da

**AIM:-**

To write a shell script to stimulate the file commands of rm,cp,cat,mv,cmp,wc, split,diff.

**Algorithm:-**

**STEP 1:** Start the process

**STEP 2**: In the normal command prompt of Linux perform the operations for following commands

**STEP 3:** use cat command to create,append and display the file on the screen **cat >filename**  -to create **cat >>filename** -to append **cat filename** to display.

**STEP 4:**use cp command to copy the old file into new file **cp filename (old) filename 2 (new).**

**STEP 5:** use **cmp** command to compare **rm** command to remove mv command to move and wc command to count the word from the file.

**Step 6 :** finally use **split** command to split the one phase into

another and **diff** command is to show the difference between

one file with another.

**Step 7** : stop the process.

**EX.NO: 1 P.G.NO**

**DATE:-** Basic Linux commands

1

OUTPUT

**\* LINUX – COMMANDS \***

\* **Clear:** **[**it clear the screen]

[tsa@telnet ~]$ clear



\***cat:**

(i) **Create a file:**

[syntax : $cat > filename]

[tsa@telnet ~]$ cat>sample1

welcome to tsa college

\***Display the file:**

[tsa@telnet ~]$ cat sample1

welcome to tsa college

(ii) \* **Append :**

[syntax : $cat > >filename]

[tsa@telnet ~]$ cat >> sample1

Always best learning institute

**After Appending:**

[tsa@telnet ~]$ cat sample1

welcome to tsa

Always best learning institute

\* **Copying the two files store into another file:**

[tsa@telnet ~]$ cat sample sample1 > biglist

[tsa@telnet ~]$ cat biglist

welcome to every body

welcome to tsa

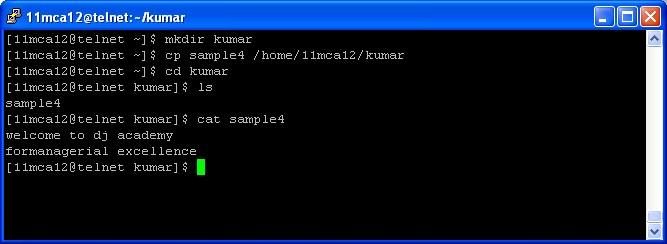
**\*cp:** [copy files and directories]

[tsa@telnet ~]$ mkdir kumar

[tsa@telnet ~]$ cp sample4 /home/tsa/kumar

[tsa@telnet ~]$ cd kumar

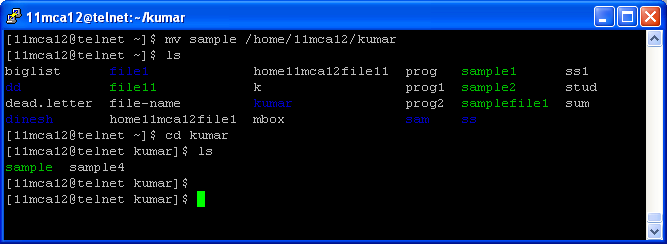
[tsa@telnet kumar]$ ls



\* **mv:** [move the files]

[tsa@telnet ~]$ mv sample /home/tsa/kumar

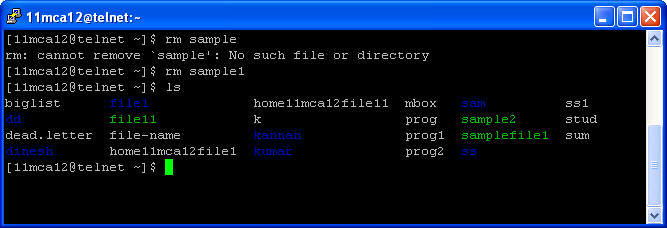
[tsa@telnet ~]$ ls



\***rm:** [remove files or directories]

