Shell scripting:

vi filename.sh

press i – to got to insert mode

enter the commands

press esc, : , wq

to run/execute the script:

sh filename.sh

./ filename.sh

echo – prints the content within “ ”

Variable – is used to hold the data

a=10

b=20

result=a+b;

To evaluate any expression in shell script- give it in back quotes- ` `

Cmd line arguments: values passed at the time of program execution and

read in the program using language specific syntax.

to take input from command line arguments- $1, $2, $3….$n

to count no of cmd line arguments- $?

#This is first Shell Script

echo "Hello world"

echo `date`

--count the no of users in the system

#count no of users

x=10

echo "value of x: $x"

users=`who | wc -l`

echo "no of users: $users"

--Count the last few lines with word Exception in a log file

#count no of line with Exception word in log file

fileName=$1

lines=` tail -8 $fileName | grep Exception | wc -l `

echo "no of lines: $lines"

--create a directory. Create files, add content to file, display the content

you added

#create directory and add files, content to files

#create directory and add files, content to files

echo "no of cmd line args: $#"

mkdir $1

cd $1

# method 1 to create a file

touch $2

touch $3

#method 2 to create a new file

> file3.txt

#method 3 to create a new file

>> file4.txt

# add content to file

cat << EOF >> $2

This is Unix class 2

we are doing shell scripts

This is Jab batch

EOF

# display content of file

echo "\*\*\*FILE CONTENT\*\*\*\*"

cat file1.txt

echo "\*\*\* LIST ALL FILES\*\*\*"

ls -l

find – used to find the particular file/files in a given directory

find newSubDir3 -name "log\*" | wc -l

env – list of env variables

echo $envVarName

ps – lists all active processes

top- list all process details like cpu, memory, time etc

kill process id- to end the process

netstat- to see active connections to system

Arithmetic operators:

+

-

\\*

%

Conditional operators:

-lt - <

-gt - >

-le - <=

-ge - >=

conditional statements:

if(condition is true){

do some logic

}else{

Do some other logic

}

If

Switch case

loop statements: Do same logic again and again by iterating the loop

print all even numbers between 1 and 100

1…100

For

while

a=$1

b=$2

sum=` expr $a + $b `

echo "sum is : $sum"

mul=` expr $a \\* $b `

echo "multiplication result is :$mul"

---if number is div by 3 print hello

if number is divisible by 5 print world

if it is not div by 3 or 5 print hello world

#number div by 3 or 5

input=$1

if [ `expr $input % 3` -eq 0 ]

then

echo "hello"

elif [ `expr $input % 5` -eq 0 ]

then

echo "world"

else

echo "hello world"

fi

one variable value to be compared with multiple other values- switch case is preferred over if

as it makes code more readable

--Display the name of the week day based on input

#switch case example

input=$1

case $input in

"sun") echo "It is Sunday";;

"mon") echo "It is monday";;

"tue") echo "It is tuesday";;

\*) echo "Its SUNDAY";;

esac

#for loop example

for i in 1 2 3 4 5

do

echo "i is $i"

done

or

// second way for loop- more dynamic

n=$1

for (( i=1; i <= $n; i++ ))

do

echo “$i”

done

--Print the maths table for given number

2 \* 1= 2

2 \* 2 =4

…… 2\*10=20

#while loop

input=$1

n=1

while [ $n -le 10 ]

do

echo " $input \* $n = `expr $input \\* $n` "

n=`expr $n + 1`

done

Sed command in unix: Its stream editor to modify files:

http://www.folkstalk.com/2012/01/sed-command-in-unix-examples.html

--

Assignment:

Create a file in a directory, move the file to home directory and display the content in dir

Sort the first five lines in a given file

Sample script::

#!/bin/bash

#### accepts the number of files to be generated by each thread

numoffiles=$1

runid=`date +%H%M%d`

touch ./awsls/ebay\_aws\_ls$runid.sh

chmod +x ./awsls/ebay\_aws\_ls$runid.sh

mkdir ./files$runid

mkdir ./logs$runid

echo 'numoffiles is ' $numoffiles

echo "Runid is:" $runid

###./ebay.sh $numoffiles $threadcount $runid> ./logs$runid/ebay$threadcount$$.out 2> ./logs$runid/ebay$threadcount$$.out &

for (( i=1; i <= $numoffiles; i++ ))

do

myRandom=$runid$i

echo 'random number is ' $myRandom

filename=EBAY\_dropshiporderrequest\_ORD"$myRandom"\_v1.xml

echo $filename

sed "s/CUSTOMERORDER/ORD$myRandom/" EBAY\_dropshiporderrequest\_1order1line.xml > $filename

mv $filename files$runid

done

aws s3 cp files$runid s3://b2b-oc-dropship-inbound-performance/ --recursive --profile PreProduction