**Shell Scripting:**

$0 – represents name of the script

$1 – 1st Argument

$2 – 2nd argument

$3 – 3rd argument

$4 – 4th argument

$\* -- All arguments passed to the string

$@ -- All arguments passed to the string as an array format

$$ -- process id of the current running process

$! – process id of the last command went into the background

$# -- it will count the total number of arguments passed to the script

$? – this is used to print status of the last executed command

* it displays 0 if the command is success
* it displays non zero, if the command was unsuccessful

**Day 15: Bash/Shell Scripting**

**Note:**

* No space for variable
* while using the expression we need to use the space
* we need to use the space for conditional statements
* we need to use the space for loops

**Example:**

name=”Phani”

id=1

sum=`expr $num1 + $num2`

if [ $num -eq 3 ]

while [ $num -gt 0 ]

**This won’t work**

name = ”Phani”

id = 1

sum=`expr $num1+$num2`

sum = `expr $num1+$num2`

if [$num -eq 3]

while [$num -gt 0]

**Write a shell script to display the statement “Welcome to the devops class”?**

**touch is used to create an empty script file**

touch display.sh

The file should have an extension .sh

vi display.sh

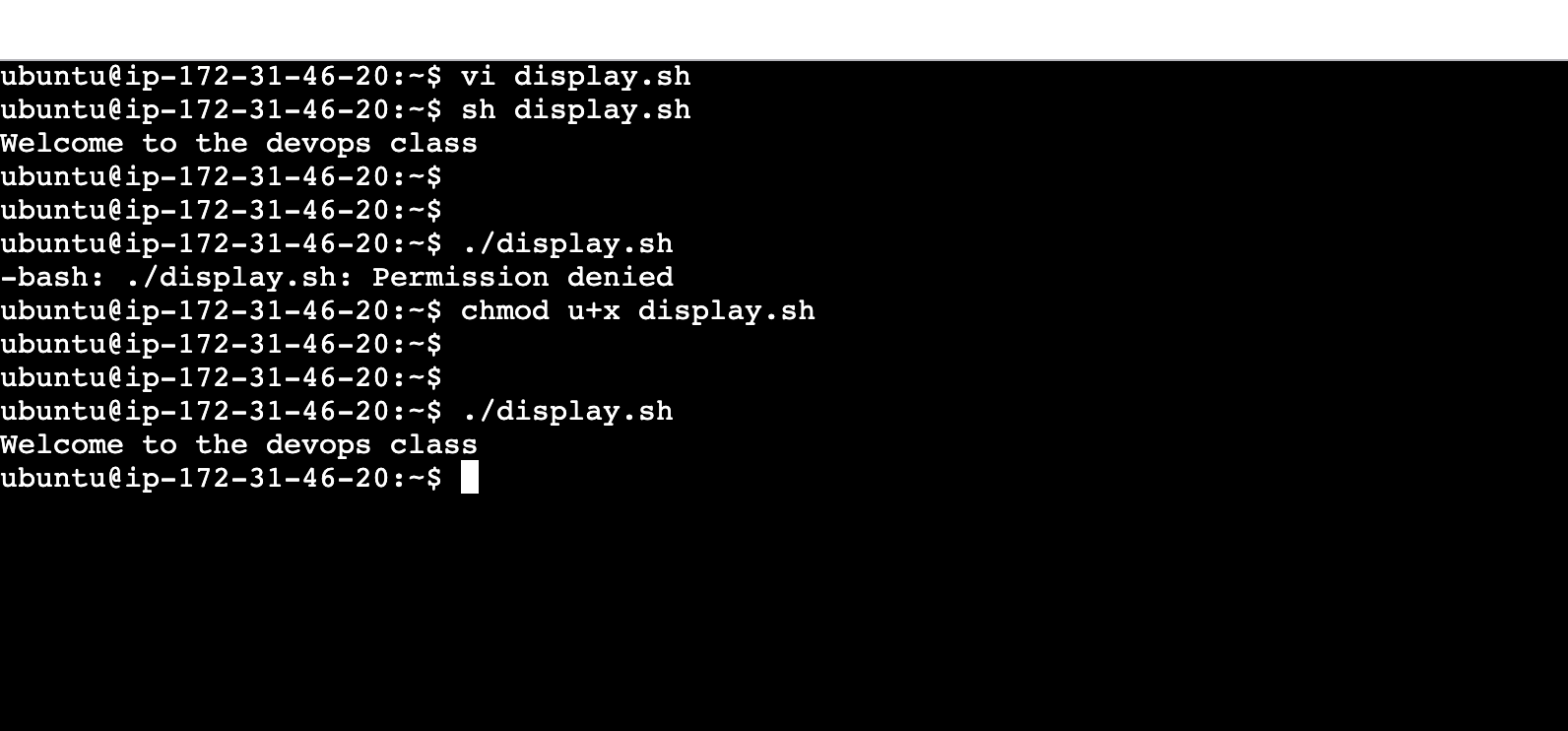
To execute the file, we need to make user we have the execute permission(x)

chmod u+x display.sh

to execute the script file

sh display.sh (sh is the compiler here)

