Terminal Command Practice Problem

Problem Statements

1. Get user info from /etc/passwd and change ownership of user's home directory.

a. View /etc/passwd

```
$ ls -al /etc/passwd
```

ls: cannot access '/etc/passwd': No such file or directory

Note: In this section, for all the questions commands are not working.

2. Moves files from one folder to respective folder

a. touch abc.txt def.txt ghi.txt.txt jkl.txt

b. ls *.txt

```
abc.txt def.txt jkl.txt
```

c. Base Name

```
for file in `ls *.txt`;
do
 baseName=` echo $file | awk -F. '{print $1}'`;
 echo $baseName;
done
```

Extension Name

```
for file in `ls *.txt`;
do
  extensionName=` echo $file | awk -F. '{print $2}'`;
  echo $extensionName;
done
```

d.
```
for file in `ls *.txt`;
do
  folderName=` echo $file | awk -F. '{print $1}'`;
  mkdir $folderName;
done
```

e.
```
for file in `ls *.txt`;
do
  folderName=` echo $file | awk -F. '{print $1}'`;
  mv $file $folderName
  done
```

3. Append current date to all log files name which has extension .log.1 from a folder.

a. touch abc.log.1 def.log.1 ghi.log.1 jkl.log.1 mno.log.1

b. ls *.log.1

c. Base Name

```
for file in `ls *.log.1`;
do
baseName=` echo $file | awk -F. '{print $1}'`;
echo $baseName;
done
```

Extension Name

```
for file in `ls *.log.1`;
do
extensionName=` echo $file | awk -F. '{print $1}'`;
echo $extensionName;
done
```

d. command date +%d-%m-%Y

e. echo abc.log.1-"`date +"%d-%m-%Y"`"

f.

```
for file in `ls *.log.1`;
    do
     echo $file-"`date +"%d-%m-%Y"`"
    done
```

4. Archive the files from /var/log folder have modified 7 days ago and move it to your backup folder

a. find $DIR -mtime -7 -type f

b.
destination="C:\Users\rites\OneDrive\Desktop\Bridgelabz\TerminalCommands\temp\backup"

for file in `find /var/log -type f -mtime +7`

do

cp $file $destination

done

5. Check if folder exists or not. If not present, create it.

#!/bin/bash -x

if [-d jerry]

then

echo "Folder alredy exists"

else

mkdir jerry

echo "folder created"

fi

6. Set environment usersecret=”dH34xJaa23” if its already not set.

a. env | grep usersecret

b. It is not set yet.

c. export usersecret=dh34xJaa23

echo $usersecret

dh34xJaa23

env | grep usersecret

usersecret=dh34xJaa23

7. Find a word “systemd” from all log files in the folder/var/log and print number of occurrence more than 0 against each file.

$ grep -wc "systemd" access.log

8. Create process list table displays process id, parent process id,command name, % of memory consumption, % of cpu utilization.

$ ps –f

9. Print last 4 frequently access urls count in sorted order from /var/log/httpd/access.log

$ cat access.log | awk '{print $11}' | sort | uniq -c |sort -nr | tail -4

4 http://fundoopush-dev.bridgelabz.com/wp-login.php

2 http://fundoopush-dev.bridgelabz.com/.well-known/acme-challenge/4xM- Y1899BrBIJ76P5Er2sj2VhEtyi_DFvfo6xiMXXw

1 https://fundoopush-dev.bridgelabz.com/dashboard/hashtags/animals

1 https://fundoopush-dev.bridgelabz.com/dashboard/archive

10. Print list of last 4 frequently access unique urls at particular hours from /var/log/httpd/access.log

$ cat access.log | awk '{print $4"["$11}' | sort | uniq -c | tail -4

4 [30/Sep/2019:12:09:03["https://fundoopush-dev.bridgelabz.com/login"

1 [30/Sep/2019:12:09:04["https://fundoopush-dev.bridgelabz.com/login"

1 [30/Sep/2019:12:09:09["https://fundoopush-dev.bridgelabz.com/login"

1 [31/Oct/2019:06:45:26[http://fundoopush-dev.bridgelabz.com/wp-login.php

11. Print list of web response code count in the unique sorted order at specific hours.

$ cat access.log | awk '{print $9}' | sort | uniq -c | head -4

3176 200

8 206

26 304

12. Print list of last 10 unique sorted client IP from /var/log/httpd/access.log

$ cat access.log | awk '{print $1}' | sort | uniq -c | sort –nr | tail -4

13. Data analysis / manipulation (awk)

a. Print EmployeeName and TotalPay who has basePay greater than 10000

$ cat data.csv | awk '{if($4>10000) print $2 " " $7}'

b. What is the aggregate TotalPay of employees whose jobtitle is ‘CAPTAIN’

$ cat data.csv| grep CAPTAIN | awk '{sum+= $4}END{ print sum}'

468427

c. Print JobTitle and overtimePay who has Overtimepay is between 7000 and 10000.

$ cat data.csv | awk '{ if($5>7000 && $5<10000) print $3" "$5}'

DEPUTYCHIEF 9737

ASSTDEPUTY 8601

d.Print average BasePay.

$ cat data.csv | awk '{sum+=$4} END {print sum/NR}'

157972