[tsa@telnet ~]$ rm sample1

[tsa@telnet ~]$ ls



**Wc:**

* Print the number of newlines, words, and bytes in files

**[tsa@telnet ~]$ wc unix1**

11 23 210 unix1

**[tsa@telnet ~]$ wc unix1 jagadesh**

11 23 210 unix1

2 5 23 santhi

1. 28 233 total

**Diff:**

* Find differences between two files

**[11tsa28@telnet ~]$ diffjagaunix**

1,2c1,11

<JAGADESH

< linux

--

> 1.high level language

> 2.multi taskingcapaability

> 3.multi user capability

> 4.portabality

> 5.communications

> 6.standard input,output and error

**Result:-** The above program is successfully executed and verified.

**Output** Has Been Verified.

**EX.NO:2 P.G.NO**

USER AND SYSTEM INFORMATION COMMANDS

**DATE:- 6**

**AIM:**

To write a shell script to implement the user and System information by commands.

**ALGORITHM:**

**Aim :**

To write a shell script to implement the user and System information by commands.

**Algorithm :**

**Step 1** : start the process.

**Step 2 :** In the shell prompt window perform the following commands use **logname** to check log name of user, shell to check the following shell information.

**Step 3 :** type **ostype** command to check the ostype of Linux os.

**Step 4 :** type path to check the path for particular directories.

**Step 5 :** type **pwd** command to view the present working directory.

**Step 6 :** **is CPU** command to check the CPU information.

**Step 7 :** stop the process

#!/bin/bash  
echo -e "username:”$USER  
echo -e "loginname:”$LOGNAME  
echo -e "currentshell:”$SHELL  
echo -e "homedirectory:$HOME  
echo -e "our pc os is:$OSTYPE  
echo -e "current directory:"pwd  
echo -e "system configuration (or) pc configuration:"lscpu  
echo -e "free memory space:"free -m

**Output**

Tsa

Tsa

Cshshell

Command ! name from package ‘home(multiverse)

Command ! name from package ‘home(universe)

**Result:-** The above program is successfully executed and verified.

**EX.NO:3 PG.NO**

**DATE:- COMBINE COMMANDS 8**

**10**

**AIM:**

To write a shell script to implement the following of pipes, redirection and the commands.

**ALGORITHM:**

**STEP 1 :** start the process.

**STEP 2 :** use **( | )** pipe command to link one command with another (or) one operation.

**STEP 3 :** use **(>>)** command to transfer the file from one part into another.

**STEP 4 :** use **More** command to check the information on the screen.

**STEP 5 :** display the result on the screen.

**STEP 6 :**Save the process.

**STEP 7:**End the process

1. **Pipes:**

[tsa03@localhost ~]**$ ls -la |wc**

**Output:**

64 569 3333

(ii) \* **Append :**

**[syntax : $cat > >filename]**

[tsa@telnet ~]$ cat >> sample1

Always best learning institute

**After Appending:**

[tsa@telnet ~]$ cat sample1

welcome to tsa

Always best learning institute

**More:**

[tsa03@localhost ~]$ **ls -la | more**

total 504

drwx------ 4 tsa03 tsa03 4096 Feb 13 08:40 .

drwxr-xr-x 53 root root 4096 Feb 18 2002 ..

-rw-rw-r-- 1 tsa03 tsa03 24 Jan 23 01:33 aa

-rw-rw-r-- 1 tsa03 tsa03 8 Feb 5 23:35 ais

-rw-rw-r-- 1 tsa03 tsa03 16 Feb 5 23:35 aismin

-rw-rw-r-- 1 tsa03 tsa03 13 Feb 6 01:28 aisminaa

**Result:-** The above program is successfully executed and verified.

**Ex.No.4 Pg.No**

DIRECTORY COMMANDS

**Date:- 10**

**AIM :**

To write a shell script for displaying **current date user name, file listing and directories** using **switch case** statement.