./display.sh



#!/bin/bash – this is known as shebang

This will let the machine know that this is a bash script

A black background with pink text

Description automatically generated

|  |
| --- |
| #!/bin/bash  echo "Welcome to the devops class " |

A black background with white text

Description automatically generated

|  |
| --- |
| #!/bin/bash  echo -e "Welcome to the devops class\nthis is the first scripting class" |

**o/p:**

Welcome to the devops class

this is the first scripting class

**To display the hostname:**

|  |
| --- |
| #!/bin/bash  echo "The servername is $HOSTNAME" |

**o/p:**

The servername is ip-172-31-46-20

**userdefined variables:**

|  |
| --- |
| #!/bin/bash  name="Phani"  id=1  echo "The name is $name and the ID value is $id" |

**o/p”**

The name is Phani and the ID value is 1

**read:**

read function is used to read the values in the real time

syntax:

read variablename

|  |
| --- |
| #!/bin/bash  name="Phani"  echo "The name is $name"  echo "Please enter the last name and the mobilenumber"  read lname  read mobile\_number  echo "The name is $name , the last name is $lname and the mobile number is $mobile\_number" |

**o/p:**

The name is Phani

Please enter the last name and the mobilenumber

nandigam

9964437999

The name is Phani , the last name is nandigam and the mobile number is 9964437999

**Day 16: Bash/Shell Scripting**

**command line arguments** – while executing the script we can pass values as inputs

swp file (hidden file) (corrupt file)

|  |
| --- |
| #!/bin/bash  echo “Printing the command line argumetns $1, $2,$@, $#, $?” |

**./5commandlinearguments.sh 10 wleomce**

**o/p**

Printing the command line argumetns 10, wleomce,10 wleomce, 2, 0

$1- 1st argument

$2- 2nd argument

$@ -- arguments passed in the string format

$# -- number of arguments passed to the script

$? – status of the last executed command

**./5commandlinearguments.sh 10 wleomce 23 30**

**o/p**

Printing the command line argumetns 10, wleomce,10 wleomce 23 30, 4, 0

**Write a script to find the addition and the multiplication of 2 numbers.**

expression (expr)

we need to use the back quote here(``) not single quotes(‘’)

back quote is used to assign the output of an expression to a variable

|  |
| --- |
| #!/bin/bash  echo "Please pass interger values number1:"  read num1  echo "Please pass interger values number2:"  read num2  sum=`expr $num1 + $num2`  multiply=`expr $num1 \\* $num2`  echo "The sum of $num1 and $num2 is $sum"  echo "The sum multiplication $num1 and $num2 is $multiply" |

**o/p:**

Please pass interger values number1:

10

Please pass interger values number2:

20

The sum of 10 and 20 is 30

The sum multiplication 10 and 20 is 200

**Assignment: Subtraction and division**

|  |
| --- |
| #!/bin/bash  echo "Please pass value for number1:"  read num1  echo "Please pass value for number2:"  read num2  subtraction=`expr $num1 - $num2`  division=`expr $num1 / $num2`  echo "The substarction of $num1 and $num2 is $subtraction"  echo "The division of $num1 and $num2 is $division" |

**o/p**

Please pass value for number1:

30

Please pass value for number2:

3

The substarction of 30 and 3 is 27

The division of 30 and 3 is 10

**conditional statements**

These are the statements used to compare the values

* **if statement:**  
  syntax: if [ condition ]  
   then   
    
   statement  
   fi
  + if the condition is true, then the statement inside if will be executed
* **if else**  
  syntax: if [ condition ]  
   then   
   statement1  
    
   else

statement2

fi

* + if the condition is true, then the statement1 will be executed and if the condition is false statement2 will be executed
* **Nested if : one if statement inside another if statement**  
  syntax: if [ condition1 ]  
   then   
   statement1  
   elif [ condition2 ]  
   then   
   statement2  
   else

statement3

fi

* + if the condition1 is true, then the statement1 will be executed
  + if the condition1 is false and condition2 is true then the statement2 will be executed
  + if the condition1 and condition2 are false then the statement3 will be executed

