**ShellScript:-** Writing more Linux commands into a single file and working on that and executing a file is called shell script.

**Shell Script Keywords**

**$#:** Display the total number of arguments.

**$\*:** All arguments passed to a script.

**$?:** Status of last executed command whether it is failed/passed.

**$!:** Most recent executed command.

**$$:** Display the present PID.

**$@:** All arguments passed to a script are stored in array format.

**$0:** Is used to display the name of script.

**# :** Used to commenting a code in a script.

**Notations to be used in the if statements**

* Conditions
* **-eq** = equal
* **-gt** > greater than
* **-ge** >= greater than or equal to
* **-lt** < lesser than
* **-le** <= lesser than or equal to
* **-ne** != not equal

**Zombi Process**: Zoombi process is dead process/ inactive process. If the child process doesn’t return the status of the process to its parent process, then child process becomes zombi process.

­­­­­­­­­­­­­­­­­­­­­­­­­­­­­­­

**Orphan Process:** When the parent process dies before the child process return the status to its parent process, then that child process becomes orphan process.

**Crontab ;-** Is a scheduler, which is used to schedule scripts on linux server.

If ec2 doesn‘t consists crontab,

Then (sudo service crond start)

Crontab –l it will display list of cronjobs

Crontab –e it will runs the cron job

Crontab –l –u specify the user name

Crontab –ir u can remove or delete the jobs

**Syntax of Crontab Jobs**

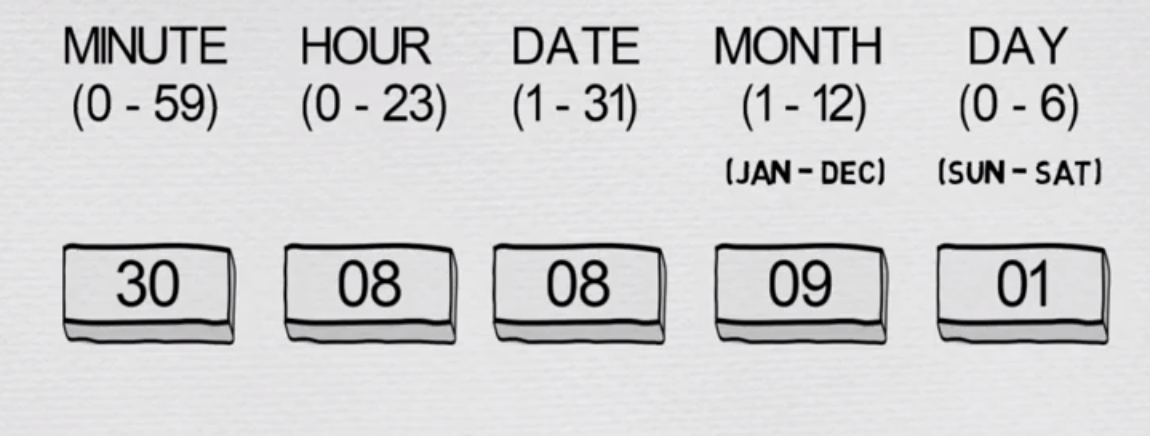
\* | \* | \* | \* | \* /script path

Mins |hrs |date|month|day of week

**Day of the week**

**Sun: 00** **Mon: 01** **Tue :02** **Wed: 03** **Thu: 04** **Fri: 05**  **Sat:06**

Timing should be given based on 24 hrs.



**Schedule to run the script on specified date and time**

**12th june at 10 am 13th june at 4 pm**

00 10 12 06 \* /script path 00 16 13 06 \* /script path

**Everyday at 2pm 03.30 pm only on tue**

00 14 \* \* \* /script path 30 15 \* \* 02 /script path

**Mon-fri at 4pm**

00 16 \* \* 01-05 /script path

**Mon and Wed at 5.30pm (when mins comes with hour, then mention the min with hours)**

 30 17 \* \* 01,03 /script path

**Mon every one hour**

 00 \* \* \* 01 /script path

**Mon every 30 mins**  (when mins are individual, then \*/30)

\*/30 \* \* \* 01 /script path

**Mon at 4pm, script should run at every 5min**

\*/5 16 \* \* 01 /script path

[**https://crontab.guru/**](https://crontab.guru/)

**05.15pm one the friday**

15 17 \* \* 05 /script path

**Sat 10pm**

00 22 \* \* 06 /script path

**Every month 1st at 11 am**

1. 1 01 \* \* /script path

**To run memch.sh for every min, Crontab –e**

\* \* \* \* /memch.sh

**Mails in Linux :- Mail command** it Is used to send email or file or information to the specific.

**email-id.**

* Mail –s “Subject Of the Mail”‖ “Body of the mail”‖ –c recipients

Mail –s “memory reached optimum level”‖

“ Memory reached 75% threshold, immediate action required to release the space”

–c [remo@ibm.com](mailto:pradi@gmail.com) [sonu@ibm.com](mailto:pradeep@gmail.com) < filename

**Problem**

* Builds were failing due to memory issue it was giving an error like ―no space on disk.
* To resolve this problem/issue, I have written a script to check disk memory and send a email notification, saying that memory is 75%, and please take appropriate action.

