Chgrp** change group of the file

Ex: chgrp groupname filename

Arguments are assigned to special variable

$0- gives the name of executed command

$\*-gives the complete set of positional parameter

$#-gives the number of arguments  
$$-gives the PID of current shell

$!-gives the PID of last background job

$?-gives the exit status of last command

$@-similar to$\*,but generally used with strings in looping constructs