**Day 17: Bash/Shell Scripting**

Write a script to check whether the given number is 3 or not?

|  |  |  |
| --- | --- | --- |
| **String** | **Number** | **Description** |
| == | -eq | Equal |
| != | -ne | Not equal |
| > | -gt | Greater than |
| >= | -ge | Greater than or equal to |
| < | -lt | Less than |
| <= | -le | Less than or equal to |

|  |
| --- |
| #!/bin/bash  echo "Please enter the integer number to compare"  read num  if [ $num -eq 3 ]  then  echo "$num is equal to 3"  else  echo "$num is not equal to 3"  fi |

**o/p:**

Please enter the integer number to compare

3

3 is equal to 3

------

Please enter the integer number to compare

5

5 is not equal to 3

Write a script to find the biggest of two numbers?

|  |
| --- |
| #!/bin/bash  echo "Please enter integer value1:"  read num1  echo "Please enter integer value2:"  read num2  if [ $num1 -gt $num2 ]  then  echo "$num1 is greater than $num2"  elif [ $num1 -lt $num2 ]  then  echo "$num1 is less than $num2"  else  echo "$num1 is equal to $num2"  fi |

**o/p:**

Please enter integer value1:

10

Please enter integer value2:

20

10 is less than 20

-----------------

Please enter integer value1:

10

Please enter integer value2:

10

10 is equal to 10

-----------------

Please enter integer value1:

20

Please enter integer value2:

10

20 is greater than 10

Write a script whether the given path is a file/ directory or a link?

**-L is link**

**-f is file**

**-d is directory**

These are predefined

|  |
| --- |
| #!/bin/bash  echo "Enter the path to find if it's a file or directory or link"  read name  if [ -L $name ]  then  echo "The given path is a link"  elif [ -d $name ]  then  echo "The given path is a directory"  elif [ -f $name ]  then  echo "The given path is a file"  else  echo "The path doesn't exists"  fi |

o/p:

Enter the path to find if it's a file or directory or link

10directoryORfileORlink.sh

The given path is a file

Enter the path to find if it's a file or directory or link

scripting

The path doesn't exists

Enter the path to find if it's a file or directory or link

syntax.sh

The given path is a file

mkdir -p Testdir

./10directoryORfileORlink.sh

Enter the path to find if it's a file or directory or link

Testdir

The given path is a directory

ln -s 10directoryORfileORlink.sh Testlink

./10directoryORfileORlink.sh

Enter the path to find if it's a file or directory or link

Testlink

The given path is a link

Note:

If we use the condition if [ -f $name ]

first and then if [ -L $name ], then the link will be displayed as file, even though it’s a link

Enter the path to find if it's a file or directory or link

Testlink

The given path is a file

Assignment:

Write a script to find biggest of 3 numbers?

|  |
| --- |
| #!/bin/bash  echo "Please enter value for Num1:"  read num1  echo "Please enter value for Num2:"  read num2  echo "Please enter value for Num3:"  read num3  if [ $num1 -gt $num2 ] && [ $num1 -gt $num3 ]  then  echo "$num1 is greater than $num2 and $num3"  elif [ $num2 -gt $num1 ] && [ $num2 -gt $num3 ]  then  echo "$num2 is greater than $num1 and $num3"  else  echo "$num3 is greater than $num1 and $num2"  fi |

**Day 18: Bash/Shell Scripting**

**Loops:**

**While loop:** The statement with in the while loop will be executedonly if the condition is true. Once the condition becomes false, the execution will come out of the whole loop and continue with the execution

**Syntax:**

while [ condition ]

do

statement

done

**Write a script to find the sum of first n numbers?**

|  |
| --- |
| #!/bin/bash  echo "Please enter the integer value for N:"  read num  sum=0  while [ $num -gt 0 ]  do  sum=`expr $sum + $num`  num=`expr $num - 1`  done  echo "The sum of N number is $sum" |

**o/p:**

Please enter the integer value for N:

5

The sum of N number is 15

**Write a script to find the factorial of a number?**

|  |
| --- |
| #!/bin/bash  echo "Please enter the integer value for N:"  read num  factorial=1  while [ $num -gt 0 ]  do  factorial=`expr $factorial \\* $num`  num=`expr $num - 1`  done  echo "The factorial of N number is $factorial" |

**o/p:**

Please enter the integer value for N:

3

The factorial of N number is 6

Please enter the integer value for N:

5

The factorial of N number is 120

**Write a script to find the number of characters in each line of a file?**

|  |
| --- |
| #!/bin/bash  echo "Please enter the name of the file:"  read file\_name  linenumber=1  while read line  do  count=`echo "$line"|wc -c`  echo "The number of characters on line $linenumber is $count"  linenumber=`expr $linenumber + 1`  done < $file\_name |

**o/p:**

Please enter the name of the file:

readline

The number of characters on line 1 is 3

The number of characters on line 2 is 28

The number of characters on line 3 is 50

**Day 19: Bash/Shell Scripting**

write a script to display all the vehicles that were built before 2015?

|  |
| --- |
| #!/bin/bash  echo "Please enter the filename:"  read filename  echo "Pleae enter the built year"  read year  echo "The vechile names which were built before $year"  while read line  do  built\_year=`echo "$line"|awk -F " " '{print $3}'`  if [ $built\_year -le $year ]  then  #echo "The vechile names which were built before $year"  echo "$line"|awk -F " " '{print $1}'  fi  done<$filename |

**o/p:**

Please enter the filename:

vehicle

Pleae enter the built year

2026

The vechile names which were built before 2026

xuv

nexon

be6e

hyrider

LnTtractor

Please enter the filename:

vehicle

Pleae enter the built year

2015

The vechile names which were built before 2015

xuv

LnTtractor

**set -x**

This is used to debug the shell script step by step

**sleep**

this is used to pause the execution for some time

syntax:

sleep NUMBER[SUFFIX]

* **“NUMBER”** represents the time duration for which the command should sleep.
* **“SUFFIX”** can be used to specify the unit of time (s for seconds, m for minutes, h for hours, etc.).
* **Note:**If no suffix is provided, the default unit is seconds.

example:

sleep 10

sleep 3m

sleep 3h

|  |
| --- |
| #!/bin/bash  echo "Please enter the number"  read num  val=1  while [ $val -le $num ]  do  sleep $val  echo "$val"  val=`expr $val + 1`  done |

**How do we check the logs for running application?**

tail -f filename

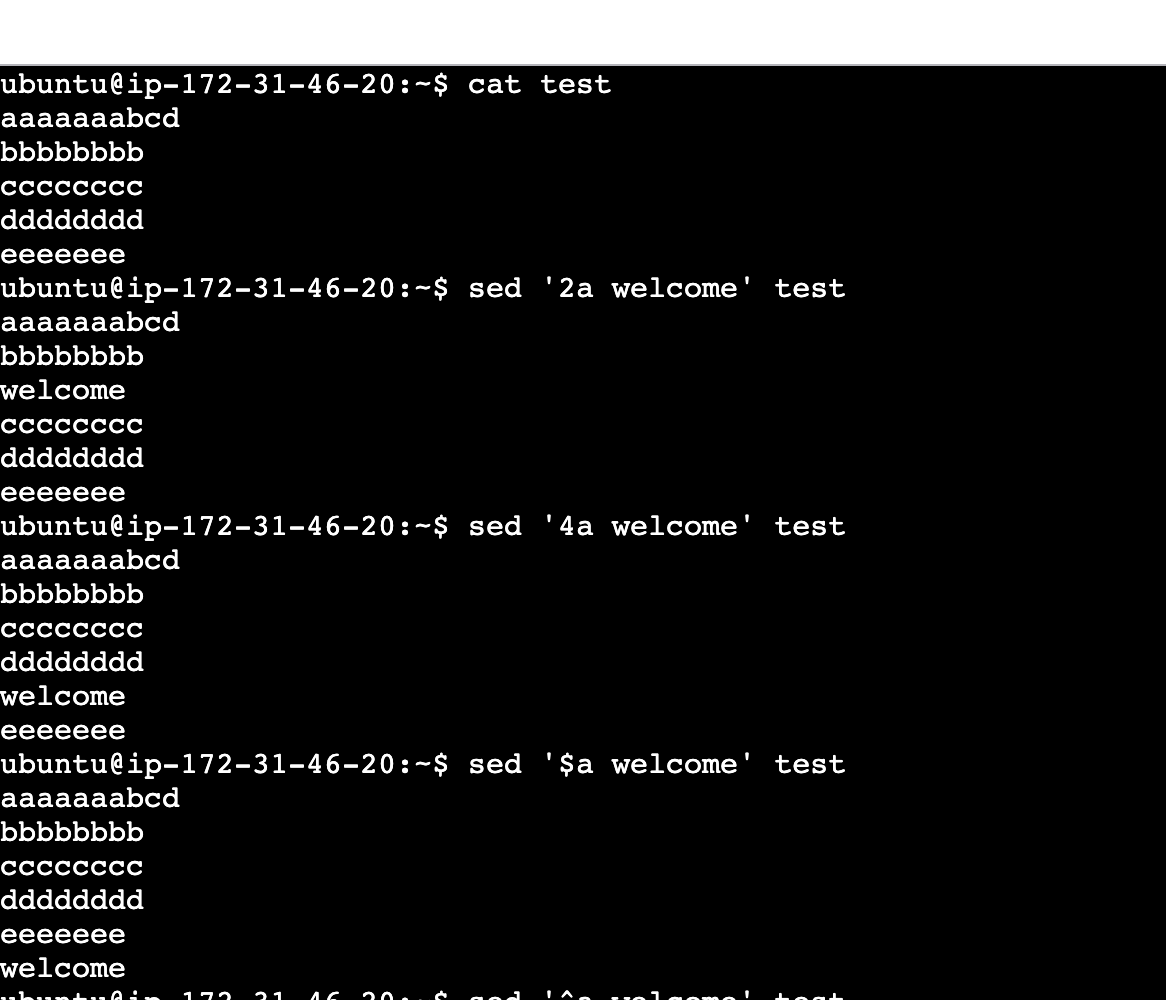
**how to add the text at the specific line number?**