**Solution**

* #!/bin/bash
* Mem=df –h | tail -1 | awk –F “{print $4}” | sed ‘s/%/ g’
* If [ $mem –gt 70 ]
* Mail –s ―memory reached‖ –c remo@gmail.com sonu@gmail.com < filenameFi

**How to run the shell script in Debug mode ?**

Sh hello.sh this is the normal way of running the shell script. So if you want to run in debug mode we are going to use the –x.

Sh –x hello.sh if excute this one we are going to run in the shellscript.

**What is the command which can use to display all the system defined env command.**

Prientenv command either we can execute.

**What do you mean by command line arruguments?**

Command line argument nothing but while running the script passing the values in the command prompat as command line arugument

**What do you mean by string ?**

Set of characters eclosed in singl equote called as string

**Shell scripting**

**Vi hello.sh**

#!/bin/bash

(This string is called she bang. It invokes bashrc shell)

* Echo “hello word”

(Echo is a print statement)

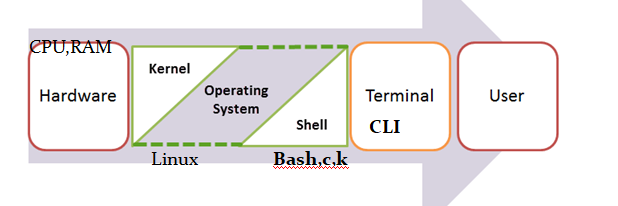
* Esc:wq!

(To save and quit from the file)

./hello.sh or sh hello.sh or bash hello.sh

To execute the file

**Shell Basic Archetecture**



**Vi ex1.sh**

* #!/bin/bash
* echo "hello"
* echo "Welcome to the shell scripting"
* Pwd
* Mkdir dir
* Echo $0

**Vi ex2.sh**

* #!/bin/bash
* echo " Hello "
* echo " Welcome to the shell script “
* pwd
* df -h

**Vi ex3.sh  
(Variables used in the shell script)**

* #!/bin/bash
* var1=mohan
* var2=953812345
* var3=Devops-Architect
* echo " my name is : " $var1
* echo " my number is : " $var2
* echo " I am also known as : "$var3

**Vi ex4.sh  
Command line arguments**

* #!/bin/bash
* echo "my name is : $1 "
* echo "my city is :$2"
* Echo "my favorite subject is :$3"
* echo "my file name is : $0”
* **./ex4.sh Sham Bangalore Maths**

**Vi ex5.sh**

* #!/bin/sh
* echo “what is your name?“
* read name
* echo “how are you ,$name?”
* Read remark
* echo “I am $remark too !“
* Note: read command is used to capture input given in terminal into variable “name”
* $name –Display the value of name variable

**If**

If statement (Conditional statement)

If [ condition ];then

Statement

fi

**Example of If Statement**

If [Covid = Negative ];then

Echo “Go to Office”

Fi

**If else**

If [ condition ];then

Statement

Else

Statement

fi

**Example of If Statement**

If [Covid = Negative ];then

Echo “Go to Office”

Else

Echo “work from home”

Fi

**Example**

* if [ $1 -gt 10 ];then
* echo "$1 is greater than 10"
* else
* echo "$1 is lesser than 10"
* fi
* ./ex.sh 2

**Example of If Else statement**

If [ Balance >5000];then

echo “Credit interest”

else

echo “Debit Penalty”

fi

**If elseif else**

If [condition ];then

Statement

Elif [ condition];then

Statement

Else

Statement

Fi

**Example of if Elseif Statement**

If [Using=fan];then

Echo “Electricity bill 500”

Elif [Using=AC];

then

Echo “Electricity bill 1000 ”

Else

Echo “No Electricity No Bill ☺ ”

Fi

**IfElse Shell Script: To check if the given number is 5**

#!/bin/bash

if [ $1 -eq 5 ];then

echo "Given number is five"

else

echo "Given number is not five"

Fi

./ex 9

./ex 5

**WAS to check the biggest of 2 numbers**

#!/bin/bash

if [ $1 -gt $2 ];then

echo "$1 is biggest number"

else

echo "$2 is biggest number"