**ALGORITHM :**

**STEP 1 :** start the process.

**STEP 2 :** use case statement for performed an different action into an single prompt.

**STEP 3 :** declare **case** variables and case command for the program.

**STEP 4 :** : if the choice is **1** then the current date will be displayed on the screen.

**STEP 5 :** : if the choice is **2** then username will be shown .if the choice is three file can be listed and finally if the choice is four directories are displayed on the screen.

**Step 6** : it none of the choice is met finally default case get executed on the screen.

**Step 7 :** stop the process

**CODING:-**

echo "1- who am i?"

echo "2- who is logged on?"

echo "3- date"

echo "4- calendar"

echo "5- Current directory"

echo "6- file listing"

echo "enter your choice:"

read n

case $n in

1) whoami;;

2) who;;

3) date;;

4) cal;;

5) pwd;;

6) ls;;

esac

**output :-**

echo "1- who am i?"

echo "2- who is logged on?"

echo "3- date"

echo "4- calendar"

echo "5- Current directory"

echo "6- file listing"

echo "enter your choice:"

2

**tsa**

**Result:-** The above program is successfully executed and verified.

**Result:-** The above program is successfully executed and verified.

Pg no

FILTER COMMANDS

12

**EX.NO:-5**

**DATE:-**

**AIM :**

To write a shell script to implement the filter commands.

**ALGORITHM :**

**STEP 1 :** start the process.

**STEP 2 :** use **Grep** command likes help to search a particular word (or) particular text on a file.

**STEP 3** : : use **sort** command to arrange an particular text (or) word on screen.

**STEP 4 :** **wc** command to check the word information on the screen.

**STEP 5 :** Head and tail command to get the information on forward and backward portion respectively.

**STEP 6 :** use **pipe** command to combine process with another and more command to get detailed information on the screen.

**STEP 7 :** **Stop** the process

**Grep**

login as: tsa03

tsa03@172.16.1.12's password:

Last login: Wed Feb 13 07:36:52 2002 from 172.16.11.210

[tsa03@localhost ~]$ cat > grp

grep is used to display the patten of file

grep -patten

grep-file name

patten-type of a file

file name is user identifier

**Grep :**

[tsa03@localhost ~]$ grep "^g" grp

**Output:**

grep is used to display the patten of file

grep -patten

grep-file name

**Grep –e:**

[tsa03@localhost ~]$ grep-e "$e" grp

**Output:**

grep is used to display the patten of file

grep -patten

grep-file name

patten-type of a file

file name is user identifier

**Grep-i:**

[tsa03@localhost ~]$ grep -i "^G" grp

**Output:**

grep -i "^" grp

grep is used to display the patten of file

grep -patten

grep-file name

patten-type of a file

file name is user identifier

**Grep-H:**

[tsa03@localhost ~]$ grep -H "^g" grp

**Output:**

grep -H "^g" grp

grp:grep is used to display the patten of file

grp:grep -patten

grp:grep-file name

**Grep-n:**

[tsa03@localhost ~]$ grep -n "^g" grp

**Output:**

1:grep is used to display the patten of file

2:grep -patten

3:grep-file name

**Redirection:**

**Input and output redirection:**

[tsa03@localhost ~]$ cat file2

welcome to unix

how is it

[tsa03@localhost ~]$ cat < file2 > file3

[tsa03@localhost ~]$ cat file3

**Output:**

welcome to unix

how is it

**Error redirection:**

[tsa03@localhost ~]$ cat fl 2 > fle.log

cat: fl: No such file or directory

cat: 2: No such file or directory

[tsa03@localhost ~]$ cat fl

**Output:**

cat: fl: No such file or directory

**Pipes:**

[tsa03@localhost ~]$ ls -la |wc

**Output:**

64 569 3333

[tsa03@localhost ~]$ ls -l | grep "dkl" |wc

**Output:**

0 0 0login as: tsa03

[tsa03@localhost ~]$ ls -l | grep "i" |wc

**Output:**

23 207 1176

**Filter:**

**Word count(wc)**

**Wc –l:**

[tsa03@localhost ~]$ wc -l ais

**Output:**

4 ais

**Wc-w:**

[tsa03@localhost ~]$ wc -w ais

**Output:**

4 ais

**Wc-c:**

[tsa03@localhost ~]$ wc -c ais

**Output:**

8 ais

**Wc-L:**

[tsa03@localhost ~]$ wc -L ais

**Output**:

1 ais

**Wc-m:**

[tsa03@localhost ~]$ wc -m ais

**Output:**

8 ais

**Sorting:**

[tsa03@localhost ~]$ cat > stud

**Output:**

name rno m1 m2 m3 tot

eso 3 90 90 90 270

ais 1 89 89 89 267

ganga 4 99 99 99 297

pri 6 90 90 90 270

j 5 99 99 99 297

barathi 2 99 99 99 297

**Sort-g:**

[tsa03@localhost ~]$ sort -g stud

**Output:**

name rno m1 m2 m3 tot

ais 1 89 89 89 267

barathi 2 99 99 99 297

ganga 4 99 99 99 297

j 5 99 99 99 297

pri 6 90 90 90 270

**Sort-d:**

[tsa03@localhost ~]$ sort -d stud

**Output:**

name rno m1 m2 m3 tot

ais 1 89 89 89 267

barathi 2 99 99 99 297

ganga 4 99 99 99 297

j 5 99 99 99 297

pri 6 90 90 90 270

**Sort-r:**

[tsa03@localhost ~]$ sort -r stud

**Output:**

name rno m1 m2 m3 tot

pri 6 90 90 90 270

j 5 99 99 99 297

ganga 4 99 99 99 297

barathi 2 99 99 99 297

ais 1 89 89 89 267

**More:**

[tsa03@localhost ~]$ ls -la | more

total 504

drwx------ 4 tsa03 tsa03 4096 Feb 13 08:40 .

drwxr-xr-x 53 root root 4096 Feb 18 2002 ..

-rw-rw-r-- 1 tsa03 tsa03 24 Jan 23 01:33 aa

-rw-rw-r-- 1 tsa03 tsa03 16 Feb 5 23:35 aismin

-rw-rw-r-- 1 tsa03 tsa03 13 Feb 6 01:28 aisminaa

-rw-rw-r-- 1 tsa03 tsa03 11 Feb 6 01:28 aisminab

-rw-rw-r-- 1 tsa03 tsa03 9 Feb 6 01:28 aisminac

-rw------- 1 tsa03 tsa03 8581 Feb 14 08:24 .bash\_history

--More—

**RESULT :-**

The Program Is Successfully Executed And Verified

**EX.NO:6 P.G.NO**

**ZERO BYTE CHARACTERS**

**DATE:- 21**

**AIM:**

To write a shell script to remove the file which has file size as zero.

**ALGORITHM:**

**STEP 1 :** start the process.

**STEP 2 :** use clear command to clear the screen from Window

**STEP 3** : use echo command to display from the screen.

**STEP 4 :** use read command to get the filename on the screen.

**STEP 5 :** use if statement to check the filename.

**STEP 6 :** : if the filename match then remove the file using email command.

**Step 7 :** otherwise display the details as file size is zero.

**Step 8 :** use f1 command to close the if statement.

**Step 9 :** display the result on the screen and finally stop the process

**Coding :-**

Clear

echo ”enter the filename”

read fname

if –s $fname

then

echo “file name size is not zero bytes”

echo “the details of the file are ”

ls-l $fname

else

echo “file name size is zero bytes”

rm $ fname

fi

output :

enter the filename jk

file name size is zero bytes

**RESULT :-** The Program Is Successfully Executed And Verified

**EX.NO:7 P.G.NO**

**SUM OF N NUMBERS**

DATE:-  **23**

**AIM :**

To write a shell script program to final the sum of N number.