syntax:

sed -i “linenumber a text” filename

a-- insert

sed -i “2a Welcome” file1



**Assignment:**

**write a script to display the vehicle which are in red color?**

|  |
| --- |
| #!/bin/bash  #set -x  echo "Please enter the filename:"  read filename  echo "Pleae enter the color you are looking for:"  read color  echo "The vechile names which are in color $color"  while read line  do  car\_color=`echo "$line"|awk -F " " '{print $2}'`  if [ $car\_color == $color ]  then  echo "$line"|awk -F " " '{print $1}'  fi  done<$filename |

o/p:

Please enter the filename:

vehicle

Pleae enter the color you are looking for:

red

The vechile names which are in color red

xuv

Please enter the filename:

vehicle

Pleae enter the color you are looking for:

grey

The vechile names which are in color grey

be6e

**write a script to display the vehicle name and model number whose price are greater than 50000?**

|  |
| --- |
| #!/bin/bash  #set -x  echo "Please enter the filename:"  read filename  echo "Please enter the price you are looking for:"  read price  echo "The vechile names which are equal to or above the price $price"  while read line  do  car\_price=`echo "$line"|awk -F " " '{print $4}'`  if [ $car\_price -ge $price ]  then  echo "$line"|awk -F " " '{print $1,$4}'  fi  done<$filename |

o/p:

Please enter the filename:

vehicle

Please enter the price you are looking for:

1800000

The vechile names which are equal to or above the price 1800000

be6e 2000000

hyrider 2400000

Please enter the filename:

vehicle

Please enter the price you are looking for:

2300000

The vechile names which are equal to or above the price 2300000

hyrider 2400000

**write a script to display the name of the employee, employee id who are working in devops department?**

|  |
| --- |
| #!/bin/bash  echo "Please enter the file:"  read filename  echo "Please enter the department you are looking for:"  read department  echo "$(awk -F "" 'NR==1 {print $0}' $filename)"  while read line  do  department\_value=`echo "$line"|awk -F " " '{print $NF}'`  if [ $department == $department\_value ]  then  echo "$line"|awk -F " " '{print $0}'  fi  done<$filename |

o/p:

Please enter the file:

employee

Please enter the department you are looking for:

Devops

ID Name Role Department

1 Phani Admin Devops

3 Abhi Tester Devops

4 Venu SRE Devops

Please enter the file:

employee

Please enter the department you are looking for:

Dev

ID Name Role Department

2 Anil Developer Dev

**write a script to display the content of a file along with the line numbers?**

|  |
| --- |
| #!/bin/bash  echo "Please enter the filename:"  read filename  echo "$(cat -n $filename)" |

**o/p:**

Please enter the filename:

vehicle

1 xuv red 2015 1000000

2 nexon white 2018 1500000

3 be6e grey 2024 2000000

4 hyrider blue 2023 2400000

5 LnTtractor greem 2014 700000

**write a script whether the given number is even or odd?**

|  |
| --- |
| #!/bin/bash  echo "Please enter the number:"  read number  val=`expr $number % 2`  if [ $val -eq 0 ]  then  echo "The given number $number is even"  else  echo "The given number $number is odd"  fi |

**o/p:**

Please enter the number:

66

The given number 66 is even

Please enter the number:

5

The given number 5 is odd

Please enter the number:

1

The given number 1 is odd

**write a script whether the given number is divisible by 7 or not?**

|  |
| --- |
| #!/bin/bash  echo "Please enter the number:"  read number  val=`expr $number % 7`  if [ $val -eq 0 ]  then  echo "The given number $number is divisble by 7"  else  echo "The given number $number is not divisble by 7"  fi |

**o/p:**

Please enter the number:

7

The given number 7 is divisble by 7

Please enter the number:

14

The given number 14 is divisble by 7

Please enter the number:

34

The given number 34 is not divisble by 7

**Day 20: Bash/Shell Scripting**

**write a script to display all the files which contain a specific pattern from the current directory as well as the sub directory if any of the file doesn’t contain a pattern, then display the error message?**

|  |
| --- |
| #!/bin/bash  echo "Please enter the pattern"  read pattern  grep -rl "$pattern" \*>filenames  if [ $? -eq 0 ]  then  echo "We have found the below files"  cat filenames  else  echo "The file doesn't exist"  fi |

**o/p:**

Please enter the pattern

Phani

We have found the below files

3userdefinedvaraible.sh

4readinput.sh

data

employee

studentsdata

Please enter the pattern

Anil

We have found the below files

employee

studentsdata

**write a script to display all the files which doesn’t contain a specific pattern from the current directory as well as the sub directory if any of the file contain a pattern, then display the message?**

|  |
| --- |
| #!/bin/bash  echo "Enter pattern:"  read pattern  ls>test  allfiles=$(ls)  pattern\_match=$(grep -il "$pattern" \*)  for item in $allfiles  do  for item1 in $pattern\_match  do  if [ $item == $item1 ]  then  delete=$(grep -n "$item1" test|awk -F ":" '{print $(NF-1)}')  $(sed -i "${delete}d" test)  fi  done  done  echo -e "\nThe files which doesn't contain the provided pattern are:"  cat test |

**o/p:**

Enter pattern:

/bash

grep: Testdir: Is a directory

The files which doesn't contain the provided pattern are:

1.html

2.html

3.html

4.html

5.html

6.html

Testdir

data

employee

filenames

readline

reversefile

studentsdata

test

vehicle

**For loops:**

**syntax:**

for i in array\_name

do

statement

done

**write a script to add the set of numbers?**

|  |
| --- |
| #!/bin/bash  set -x  number="10 20 30 40"  sum=0  for i in $number  do  #echo "The value of i is $i"  sum=`expr $sum + $i`  done  echo "The sum is $sum" |

**o/p:**

The sum is 100

**write a script to rename all the text files into html files?**

|  |
| --- |
| #!/bin/bash  ls \*.txt>filenames  while read line  do  name=`echo "$line" |awk -F "." '{print $1}'`  #echo $name  #echo $line  echo "line value $line"  mv $line $name.html  done<filenames |

**Assigment:**

**Write a script to find the factorial for the set of numbers?**

|  |
| --- |
| #!/bin/bash  #iset -x  number="5 6 8 10"  for value in $number  do  sum=1  initial\_value=1  while [ $initial\_value -le $value ]  do  sum=`expr $sum \\* $initial\_value`  initial\_value=`expr $initial\_value + 1`  done  echo "The factorial of $value is $sum"  done |

**o/p:**

The factorial of 5 is 120

The factorial of 6 is 720

The factorial of 8 is 40320

The factorial of 10 is 3628800

**write a script to reverse the content of a file without using tac, which is last line will be displayed as first line, last but one line will be displayed as 2nd line and so on?**

|  |
| --- |
| #!/bin/bash  echo "Please enter the file name:"  read filename  echo -e "\n"  number\_of\_lines=$(wc -l $filename|cut -d " " -f1)  while [ $number\_of\_lines -gt 0 ]  do  $(head -$number\_of\_lines $filename|tail -1>>reversefile)  number\_of\_lines=`expr $number\_of\_lines - 1`  done  echo -e "The orginal file is:\n"  cat $filename  echo -e "\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n"  echo -e "The reverse file is:\n"  cat reversefile  echo -e "\n" |

**o/p:**

Please enter the file name:

employee

The orginal file is:

ID Name Role Department

1 Phani Admin Devops

2 Anil Developer Dev

3 Abhi Tester Devops

4 Venu SRE Devops

5 Satish SSE Support

6 Hari Admnin IT

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

The reverse file is:

6 Hari Admnin IT

5 Satish SSE Support

4 Venu SRE Devops

3 Abhi Tester Devops

2 Anil Developer Dev

1 Phani Admin Devops

ID Name Role Department

**Day 21: Bash/Shell Scripting**

**Write a script to monitor disc usage, if the disc usage reaches the threshold value then I need to send email notification to a concerned user?**

|  |
| --- |
| #!/bin/bash  current\_usage=$(df -h .|awk -F " " 'NR>1 {print $(NF-1)}'|cut -c 1-2)  echo $current\_usage  if [ $current\_usage -gt 35 ]  then  echo "The usage has been crossed"  echo "memory reached thresold value" | mail -s "Clean up the memory" phani.nandigam@moengage.com  fi |

**Assignment:**

Configure the mail server and execute the script

**stress:**

**This** is a command-line tool available in Linux to test the performance and reliability of computer systems.

syntax:

stress -c N

N is an integer specifying the number of CPUs to stress.

stress -c 8 --timeout 10

The above command will increase the cpu utilization for 10 seconds

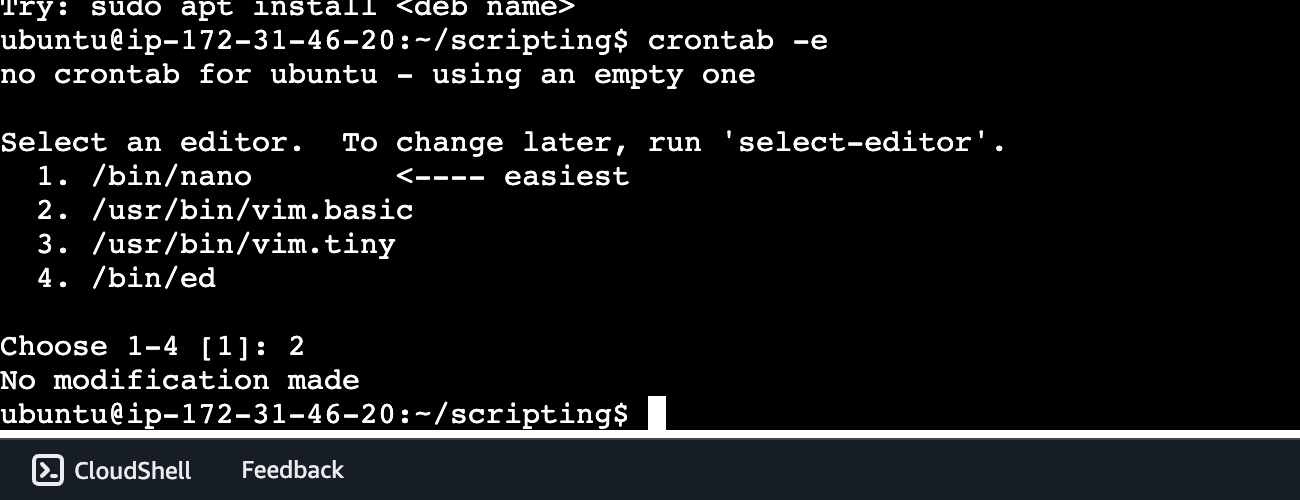
**Write a script to check the CPU utilization and if reached the threshold value of 70% send out an email? (use stress command to increase the CPU utilization)**

|  |
| --- |
| #!/bin/bash  top -b -n 1 > CPUfile  current\_usage=`awk -F " " 'NR==8 {print $(NF-3)}' CPUfile|cut -d "." -f1`  echo $current\_usage  if [ $current\_usage -gt 70 ]  then  echo $current\_usage  echo "The CPU has been increased thresold value $current\_usage, please take actions"|mail -s "CPU Usage has reached thresold" phani.nandigam@moengage.com  fi |

**crontab : (**is used for Scheduling and Automating Tasks**)**Crontab is a powerful utility that is used for Scheduling and Automating Tasks in Unix-like operating systems. It facilitates the users to run the scripts or Linux Commands at specified times and intervals. It is ideal for repetitive tasks such as system maintenance, backups, and updates.

to add the content, use crontab -e

crontab -l is to view the content



\* \* \* \* \* path of the file

**1st \*** -- It represents minutes

**\*** -- it represents every minute

if provide the value as 15, then the script will execute for 15th minute of every hour

example: 8:15, 9:15, 10:15,11:15 …..

