Day 1

03-09-2021

Unix Training

Unix is an Operating system (OS) developed in Bell laboratories.

Unix is multi-tasking, multi-user, high secure and very high performance os

In Unix OS we can customize according to own requirement.

Unix is Open Source.

Unix is command base OS.

Unix flavors

Sun : Solaris

IBM : AIX

Apple : MAC

HP : HP-UX

Google : Android

Unix

And

Linux :it is GUI base

Shell : It is a command or instruction given to kernel through System call to execute some task.

Cd

Shell

1. Read line from standard input device (each of these line is called pipe line).
2. Splits the command into tokens.
3. Checks the token to see If it is a keyword.
4. Check the first word alias.
5. Then execute the command.

Unix Commands

1. echo
2. print
3. date --date ‘Date Format’: check other formats.
4. date +%d-%M-%Y : check other formats.

**File system :**

**pwd** : current directory path.

**ls** : it display all files and folder present in current directory.

mkdir folderName : This command is use to create the folder or directory

mkdir folderName folderName folderName : this command is use to create more than one folder.

rmdir folderName : This command is use to delete the folder.

cd folderName : this command is use to move inside a folder.

mv oldFolderName newFolderName : This command is use to change the folder.

folderName start with pre-fix . consider as hidden folder.

mkdir .folderName

ls : this command is use to display. Non hidden folder.

ls –a : This command I use to show normal as well as hidden folder.

Mv oldFolderName .folderName : This command is use to hide the folder.

cd .. : this command is use to move to parent directory of current directory.

cd ~ : this command is use to move to root directory.

Unix file system

Creating the file in Unix

1. cat command :
   1. creating file using cat command

cat > filename.txt : it create the new file with allow to enter the contents inside a file.

* 1. Reading the content from file using cat command

Cat < filename.txt or cat filename

* 1. Cat >> filename : this command is use to append the content in existing file.

1. Touch filename : This command is use to create the empty file

Touch filename filename filename

1. Using echo

echo “Welcome to Unix Training “ > filename

head and tail command :

head –n filename : this command is use to display the top n number line from a file

tail –n filename : this command is use to display the bottom n number of line from a file

wc : word count :

wc –l filename : number of lines

wc –w filename : number of words

wc –c filename : number of character

cp : copy the content from one file to another file.

cp sourcefilename destinationfilename.

Delete the file

rm filename

rm –I filename : it ask the confirmation to delete the file.

ls folderName : This command is use to display directory details.

rmdir folderName This command is use to delete the folder if folder is empty.

rm –r folderName This command is use to delete the folder doesn’t matter folder empty or contains set of files.

Filter command : flat file : in a file data entered by using delimiter ie space , \_, tab etc.

Employee.txt file with few records

cut -c 1 Employee.txt : 1 index position

cut -c 3 Employee.txt : 3 index position

cut -c 3-6 Employee.txt : 3 to 6 range index position

cut -d ' ' -f2 Employee.txt : -d delimiter ‘ ’ space f2 2nd columns

cut -d ',' -f1 Manager.txt : -d delimiter ‘,’

paste Employee.txt : display the content

paste –s Employee.txt : serial format

paste Employee.txt Manager.txt both file contents.

Tr ( translate command)

This command is use to translate data from one format to another format.

tr SET1 SET2 < Employee.txt

search SET1 content and replace by SET2 contents in file.

tr abcde ABCDE < Employee.txt

a-z A-Z < Employee.txt

tr ' ' , < Employee.txt

tr [:lower:] [:upper:] < Employee.txt : please work more pre-defined keywords to translate the content from file.

sort Employee.txt : ascending order as 1st field

sort –r Employee.txt : descending order

sort -k2 Manager.txt k2 send field

sort -k2 -t ',' Manager.txt -t delimiter ‘,’ by default space consider.

EmployeeInfo.txt

Id,Name,Salary,Desg, City, DeptId

4,Raj,19000,Developer,Bangalore,10

2,Raju,18000,Developer,Delhi,20

5,Ramesh,15000,Developer,Bangalore,20

8,Ajay,12000,Developer,Delhi,10

9,Vikash,19000,Developer,Bangalore,20

6,Ram,12000,Developer,Delhi,20

7,Vijay,18000,Developer,Bangalore,10

1. Sort by Id
2. Sort by city
3. Sort by City and sub sort by DeptId

Note : column name not sort(ignore).

GREP Commands GREP Global Regular Express Print

Grep command actually search then the content in file base upon the pattern.

The grep command mainly divided into three parts.

1st grep command followed by pattern to search in file and then filename or contents.

grep 'e' grepdemo.txt : display the line where e character present

grep –i 'e' grepdemo.txt : ignore case sensitive

grep -c -i 'E' grepdemo.txt : it display the number of line where E character present.

grep -n -i 'e' grepdemo.txt : it display the line which contains e character as well as it display the line numbers.