Fi

./ex 6 3

./ex 2 4

**To check the execution line by line add set –x (debug mode)**

* #!/bin/bash
* set -x
* if [ $1 -gt $2 ];then
* echo "$1 is biggest number "
* else
* echo "$2 is biggest numbr "
* fi

**To find the biggest of 3 numbers**

* #!/bin/bash
* set -x
* if [ $1 -gt $2 ] && [ $1 -gt $3 ];then
* echo " $1 is big "
* elif [ $2 -gt $3 ];then
* echo " $2 is big "
* else
* echo "$3 is big"
* Fi
* ./ex 3 7 4
* ./ex 3 4 7
* ./ex 7 6 3

**WAS to add 2 numbers**

* #!/bin/bash.
* echo -n "Enter the two numbers : "
* read num1 num2
* sum=`expr $num1 + $num2`
* echo "sum of two numbers is $sum“
* ./add.sh
* 4 5

**Syntax of While loop**

While loop

While [condition ]

Do

Statements

Done

**WAS to print entered number to 1**

* #!/bin/bash
* echo "Enter a number"
* read num
* while [ $num -gt 0 ]
* do
* echo $num
* num=`expr $num - 1`
* Done
* ./numto1.sh
* 5

**WAS to find the factorial of a number**

* #!/bin/bash
* echo "Enter a number"
* read num
* fact=1
* while [ $num -gt 1 ]
* do
* fact=$((fact \* num)) #fact = fact \* num
* num=$((num - 1)) #num = num - 1
* Done
* <https://www.geeksforgeeks.org/shell-program-to-calculate-the-factorial-of-a-number/>

**To find the greatest of 2 numbers and find the difference of it**

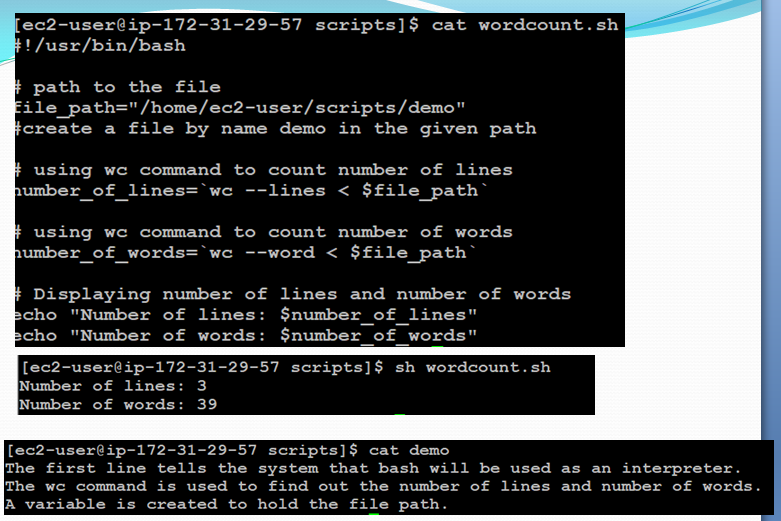
* #!/bin/bash
* if [ $1 -gt $2 ];then
* echo "$1 is the biggest"
* sub=`expr $1 - $2`
* else
* echo "$2 is the biggest"
* sub=`expr $2 - $1`
* fi
* echo "The difference is $sub“
* ./sub.sh 5 6

**WAS to add given number to 1**

* #!/bin/bash
* num=$1
* add=0
* while [ $num -gt 0 ]
* do
* add=`expr $add + $num`
* num=`expr $num - 1`
* done
* echo "$add“

./addnumto1 3

**To count the number of words in each line (Ignore-this script)**



**Keywords in IT**

* EOD
* OOO
* DL
* Domain
* ASAP
* CC/loop
* Offshore
* Onshore / to work in client location
* On which **Platform** did you work?
* RM
* Timesheet
* Swipein swipeout/ tailgate
* KT

Shell script java installed or not

Vi installationjava.sh

Give the user permissions chmod u+x installationjava.sh

Run the script :- ./installationjava.sh

#!/bin/bash

#Excecute the below command to see which version of java has installed , if it is already installed

Java –version

#check the previous command execution status

If [ $? –eq 0 ];then

echo “java has already installed..”

else

echo “java has not installed”

echo “Installing java by using yum command”

sudo yum install java-11-openjdk –y

echo “java has installed successfully”

echo “java version is : ”

java –version

fi

#!/bin/bash

#Excecute the below command to see which version of java has installed , if it is already installed

Java –version

#check the previous command execution status

If [ $? –eq 0 ];then

echo “java has already installed..”

else

echo “java has not installed”

echo “Please type the java package which version you want to install”

read javapackagewithversion

sudo yum install $javaPackagewithVersion -y

echo “java has installed successfully”

echo “java version is: ”

java –version

fi