Ques. Move files from one folder to the respective folders.

Solution :

```
$ ls
abc.txt    def.txt    ghi.txt

$ nano moveFile.sh
#!/bin/bash -x
for file in `ls *.txt`
do
        fileName=`echo $file | awk -F. '{print $1}'`
        mkdir $fileName
        mv $file $fileName
done

$ ./moveFile.sh
++ ls abc.txt def.txt ghi.txt
+ for file in `ls *.txt`
++ echo abc.txt
++ awk -F. '{print $1}'
+ fileName=abc
+ mkdir abc
+ mv abc.txt abc
+ for file in `ls *.txt`
++ echo def.txt
++ awk -F. '{print $1}'
+ fileName=def
+ mkdir def
+ mv def.txt def
+ for file in `ls *.txt`
++ echo ghi.txt
++ awk -F. '{print $1}'
+ fileName=ghi
+ mkdir ghi
+ mv ghi.txt ghi
```

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

Solution :

```
$ ls
abc.log.1  appendCurrentDate.sh*  def.log.1  jkl.log.1

$ nano appendCurrentDate.sh
#!/bin/bash -x
currentDate=$(date +'%d%m%Y')
for file in `ls *.log.1`
do
        fileName=`echo $file | awk -F. '{print $1}'`
        newFileName=`echo "$fileName-$currentDate.log"`
        mv $file $newFileName
done

$ ./appendCurrentDate.sh
++ date +%d%m%Y
+ currentDate=31082020
++ ls abc.log.1 def.log.1 jkl.log.1
+ for file in `ls *.log.1`
++ echo abc.log.1
++ awk -F. '{print $1}'
+ fileName=abc
++ echo abc-31082020.log
+ newFileName=abc-31082020.log
+ mv abc.log.1 abc-31082020.log
+ for file in `ls *.log.1`
++ echo def.log.1
++ awk -F. '{print $1}'
+ fileName=def
++ echo def-31082020.log
+ newFileName=def-31082020.log
+ mv def.log.1 def-31082020.log
+ for file in `ls *.log.1`
++ echo jkl.log.1
++ awk -F. '{print $1}'
+ fileName=jkl
++ echo jkl-31082020.log
+ newFileName=jkl-31082020.log
+ mv jkl.log.1 jkl-31082020.log
```