**ALGORITHM:**

**STEP 1 :** start the process..

**STEP 2 :** read the number to get random number.

**STEP 3 :** read the Random number to find **sum**.

**STEP 4 :** using do while loop with the condition

**STEP 5 :** added result will be printed using “**echo**”.

**STEP 6 :** Save the process.

**STEP 7 :** End the process

**CODING**

#!/bin/bash

echo "enter the number"

read n

sum=0

a=$n

while(($n >0))

do

x=`expr $n % 10`

sum=`expr $sum + $x`

n=`expr $n / 10`

done

echo "the sum of $a is :”$sum

**OUTPUT :-**

Enter the number 10

The sum of 10 is 55

**RESULT :-** The Program Is Successfully Executed And Verified

**PG NO**

**25**

EX.NO:-8

DATE:-

GREATEST OF GIVEN NUMBERS

**AIM :**

To write a shell script to find the greatest of set of numbers using command line Argument.

**ALGORITHM :**

**STEP 1 :** start the process.

**STEP 2 :** read three numbers 1 is greater than number 2 and number 3.

**STEP 3 :** if it is true, then print number 1 is greatest.

**STEP 4 :** else if check number 2 ks greater than number 1 and number 3.

**STEP 5 :** if it is true, then print number 2 is greatest.

**STEP 6 :** else print number 3 is greatest.

**STEP 7 :** End the process.

.

**Coding**

#!/bin/bash

echo "enter first number"

read first

echo "enter second number"

read sec

echo "enter third number"

read third

if [ $first -gt $sec ] ; then

if [ $first -gt $third ] ; then

echo -e " $first is greatest number "

else

echo -e " $third is greatest number "

fi

else

if [ $sec -gt $third ] ; then

echo -e " $sec is greatest number "

else

echo -e " $third is greatest number "

fi

fi

**OUTPUT**

ENTER THE FIRST NUMBER :50

ENTER THE SECOND NUMBER :60

ENTER THE THIRD NUMBER :40

SEC IS GREATEST NUMBER

**RESULT :-**

The Program Is Successfully Executed And Verified

PG.NO

27

**EXNO:-9**

**DATE:-**

**PALINDROME**

**AIM :**

To write a shell script program for **palindrome** using command Arguments.

**ALGORITHM :**

**STEP 1 :** Start the process.

**STEP 2 :** Input the string using command line Argument assign the value 10 &amp; 1.

**STEP 3 :** use the “reverse” to reverse the string and store in the variable reverse.

**STEP 4 :** use it condition and check $str==$reverse if both are equal it is a palindrome.

**STEP 5 :** otherwise display the string is not a palindrome.

**STEP 6 :** Save the process.

**STEP 7 :** End the process.

**Coding**

echo "Enter the string"

read s

echo $s > temp

rvs="$(rev temp)"

if [ $s = $rvs ]

then

echo "it is palindrome"

else

echo " it is not a palindrome"

fi

**OUTPUT**

Enter the string

**Mam**

It is a Palindrome

**RESULT :-**

The Program Is Successfully Executed And Verified

**EX.NO: 10 P.G.NO**

**MULTIPLICATION OF TABLES**

**DATE:- 29**

**Aim :-**

To write a shell script to print multiplication table of the given Argument.

**Algorithm :-**

**Step1;** start the process.

**Step2;** declare the variables.

**Step3;** initialize i=1.

**Step4;** we use while loop with an less than condition on the screen

**Step5;** use echo statement with express command to get the output on the screen.

**Step6;** while start and ends with do and done keywords.

**Step7;** Save the process

**Step8;** Stop the process.

**CODING :-**

**echo "Enter a Number"**

read n

echo "Enter Range"

read r

i=0

while [ $i -le $r ]

do

echo " $n x $i = `expr $n \\* $i`"

i=`expr $i + 1`

done

**output:-**

1\*5=5

2\*5=10

3\*5=15

4\*5=20

5\*5=25

6\*5=30

7\*5=35

8\*5=40

9\*5=45

10\*5=50

**RESULT :-** The Program Is Successfully Executed And Verified