**Ashish\_Quiz6**

All Shell scripts should display usage function, they should also validate input arguments to be correct, use fuctions and recursion as much as possible.

1. Write a shell script to find the number of lines in a list of files using sed.

You should write a for loop to go through all files in a directory and then count the number of lines in each file, display it as:

./script <Full path to directory>

File1 has 45 lines

File2 has 20 lines

2 Files in total, 65 lines in total

Ans:

2. Write a shell script to substitute one pattern for another in a text file.

./script.sh oldpatternnewpattern

Ans :

#!/bin/bash

OLD="xyz"

NEW="abc"

DPATH="/home/you/foo/\*.txt"

BPATH="/home/you/bakup/foo"

TFILE="/tmp/out.tmp.$$"

[ ! -d $BPATH ] && mkdir -p $BPATH || :

for f in $DPATH

do

if [ -f $f -a -r $f ]; then

/bin/cp -f $f $BPATH

sed "s/$OLD/$NEW/g" "$f" > $TFILE && mv $TFILE "$f"

else

echo "Error: Cannot read $f"

fi

done

/bin/rm $TFILE

3. Write a shell script to print complete pathname associated with pid. User has to pass the PID from command line.

./script.sh PID

Ans :

#! /bin/sh

echo "$(cd "$(dirname "$1")"; pwd)/$(basename "$1")"

4. Write a shell script to print all users on system using awk.

Hint: Learn what is /etc/passwd file in Linux

Ans:

#!/bin/bash

# Name: listusers.bash

\_l="/etc/login.defs"

\_p="/etc/passwd"

5. Write a shell script to list the frequency of words used in a file.

Hint: Sort and uniq commands will help.

Ans: function wordfrequency() {

awk '

BEGIN { FS="[^a-zA-Z]+" } {

for (i=1; i<=NF; i++) {

word = tolower($i)

words[word]++

}

}

END {

for (w in words)

printf("%3d %s\n", words[w], w)

} ' | sort -rn

}

6. Write a script to take backup of files changed in last 24 hours and archive them.

Hint: Read the Find command tutorial in Linux folder. We typically take backups of a folder by “tar”-ring the entire folders.

Ans :

favorite

I have written this backup script that looks in a file and copies recent files into a folder.

#!/usr/bin/bash

# the number of days to do the backup for.

days=7;

# the files to backup.

location[0]='/opt/location'

# the location to copy the file to

copyLocation='/users/me/Backup/firstBackupScriptTry'

# preform the back up

for i in ${location[\*]}

do

find $i \! -name '\*.class' -mtime -$days \! -type d -exec cp {} $copyLocation \;

done

7. Write a shell script to determine if a particular service is active or not. For eg: if SSH service is active it should display yes and vice versa. Use netstat ,ps commands etc

Ans:

# Check if gedit is running

if pgrep "xxx" > /dev/null

then

echo "Running"

else

echo "Stopped"

fi

8. Write a shell script to remove spaces from filenames and replace it with underscore

Hint: you can use mv command to re-name files

Ans:

#!bin/bash

find $1 -name "\* \*.xml" -type f -print0 | \

while read -d $'\0' f; do mv -v "$f" "${f// /\_}";

done.

9. Write a shell script which prints the df output in more formatted way as below

Filesystem Size Used Avail Capacity Mounted

/dev/sda1 446.71G 18.11G 405.88G 5% /

udev 10M 0 10M 0% /dev

tmpfs 1.14G 9.16M 1.13G 1% /run

10. Write a shell script to summarize available disk space and present in a logical and readable fashion

Ans:

#!/bin/sh

# diskspace - summarize available disk space and present in a logical

# and readable fashion

tempfile="/tmp/available.$$"

trap "rm -f $tempfile" EXIT

cat << 'EOF' > $tempfile

{ sum += $4 }

END { mb = sum / 1024

gb = mb / 1024

printf "%.0f MB (%.2fGB) of available disk space\n", mb, gb

}

EOF

df -k | awk -f $tempfile

exit 0

11. Write a shell function to rename .txt files to .text

Ans : #!bin/Bash

for f in \*.txt

do

[ -f "$f" ] && mv "$f" "${f%txt}text"

done