2) Move files from one folder to their respective folders

Code:

#! /bin/bash -x

for file in `ls \*.txt`

do

foldername=`echo $file | awk -F. '{print $1}'`;

if [ -d $foldername ]

then

rm -r $foldername

fi

mkdir $foldername;

echo $foldername created;

cp $file $foldername;

echo $file copied to $foldername sucessfully;

done

Output:

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

$ nano moveFilestoFolder.sh

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

$ ./moveFilestoFolder.sh

++ ls abc.txt def.txt q1.txt q2.txt

+ for file in `ls \*.txt`

++ echo abc.txt

++ awk -F. '{print $1}'

+ foldername=abc

+ '[' -d abc ']'

+ rm -r abc

+ mkdir abc

+ echo abc created

abc created

+ cp abc.txt abc

+ echo abc.txt copied to abc sucessfully

abc.txt copied to abc sucessfully

+ for file in `ls \*.txt`

++ echo def.txt

++ awk -F. '{print $1}'

+ foldername=def

+ '[' -d def ']'

+ rm -r def

+ mkdir def

+ echo def created

def created

+ cp def.txt def

+ echo def.txt copied to def sucessfully

def.txt copied to def sucessfully

+ for file in `ls \*.txt`

++ echo q1.txt

++ awk -F. '{print $1}'

+ foldername=q1

+ '[' -d q1 ']'

+ rm -r q1

+ mkdir q1

+ echo q1 created

q1 created

+ cp q1.txt q1

+ echo q1.txt copied to q1 sucessfully

q1.txt copied to q1 sucessfully

+ for file in `ls \*.txt`

++ echo q2.txt

++ awk -F. '{print $1}'

+ foldername=q2

+ '[' -d q2 ']'

+ rm -r q2

+ mkdir q2

+ echo q2 created

q2 created

+ cp q2.txt q2

+ echo q2.txt copied to q2 sucessfully

q2.txt copied to q2 sucessfully

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

3) Append current date to all log files

Code:

#! /bin/bash

touch aaa.log.1 bbb.log.1 ccc.log.1 ddd.log.1

echo files created : \*.log.1

for file in `ls \*.log.1`;

do

file1=`echo $file | awk -F. '{print $1}'`;

echo $file1;

date\_append=`date +"%d-%m-%Y"`;

echo $date\_append;

append\_file="${file1}.${date\_append}.log";

echo $append\_file;

mv $file $append\_file;

echo date appended sucessfully;

done

Output:

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

$ nano append\_date.sh

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

$ ./append\_date.sh

files created : aaa.log.1 bbb.log.1 ccc.log.1 ddd.log.1

aaa

05-07-2020

aaa.05-07-2020.log

date appended sucessfully

bbb

05-07-2020

bbb.05-07-2020.log

date appended sucessfully

ccc

05-07-2020

ccc.05-07-2020.log

date appended sucessfully

ddd

05-07-2020

ddd.05-07-2020.log

date appended successfully

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

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

Code:

#! /bin/bash

read -p "Enter the name of the folder: " folder\_name

if [ -d $folder\_name ]

then

echo $folder\_name already exists

else

mkdir $folder\_name;

echo $folder\_name created sucessfully

fi

Output:

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

$ ./check\_folder.sh

Enter the name of the folder: abc

abc already exists

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

$ ./check\_folder.sh

Enter the name of the folder: qwerty

qwerty created sucessfully

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

$ ls

aaa.05-07-2020.log access.log ccc.05-07-2020.log ddd.05-07-2020.log linux\_chit\_sheet.pdf q1/ q2.txt

abc/ append\_date.sh\* check\_folder.sh\* def/ linux\_problem\_sheet.pdf q1.txt **qwerty/**

abc.txt bbb.05-07-2020.log data.csv def.txt moveFilestoFolder.sh\* q2/ README.md

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

6) Check execution status for hello and ls commands

Output:

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

$ hello

bash: hello: command not found

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

$ $?

bash: 127: command not found

**This means command not executed as we got error code 127 after $?**

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

$ ls

aaa.05-07-2020.log access.log ccc.05-07-2020.log ddd.05-07-2020.log linux\_chit\_sheet.pdf q1/ q2.txt

abc/ append\_date.sh\* check\_folder.sh\* def/ linux\_problem\_sheet.pdf q1.txt qwerty/

abc.txt bbb.05-07-2020.log data.csv def.txt moveFilestoFolder.sh\* q2/ README.md

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

$ $?