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

Solution :

```
$ nano moveBackup.sh
#!/bin/bash -x
mkdir backup
for files in `find . -mtime +7 -type f -exec ls {} \;`
do
	mv $files backup
done
```

```
$ ./moveBackup.sh
+ mkdir backup
++ find . -mtime +5 -type f -exec ls '{}' ';'
+ for files in `find . -mtime +5 -type f -exec ls {} \;`
+ mv ./.git/config backup
+ for files in `find . -mtime +5 -type f -exec ls {} \;`
+ mv ./.git/description backup
+ for files in `find . -mtime +5 -type f -exec ls {} \;`
+ mv ./.git/HEAD backup
+ for files in `find . -mtime +5 -type f -exec ls {} \;`
+ mv ./.git/hooks/applypatch-msg.sample backup
+ for files in `find . -mtime +5 -type f -exec ls {} \;`
+ mv ./.git/hooks/commit-msg.sample backup
+ for files in `find . -mtime +5 -type f -exec ls {} \;`
+ mv ./.git/hooks/fsmonitor-watchman.sample backup
+ for files in `find . -mtime +5 -type f -exec ls {} \;`
+ mv ./.git/hooks/post-update.sample backup
+ for files in `find . -mtime +5 -type f -exec ls {} \;`
+ mv ./.git/hooks/pre-applypatch.sample backup
+ for files in `find . -mtime +5 -type f -exec ls {} \;`
+ mv ./.git/hooks/pre-commit.sample backup
+ for files in `find . -mtime +5 -type f -exec ls {} \;`
+ mv ./.git/hooks/pre-push.sample backup
+ for files in `find . -mtime +5 -type f -exec ls {} \;`
+ mv ./.git/hooks/pre-rebase.sample backup
+ for files in `find . -mtime +5 -type f -exec ls {} \;`
+ mv ./.git/hooks/pre-receive.sample backup
+ for files in `find . -mtime +5 -type f -exec ls {} \;`
+ mv ./.git/hooks/prepare-commit-msg.sample backup
+ for files in `find . -mtime +5 -type f -exec ls {} \;`
+ mv ./.git/hooks/update.sample backup
+ for files in `find . -mtime +5 -type f -exec ls {} \;`
+ mv ./.git/index backup
+ for files in `find . -mtime +5 -type f -exec ls {} \;`
+ mv ./.git/info/exclude backup
+ for files in `find . -mtime +5 -type f -exec ls {} \;`
+ mv ./.git/logs/HEAD backup
+ for files in `find . -mtime +5 -type f -exec ls {} \;`
+ mv ./.git/logs/refs/heads/master backup
+ for files in `find . -mtime +5 -type f -exec ls {} \;`
+ mv ./.git/logs/refs/remotes/origin/HEAD backup
+ for files in `find . -mtime +5 -type f -exec ls {} \;`
+ mv ./.git/objects/2b/b9e2d5d41336c901d25a51b89a6adaa578c946 backup
+ for files in `find . -mtime +5 -type f -exec ls {} \;`
+ mv ./.git/objects/51/989096178be32d45ad7f769e1d4e1808a65bc5 backup
+ for files in `find . -mtime +5 -type f -exec ls {} \;`
+ mv ./.git/objects/5b/7f01c7e2c8adef8c6c534d0f4e9a717207b65c backup
+ for files in `find . -mtime +5 -type f -exec ls {} \;`
+ mv ./.git/objects/90/a6b30518edfe726633839a6cbac814c2fdf028 backup
+ for files in `find . -mtime +5 -type f -exec ls {} \;`
+ mv ./.git/objects/b6/04c4fde493cb44468855d37d0ae0973164cc67 backup
+ for files in `find . -mtime +5 -type f -exec ls {} \;`
+ mv ./.git/objects/b7/712f41b2e5692497655225612b1ff44514c84e backup
+ for files in `find . -mtime +5 -type f -exec ls {} \;`
+ mv ./.git/objects/b8/8c9d024c0d3a154a2c9ad2ab677f61b2e24ade backup
+ for files in `find . -mtime +5 -type f -exec ls {} \;`
+ mv ./.git/objects/d8/cd9254380212f75de2562efecbd0fea1ec4925 backup
+ for files in `find . -mtime +5 -type f -exec ls {} \;`
+ mv ./.git/objects/f8/23a9e3e8367eea85159049d9dde3fdd8df168f backup
+ for files in `find . -mtime +5 -type f -exec ls {} \;`
+ mv ./.git/packed-refs backup
+ for files in `find . -mtime +5 -type f -exec ls {} \;`
+ mv ./.git/refs/heads/master backup
+ for files in `find . -mtime +5 -type f -exec ls {} \;`
+ mv ./.git/refs/remotes/origin/HEAD backup
+ for files in `find . -mtime +5 -type f -exec ls {} \;`
+ mv ./access.log backup
+ for files in `find . -mtime +5 -type f -exec ls {} \;`
+ mv ./data.csv backup
+ for files in `find . -mtime +5 -type f -exec ls {} \;`
+ mv ./linux_chit_sheet.pdf backup
+ for files in `find . -mtime +5 -type f -exec ls {} \;`
+ mv ./linux_problem_sheet.pdf backup
+ for files in `find . -mtime +5 -type f -exec ls {} \;`
+ mv ./README.md backup
```

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

Solution :

```
$ cat access.log | awk '{print $4":"$11}' | awk -F: '{print $5":"$6}' | sort | uniq -c |
sort -r | head -4
1475 "https://fundoopush-dev.bridgelabz.com/login"
1141 "https://fundoopush-dev.bridgelabz.com/dashboard/article"
377 "-":
176 "https://fundoopush-dev.bridgelabz.com/add-post”
```

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

Solution :

```
$ cat access.log | grep Oct | awk '{print $11}' | sort | uniq -c | sort -r | head -4
678 "https://fundoopush-dev.bridgelabz.com/login"
368 "https://fundoopush-dev.bridgelabz.com/dashboard/article"
177 "-"
18 "https://fundoopush-dev.bridgelabz.com/"
```

Ques. Print List of web response code count in the unique sorted order at specific hours.