**Note: This is not an interval time**

If we want to execute the script for every 10 minutes, then we need to use the below value

\*/10

**2nd \*** -- It represents the hour

**\*** -- it represents every hour

**8** – the script will run at 8 AM

**22**- Th script will run at 10 PM

\*/2 – this will execute the script every 2 hours

**3rd \*** -- it represents date

08 – this will execute the script on day 8

**4th \*** -- it represents month

1. Jan
2. feb
3. march

**5th \*** -- it represents the day of the week

00-sun

01-Monday

02-Tuesday

I want to run the script on 14th September at 11 PM

00 23 14 09 \* /home/ubuntu/scripting/scriptname.sh

I need to execute the script on every Monday at 6 AM

00 06 \* \* 01 /home/ubuntu/scripting/scriptname.sh

I need to execute the script on Monday and Wednesday at 11:20 AM

20 11 \* \* 01,03 /home/ubuntu/scripting/scriptname.sh

I need to execute the script on Monday to Wednesday at 11:20 AM

20 11 \* \* 01-03 /home/ubuntu/scripting/scriptname.sh

execute the script for every 20 minutes

\*/20 \* \* \* \* /home/ubuntu/scripting/scriptname.sh

**crontab**

This is a scheduler which is used to schedule to scripts in linux

crontab -l – this will list the scheduled scripts in crontab

crontab -e

add the automation at the end of the line

18 8 \* \* \* /home/ubuntu/scripting/1display.sh > /home/ubuntu/scripting>Automatescript.txt

this will run at 8 AM at 18th minute and will write the output to the Automatescript.txt

I need to execute the script for every minute of 6 AM on Monday

\* 06 \* \* 01 /home/ubuntu/scripting/scriptname.sh

**Day 22: Bash/Shell Scripting**

write a script to monitor the set of services, If any of the services are not running, then it should send a mail notification to the concerned user

|  |
| --- |
| * #!/bin/bash * service\_name="ss jenkins" * for i in @service\_name * do * ps -C "$i" * if [ $? -ne 0 ] * then * echo $i>>services * fi * done * if [ -s services ] * then * cat services | mail -s "The services are not running" phani.nandigam@moengage.com * rm -rf services   fi |

Note: if the file contains the data, then it will go through the if

Here -s will verify if the file contains the content or not.

**write a script to clean up the old builds and to retain the recent n number of builds and delete all the builds?**

|  |
| --- |
| * #!/bin/bash * count =3 * ls -lrt|awk -F “ “ ‘NR>1 {print $NF}’>file1 * total=`cat file1|wc -l` * delete=`expr $total - $count` * head -$delete file1| xargs rm -rf |

**switch cases**

**syntax:**

case $variable\_name in

value1) statement1

;;

value2) statement2

;;

value3) statetment

;;

.

.

.

.

default) statement

esac

**here ;; refers to the break**

if the input is integer

1)

if the value is string

‘1’)

**Assignment:**

**Write a script with the below use case:**

**mon find the file which were modified 10 days ago**

**tue list all the files which contain a pattern**

**for Wednesday list all the files which doesn’t contain a pattern**

**Thrusday check whether the give path is file or a directory or a link**

**Friday find the empty files , if the file is not found, then display as empty file doesn’t exist**

**for sat and sun: no task**

|  |
| --- |
| * #!/bin/bash * echo "Please enter the value:" * read value * date '+%A' * echo $date * case $value in * 'Monday') echo "The files which were modified 10 days ago" * find . -type f -mtime +5 * ;; * 'Tuesday') echo "Please enter the pattern:" * read pattern * echo "Below files conatin the provided pattern" * grep -i "$pattern" \* * ;; * 'Wednesday') echo "Please enter the pattern:" * read pattern * echo "Below files doesn't conatin the provided pattern" * grep -iL "$pattern" \* * ;; * 'Thrusday') echo "Please enter the path" * read path * if [ -L $path ] * then * echo "The given path is a link" * elif [ -f $path ] * then * echo "The given path is a file" * elif [ -d $path ] * then * echo " The give path is a directory" * else * echo "We didn't find the file" * fi * ;; * 'Friday') lines=`find . -type f -empty|wc -l` * if [ $lines -eq 0 ] * then * echo "No empty files" * else * echo "We have found the $lines empty files" * find . -type f -empty * fi * ;; * 'Saturday' | 'Sunday') echo -e "No Task for today \n Enjoy the Weekend" * ;; * \*) echo "Please enter the correct value" * ;;   esac |

**Have you written any scripts?**

Yes, I have written many scrips and some of them are

1.Written a script to monitor the usage of the server memory, if the server memory reaches its threshold

value (70%). It will send an email to the concerned person.

2. Written a script to monitor the services, if the service stop automatically it has to send an email notification to the concerned person.

3. written a script to clean up the old builds (we need to retain new (latest 10) builds and delete old builds)