bash: 0: command not found

**This means command executed successfully as we got output and it returned 0 after $?**

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

7) Set environment usersecret=”dH34xJaa23” if its already not set

Output:

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

$ env | grep usersecret

**So environment usersecret was not set as it is not assigned any value**

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

$ export usersecret=dH34xJaa23

**Using this command we setup environment for usersecret with some value**

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

$ env | grep usersecret

usersecret=dH34xJaa23

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

8) Find a specific word and print no of occurrences of that word in access.log file

Output:

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

$ grep -c systemd access.log

0

**As the word is not present in access.log file it returned count as 0**

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

$ grep -c bridgelabz access.log

2833

**It returned the actual count of above word in the file access.log**

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

9) Create process list table displaying pid, ppid, command name, cpu%, mem%

Output:

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

$ ps aux

PID PPID PGID WINPID TTY UID STIME COMMAND

1776 1450 1776 9396 pty0 197608 18:02:27 /usr/bin/ps

1450 1449 1450 1156 pty0 197608 16:27:13 /usr/bin/bash

1449 1 1449 2828 ? 197608 16:27:13 /usr/bin/mintty

**The above command executed in git bash doesn’t display cpu and memory utilization but it can give required result in linux operating system with above command or top command also using command like : => ps –eo pid,ppid,cmd,%mem,%cpu**

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

10) Print last 4 frequently access urls count in sorted order from access.log

Output:

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

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

377 "-"

2 "http://fundoopush-dev.bridgelabz.com/.well-known/acme-challenge/4xM-Y1899BrBIJ76P5Er2sj2VhEtyi\_DFvfo6xiMXXw"

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

28 "https://fundoopush-dev.bridgelabz.com/"

176 "https://fundoopush-dev.bridgelabz.com/add-post"

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

1141 "https://fundoopush-dev.bridgelabz.com/dashboard/article"

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

5 "https://fundoopush-dev.bridgelabz.com/dashboard/jobs"

1475 "https://fundoopush-dev.bridgelabz.com/login"

**It sorts and counts occurrence of URL using sort and uniq command**

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

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

1475 "https://fundoopush-dev.bridgelabz.com/login"

1141 "https://fundoopush-dev.bridgelabz.com/dashboard/article"

377 "-"

176 "https://fundoopush-dev.bridgelabz.com/add-post"

28 "https://fundoopush-dev.bridgelabz.com/"

5 "https://fundoopush-dev.bridgelabz.com/dashboard/jobs"

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"

**It further arranges it in reverse order so as to get frequent URLs**

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

$ cat access.log | awk '{print $11}' | sort | uniq -c | sort -nr | head -n 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"

**$11 denotes the array index of url after sliced using awk command while uniq is to count the occurences of particular URL with simple sort which numerically sorts and sort –nr sorts it in reverse order to give frequent URL count and head –n 4 gives top 4 URL**

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

11) Print list of last 4 frequently accessed URLs at particular time from access.log

Output:

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

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

**Sort –k 4 sorts as per key and 4 denotes time stamp field**

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

12) Print list of web response codes in unique sorted order

Output:

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

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

3176 200

26 304

8 206

**It is not done for a particular timestamp because there are just three response codes and it was not possible to choose a particular time stamp where all three response codes occurred and hence extracting all response codes for particular time stamp made no sense(as only three response codes occurred with 200 occurring for almost more than 90% entries)**

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

13) Print list of last 10 unique sorted client IP from access.log

Output: MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

$ cat access.log | awk '{print $1}' | sort | uniq -c | sort -nr | head -n 10

383 10.56.2.2

345 10.56.19.3

323 10.56.21.2

285 10.56.6.4

283 10.56.1.3

261 10.56.0.3

251 10.56.22.3

242 10.56.4.2

214 10.56.3.4

213 10.56.10.2

**head –n 10 is used to print last 10 client IPs which are sorted reversely using sort –nr and uniq is used to count and sort IPs numerically**

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

14) Data analysis

i) Print EmployeeName and TotalPay who have Basepay greater than 100,000

Output:

MANISH@MANISH-VAIO MINGW64 ~/Desktop/Assignment/linux-content (master)

$ awk '{

if($4>100000)