Please work more GREP Commands

SED : The SED command Unix stands for stream editing commands.

Using SED command we can modify the contents of the file.

SED command in Unix basically use search and replace the text or contents from a file.

Pipe commands

Pip e command is use to combine more than command

So first command execute it provide the output and that is input for another command.

echo “Welcome to Unix Training “ |

SED command

1. Read a line from input stream
2. Execute the command on a line
3. Display the result on output stream.

In SED command we use s and g

S : substitution

G : means many occurrence.

sed s/Raj/'Raj Deep'/g seddemo.txt

sed s/EE/i/ seddemo.txt

sed s/ee/i/g seddemo.txt

sed s/ee/i/ seddemo.txt

sed '2 s/R/r/g' seddemo.txt

sed -i s/Raj/'Raj Deep'/gi seddemo.txt : this command search, replace and update in same file.

sed s/'Hor r you'/'How are You'/g a.txt

SED commands

Day 2

06-09-2021

**Awk : awk is one of the most powerful tool in Unix use for processing the rows and column in the file base upon the delimiter.**

**Syntax**

**Awk ‘BEGIN {start\_action} {action} END {stop\_action}’ filename**

**Start action : initialization**

**Action : looping**

**End action : at the last**

**awk -F ',' '{print $0}' Employee.txt : display all rows**

**awk -F ',' '{print $1}' Employee.txt : display specific fields from a file**

**awk -F ',' '{print $2}' Employee.txt : display specific fields from a file**

**NF : Number of Fields**

**NR : Number of records**

**GREP : grep is useful if you want to quickly search data from line or file that match our pattern. It also return some other simple information like matching number, match count, and file names**

**SED : it is use when you want to make any changes in a file base upon the pattern match. If allow you to easily match part of line, makes modification and print out result or update in file.**

**AWK : it is entire programming language. Build around reading any type of file (CSV (comma separated values) style file, processing records and optionally print out the result.**

**File access permission**

**File**

**Types of users**

1. **File owner**
2. **Group owner**
3. **Others**

**U -> it represent the owner**

**G -> it represent group of owner**

**O-> it represents any other user but not belong to user group.**

**Permission for regular file**

1. **Open file : Read mode**
2. **Write file or modify : write mode**
3. **Execute : Read and Execute**

**ls –l filename : this command is use to display the properties of a file.**

1. **First field display - : means it is a file**
2. **The owner has over the file. rw**
3. **The group has over the file. r**
4. **Other people has over the file r**
5. **Fifth : number of links for that folder or file. 1**
6. **User that own the file or directory**
7. **Group details.**
8. **Size of the file in byte**
9. **Time when last modification happen**
10. **File name**

**chmod : change mode**

**chmod <permission> <filename>**

**code**

**a : all**

**u : user**

**g : group**

**o : other**

**+ : add**

* **: remove**

**R : read , w : write , x : execute**

**0 🡪 no permission**

**1 🡪 execute permission**

**2 -> write permission**

**4 🡪 read permission**

**4 + 2 +1 : 7 (read, write and execute)**

**4+0+1 : 5 (read, no write and execute)**

**4+0+0 : 4 (read, no write and no execute)**

**chmod 754 fileName**

**User 🡪 read, write and execute**

**Group 🡪 read, no write and execute**

**Other 🡪 read, not write and execute**

**Chmod 777 filename**

Shell scripting

A shell is a program that takes command typed by user and send the instruction to kernel through system call and Kernel run the command.

A shell is a program that acts as the interface between you and the Unix or Linux operating system.

Types of Shell

1. The Bourne Shell
2. Sh : Posix
3. Ssh : Korne shell
4. Bash :
5. C shell
   1. csh
   2. Tops tcsh

sudo apt install shellName

shell programming

we can do shell programming using two ways

1. You can type a sequence of command and allow the shell to execute them interactively.
2. You can store those commands in a file and you can run the file.

VI Editor

Open the file with vi editor

vi filename.txt

1. Command mode
2. Insert mode
3. Exit mode

Open the open the file to move from command mode to insert mode we have write i

Then write the contents

To come out from insert mode to command mode we have to click esc key.

To save the write :w

To quite the file :q

To save and quite : wq!

**Shell scripting**

You create the file and write one or more command and save the file with extension .sh

Variable : variable is a name which hold some value and the value can change during the execution of a program.

2 types

1. pre-defined variable : This provide the OS details. All pre-defend variable are in upper case. When we are using those variable we have to start with $ symbol.
2. User-defined variable

Taking the value through keyboards

1. Command line arguments.

sh ./filename firstargument secondarguemnt thirdargument

$1 : first parameter

$2 : second parameter

$3 : third parameter

$0 : script file name

$\* : number of arguments

$# : all arguments .

1. Using read property

-gt

-gte

-lt

-lte

-eq

-ne