Solution :

```
$ cat access.log |awk '{print $9}' | sort | uniq -c
3176 200
8 206
26 304
```

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

Solution :

```
$ cat access.log |awk '{print $NF}' | sort | uniq -c | sort -r | head -10
 3049 "114.79.180.62"
     33 "157.33.193.244"
     17 "66.249.79.48"
     15 "66.249.79.45"
     10 "66.249.79.42"
     10 "45.113.248.190"
      7 "45.93.20.4"
      7 "182.48.221.222"
      7 "14.142.39.102"
      6 "66.249.73.95"
```

Ques. Check if a folder exists or not. If it's not present, create it.

Solution :

```
$ nano file.sh
#!/bin/bash -x
if [ -f Hello ]; then
        echo "File Already Exists"
else
        echo "File Doesn't Exists"
        mkdir Hello
fi

$ ./file.sh
+ '[' -f Hello ']'
+ echo 'File Doesn'\''t Exists'
File Doesn't Exists
+ mkdir Hello
```

Ques. Execute command “hello” and “ls” and check its execution status and print whether command executed successful or not.

Solution :

```
$ hello
bash: hello: command not found

$ echo $?
127

$ ls
abc.log.1  appendCurrentDate.sh*  def.log.1  jkl.log.1

$ echo $?
0
```

Ques. Set environment usersecret=“dH34xJaa23” if it's already not set.

Solution :

```
$ echo $usersecret
$

$ export usersecret=“dH34xJaa23”

$ echo $usersecret
dH34xJaa23
```

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

Solution :

```
$ nano log.sh
#!/bin/bash -x
for files in `ls *.log`
do
      grep -r "systemd" $files | uniq -c
done
```

```
$ ./log.sh
++ ls a.log abc-31082020.log b.log c.log d.log def-31082020.log e.log jkl-31082020.log
+ for files in `ls *.log`
+ grep -r systemd a.log
+ uniq -c
    1 systemd
+ for files in `ls *.log`
+ grep -r systemd abc-31082020.log
+ uniq -c
+ for files in `ls *.log`
+ grep -r systemd b.log
+ uniq -c
    2 systemd
+ for files in `ls *.log`
+ grep -r systemd c.log
+ uniq -c
    1 systemd
+ for files in `ls *.log`
+ grep -r systemd d.log
+ uniq -c
    3 systemd
+ for files in `ls *.log`
+ grep -r systemd def-31082020.log
+ uniq -c
+ for files in `ls *.log`
+ grep -r systemd e.log
+ uniq -c
    2 systemd
+ for files in `ls *.log`
+ grep -r systemd jkl-31082020.log
+ uniq -c
```

Data Analysis / Manipulation (AWK)

i) Print EmployeeName and TotalPay who has BasePay greater than 10000

```
$ awk '$4 >10000' data.csv | awk '{print $2" : "$7}'
EmployeeName : TotalPay
NATHANIEL : 567595
GARY : 538909
ALBERT : 335279
CHRISTOPHER : 332343
```

```
PATRICK : 326373
DAVID : 316285
ALSON : 315981
DAVID : 307899
JOANNE : 302377
PATRICIA : 297608
EDWARD : 294580
```

ii) What is the aggregate TotalPay of employees whose jobtitle is ‘CAPTAIN’

```
$ cat data.csv | grep CAPTAIN | awk '{sum+=$7}END{print "TotalPay of CAPTAIN : "sum/NR}'
TotalPay of CAPTAIN : 390599
```

iii) Print JobTitle and OvertimePay who has OvertimePay is between 7000 and 10000

```
$ awk '$5 >7000 && $5 <10000' data.csv | awk '{print $3" : "$5}'
DEPUTYCHIEF : 9737
ASSTDEPUTY : 8601
```

iv) Print the average BasePay

```
$ awk '{sum+=$4}END{print "Average BasePay : "sum/(NR-1)}' data.csv
Average BasePay : 172333
```