{ print $2 " " $7;}

}' data.csv

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) aggregate TotalPay of employess with Job Title CAPTAIN

Output:

MANISH@MANISH-VAIO MINGW64 ~/Desktop/Assignment/linux-content (master)

$ cat data.csv | grep -i captain | awk ' {sum+=$7}END{print sum}'

1171796

iii) Print Job title and OvertimePay whose Overtime Pay in between 7000 and 10000

Output: MANISH@MANISH-VAIO MINGW64 ~/Desktop/Assignment/linux-content (master)

$ awk '{

if($5>7000 && $5<10000)

{ print $3 " " $5;}

}' data.csv

DEPUTYCHIEF 9737

ASSTDEPUTY 8601

iv) Print average basepay

Output:

MANISH@MANISH-VAIO MINGW64 ~/Desktop/Assignment/linux-content (master)

$ cat data.csv | awk ' {sum+=$4}END{print sum/(NR-1)}'

172333

**$4 is field for basepay and after calculate total sum of basepay it is divided by total no of entries which NR(no of rows) – 1 because it contains header as well**

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

15) Find difference between original file and updated file

Output:

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

$ mkdir original

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

$ mkdir updated

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

$ ls

aaa.05-07-2020.log access.log ccc.05-07-2020.log ddd.05-07-2020.log linux\_chit\_sheet.pdf original/ q2/ README.md

abc/ append\_date.sh\* check\_folder.sh\* def/ linux\_problem\_sheet.pdf q1/ q2.txt updated/

abc.txt bbb.05-07-2020.log data.csv def.txt moveFilestoFolder.sh\* q1.txt qwerty/

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

$ touch original-file.sh

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

$ cp original-file.sh original/

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

$ touch updated-file.sh

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

$ cp updated-file.sh updates/

cp: cannot create regular file 'updates/': Not a directory

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

$ cp updated-file.sh updated/

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

$ ls

aaa.05-07-2020.log append\_date.sh\* data.csv linux\_chit\_sheet.pdf original-file.sh q2.txt updated-file.sh

abc/ bbb.05-07-2020.log ddd.05-07-2020.log linux\_problem\_sheet.pdf q1/ qwerty/

abc.txt ccc.05-07-2020.log def/ moveFilestoFolder.sh\* q1.txt README.md

access.log check\_folder.sh\* def.txt original/ q2/ updated/

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

$ cd original

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content/original (master)

$ ls

original-file.sh

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content/original (master)

$ cd ..

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

$ cd updated

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content/updated (master)

$ ls

updated-file.sh

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content/updated (master)

$ cd ..

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

**$ diff original/ updated/**

**Only in original/: original-file.sh**

**Only in updated/: updated-file.sh**

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

$ diff -q original/ updated/

Only in original/: original-file.sh

Only in updated/: updated-file.sh

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

$ ls

aaa.05-07-2020.log append\_date.sh\* data.csv linux\_chit\_sheet.pdf original-file.sh q2.txt updated-file.sh

abc/ bbb.05-07-2020.log ddd.05-07-2020.log linux\_problem\_sheet.pdf q1/ qwerty/

abc.txt ccc.05-07-2020.log def/ moveFilestoFolder.sh\* q1.txt README.md

access.log check\_folder.sh\* def.txt original/ q2/ updated/

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

$ cp -r original/ original-backup/

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

$ ls

aaa.05-07-2020.log append\_date.sh\* data.csv linux\_chit\_sheet.pdf original-backup/ q2/ updated/

abc/ bbb.05-07-2020.log ddd.05-07-2020.log linux\_problem\_sheet.pdf original-file.sh q2.txt updated-file.sh

abc.txt ccc.05-07-2020.log def/ moveFilestoFolder.sh\* q1/ qwerty/

access.log check\_folder.sh\* def.txt original/ q1.txt README.md

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

$ cd original-backup/

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content/original-backup (master)

$ ls

original-file.sh

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content/original-backup (master)

$ cd ..

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

**$ diff -q original-backup/ updated/**

**Only in original-backup/: original-file.sh**

**Only in updated/: updated-file.sh**

MANISH@MANISH-VAIO MINGW64 ~/Desktop/TerminalCommands/Assignment/linux-content (master)

$

**After making use of diff command it showed files in each directory and it showed same result with backup folder and updated folder after copying contents from original directory to original backup folder.**

**It showed no difference between both diff commands performed.**