july 23,2022:

==============

shell scripting:

Q: what is a shell script?

Ans: A shell script is a collection of commands to perform automated tasks in linux/unix OS

Q: How does the shell intepret commands within a shell script?

Ans: The shell intepret commands in chronological steps one line at a time

Q: what is a shebang? #!/bin/bash

Ans: A shebang is an indicator of what shell will be use to run the commands within the script

Q: why do we need to learn shell scripting?

Ans: we learn shell scripting to avoid repeating tasks and safe time of administration

Scripting in action:

====================

steps to write a shell script

- create a file with extentsion .sh for example sudo vi scriptname.sh

- enter the shebang first line for example #!/bin/bash, #!/bin/sh

- enter your commands you want to automate for example 'ls' to create a script to automate ls

Puting the above steps into a code give us these;

#!/bin/bash

ls

bash -x test.sh --> use to debug a running script

sed -n '/home/p' /etc/passwd --> use to search for word home in files and print it( can also modify )

sed 's/akudo/hello/g' /etc/passwd --> search for the word 'akudo' from /etc/passwd then replace with 'hello'

awk '{print $1 " " $2}' myfile.txt

What is the difference between sed and awk commands?

ans:

awk is use to search data within a file and print it as output while sed is use to search characters and modify them

awk is more robust than sed for file administration

script to automate system resources checks;

sudo vi monitor.sh

#!/bin/bash

echo "script to check system resources"

sudo logname # to check who log into the server

sudo whoami # to check who is currently using the shell

sudo df -h # to check the disk free space of the entire server

sudo free -m # to check memory

sudo lscpu # to check cpu informations

NB: when you create a shell script for the first time it is not executable so make executable

run this command

sudo chmod +x scriptname.sh

to run the script you can do it in two ways

1. sh scriptname.sh # not good for company environment

2. ./scriptname.sh

==========================================================================================

special words to remember when writting shell scripts

'echo' is use to print information on the terminal or append, redirect

'echo -n' to print statement and answer on the same line

'read' is use to read input data from the shell user

'read -s' is use to hide input data as password,

'read -p' is use to read input data and print data at same time

y='string' is use to defined string variable

y= number no need for quotes

$variable is use to read the value of a variable

#this script will help us create a file and assign date created with user who runs the script

#!/bin/bash

echo -n " Please Enter the filename: "

read filename

today=`date +%m-%d-%y`

user=`whoami`

touch $filename.$user.$today

echo " thank you "

echo " welcome to Etech Consulting " >> $filename.$user.$today

df -h $filename.$user.$today

# script to design an automated calculator

#!/bin/bash

echo -n " Please enter the first number: "

read num1

echo -n " Please Enter the second number: "

read num2

sum=$(echo "scale=4; $num1 + $num2" | bc)

dif=$(echo "scale=4; $num1 - $num2" | bc)

mult=$(echo "scale=4; $num1\*$num2" | bc)

divi=$(echo "scale=4; $num1 % $num2" | bc)

float=$(echo "scale=4; $num2/$num1" | bc)

echo " The sum is: $sum "

echo " The difference is: $dif "

echo " The division is: $divi "

echo " The multiplication is: $mult "

echo " This is awesome we just build our first calculator for Etech Consulting Engineers "

echo " The best scripts will be rewarded a sum of \$20 by cashApp "

echo " The floating point answer is: $float "

#!/bin/bash

#this script will be use to create git repositories

echo -n " please enter the directory name: "

read directory

sudo mkdir $directory

git init $(pwd)/$directory

echo " use your command line terminal to access $directory directory"

echo "And make sure it a git repository to test the script"

ls -a $directory | grep .git #this line helps you to detect a git repository special file

echo " make sure you see the .git file above to confirm that you actually created the repo "

echo " This is a property of Etech ConsultingDevopsBooster course!!! "

The if...else...fi statement is the next form of control statement that allows \

Shell to execute statements in a controlled way and make the right choice.

syntax:

if [ expression ]

then

Statement(s) to be executed if expression is true

else

Statement(s) to be executed if expression is not true

fi

example:

Atmcard.sh

#!/bin/bash

pin="2022"

echo -n " please enter your pinnumber:"

read -s pinnumber

if [[ "$pinnumber" -eq "$pin" ]]; then

echo "Welcome to bank of America"

else

echo "Sorry incorrect pin: please try again!!"

fi

#!/bin/sh

a=10

b=20

if [ $a == $b ]

then

echo "a is equal to b"

else

echo "a is not equal to b"

fi

#script to analysed special variables

#!/bin/sh

echo "File Name: $0"

echo "First Parameter : $1"

echo "Second Parameter : $2"

echo "Quoted Values: $@"

echo "Quoted Values: $\*"

echo "Total Number of Parameters : $#"

output:

$./test.sh Zara Ali

File Name : ./test.sh

First Parameter : Zara

Second Parameter : Ali

Quoted Values: Zara Ali

Quoted Values: Zara Ali

Total Number of Parameters : 2uykuyk

#!/bin/bash

#Author:Engr Rudolph

#Company: Etech Consulting llc

#Script will be use to test if tomcat website is up and running and take actions

if curl -s http://etechdemo.eastus.cloudapp.azure.com:9000/projects; then

echo "tomcat is up and running"

exit 0

else

echo "tomcat is down"

exit 1

echo "The exit status was captured to be `echo $?`"

#api call connection to azure function to trigger email notifications

fi

#this script will be used to check if a user exist in a linux server and

#Also check if the user directory is present or not

#Before we test this script let's make sure we have already created the script

#To add user

#!/bin/bash

echo -n " Please enter the username to be checked: "

read username

echo " Search Engine Activated!!..searching.. "

if grep $username /etc/passwd; then

echo " The user account for $username exist in the system.. "

echo " The user $username is part of Etech ConsultingDevops course "

elif ls -d /home/$username/; then

echo " The directory for $username exist in the system "

echo " Even though the $username acount does not exist "

else

echo " The user directory does not exist in the system "

echo " The user account for $username does not exist "

fi

=======================================================================================

df -h where h stands for human readable format

free -m

lscpu

when you type 'vi'

press 'i' on your keyboard (insert)

commands.....

press 'esc'

press 'shift' + ':'

:wq! to save and quit the file

:q! to quit without saving

work: write a script to check who is log into the system

/tmp/monitor.login

/root/logincheck.sh

systemctl stop cron = to control cron jobs (start,stop, restart,status)

watch ls = to see how jobs are running in a directory for every 2 seconds

watch cat <filename> = to see real time processes in a file

n jobs running and stop the deamon all the other cron jobs in same system are dead

/etc/crontab = this is the file that you use to configure cronjobs

Ex: configure a simple cronjob to continousely print "Thank you" into a file \

called check.log every minutes

sudo apt update -y to update your server

sudo apt update -y for ubuntu linux

sudo yum update -y for Redhat linux

solutions:

1.create the file script called print.sh

etch.txt.root.

login=`logname`

echo $login

sudo yum install bc <enter>

sudo apt install bc <enter> where bc stands for bash calculator

``

$() == ls $(pwd)

ls `pwd`

text=$(echo "thank you") $output = text= thank you

$text

\$5000

$5 000