

Question 1 :

Description - To read n and generate a below pattern

```
1
1 2
1 2 3
1 2 3 4
```

```
saaurabh@LenovoLegion5: ~/ShellScripting/ShellQuestions
(saurabh@ LenovoLegion5)-[~/ShellScripting/ShellQuestions]
$ cat 01_pattern.sh
#!/bin/bash

echo "Enter the value of n : "
read n

for(( i=1; i<=$n; i++ ))
do
    k=1;
    for (( j=1; j<=i; j++ ))
    do
        echo -n "$k "
        (( k++ ))
    done
    echo ""
done

(saurabh@ LenovoLegion5)-[~/ShellScripting/ShellQuestions]
$ ./01_pattern.sh
Enter the value of n :
6
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
1 2 3 4 5 6

(saurabh@ LenovoLegion5)-[~/ShellScripting/ShellQuestions]
$
```

Question 2 :

Description-To read n and generate a below pattern

```
1
2 3
4 5 6
7 8 9 10
```

```
Select saaurabh@LenovoLegion5: ~/ShellScripting/ShellQuestions
(saurabh@ LenovoLegion5)-[~/ShellScripting/ShellQuestions]
$ vi 02_pattern.sh

(saurabh@ LenovoLegion5)-[~/ShellScripting/ShellQuestions]
$ chmod u+x 02_pattern.sh

(saurabh@ LenovoLegion5)-[~/ShellScripting/ShellQuestions]
$ cat 02_pattern.sh
#!/bin/bash

echo -n "Enter the value of n : "
read n
k=1;
for(( i=1; i<=n; i++ ))
do
    for(( j=1; j<=i; j++ ))
    do
        echo -n "$k "
        ((k++))
    done
    echo ""
done

(saurabh@ LenovoLegion5)-[~/ShellScripting/ShellQuestions]
$ ./02_pattern.sh
```

```
done
echo ""

(saurabh@ LenovoLegion5) ~/ShellScripting/ShellQuestions
$ ./02_pattern.sh
Enter the value of n : 4
1
2 3
4 5 6
7 8 9 10

(saurabh@ LenovoLegion5) ~/ShellScripting/ShellQuestions
$
```

Question3 :

Description-Script for addition of two real numbers

Input- bash 03_real_add.sh

Enter the numbers to addition: 4.28 1.21

Output- The sum of 4.28 and 1.21 is 5.49

```
Select saurabh@LenovoLegion5: ~/ShellScripting/ShellQuestions

(saurabh@ LenovoLegion5) ~/ShellScripting/ShellQuestions
$ vi 03_real_add.sh

(saurabh@ LenovoLegion5) ~/ShellScripting/ShellQuestions
$ chmod u+x 03_real_add.sh

(saurabh@ LenovoLegion5) ~/ShellScripting/ShellQuestions
$ cat 03_real_add.sh
#!/bin/bash

read -p "Enter the numbers to addition : " num1 num2

sum=$(echo "$num1 + $num2" | bc )

echo "The addition number $1 and $2 is $sum"

(saurabh@ LenovoLegion5) ~/ShellScripting/ShellQuestions
$ ./03_real_add.sh
Enter the numbers to addition : 1.22 1.23
The addition number and is 2.45

(saurabh@ LenovoLegion5) ~/ShellScripting/ShellQuestions
$
```

Question4:

Description- Script for arithmetic calculator using command line arguments

Input- bash 04_calculator.sh 1.2 + 2.6

Output- The sum of 1.2 and 2.6 is 3.8

```
Select saurabh@LenovoLegion5: ~/ShellScripting/ShellQuestions

(saurabh@ LenovoLegion5)~/ShellScripting/ShellQuestions
$ vi 04_calculator.sh

(saurabh@ LenovoLegion5)~/ShellScripting/ShellQuestions
$ chmod u+x 04_calculator.sh

(saurabh@ LenovoLegion5)~/ShellScripting/ShellQuestions
$ cat 04_calculator.sh
#!/bin/bash

num1=$1
num2=$2

sum=$(echo "$num1 + $num2" | bc)

echo "The addition of number $1 and $2 is $sum"

(saurabh@ LenovoLegion5)~/ShellScripting/ShellQuestions
$ ./04_calculator.sh 1.22 1.24
The addition of number 1.22 and 1.24 is 2.46

(saurabh@ LenovoLegion5)~/ShellScripting/ShellQuestions
$
```

Question5:

Description- Script to compare larger integer values from a 'n' number of arguments using command line arguments

Input- bash 05_largest.sh 1 3 8 6 5 7 9 2

Output- The largest value is 9

```
Select saurabh@LenovoLegion5: ~/ShellScripting/ShellQuestions

(saurabh@ LenovoLegion5)~/ShellScripting/ShellQuestions
$ vi 05_largest.sh

(saurabh@ LenovoLegion5)~/ShellScripting/ShellQuestions
$ chmod u+x 05_largest.sh

(saurabh@ LenovoLegion5)~/ShellScripting/ShellQuestions
$ cat 05_largest.sh
#!/bin/bash

#Setting first num as max
max=$1

#By using $@ we can iterate through all the elements added from command line
for i in "$@"
do
    #Compared each num with max
    if [ $1 -gt $max ]
    then
        max=$i
    fi
done

echo "The largest number from given input is $max"

(saurabh@ LenovoLegion5)~/ShellScripting/ShellQuestions
$ ./05_largest.sh 1 3 8 7 6 5 9 2
The largest number from given input is 9

(saurabh@ LenovoLegion5)~/ShellScripting/ShellQuestions
$
```

Question6:

Description- Script to print a given number in reverse order.

Input- bash 06_reverse.sh 639872

Output- The reversed number of entered number is 278936

```
Select saurabh@LenovoLegion5: ~/ShellScripting/ShellQuestions

(saurabh@ LenovoLegion5)~[~/ShellScripting/ShellQuestions]
$ vi 06_reverse.sh

(saurabh@ LenovoLegion5)~[~/ShellScripting/ShellQuestions]
$ chmod u+x 06_reverse.sh

(saurabh@ LenovoLegion5)~[~/ShellScripting/ShellQuestions]
$ ./06_reverse.sh 639872
The reverse of given number is 278936

(saurabh@ LenovoLegion5)~[~/ShellScripting/ShellQuestions]
$ cat 06_reverse.sh
#!/bin/bash

number=$1

reversenumber=$(echo " $1 " | rev )

echo "The reverse of given number is $reversenumber"

(saurabh@ LenovoLegion5)~[~/ShellScripting/ShellQuestions]
$ ./06_reverse.sh 639872
The reverse of given number is 278936

(saurabh@ LenovoLegion5)~[~/ShellScripting/ShellQuestions]
$
```

Question7:

Description- Script to delete empty lines from a file

Input- bash 07_delete_empty_lines.sh file.txt

Output- All empty lines of the file file.txt will be deleted

Before script running, content of the file file.txt :

Hello, I am Siddaling.

I am from Belgaum.

I studied B-Tech in ECE.

After script running, content of the file file.txt :

Hello, I am Siddaling.

I am from Belgaum.

I studied B-Tech in ECE.

```
Select saurabh@LenovoLegion5: ~/ShellScripting/ShellQuestions

(saurabh@ LenovoLegion5)~[~/ShellScripting/ShellQuestions]
$ vi file.txt

(saurabh@ LenovoLegion5)~[~/ShellScripting/ShellQuestions]
$ vi 07_delete_empty_lines.sh

(saurabh@ LenovoLegion5)~[~/ShellScripting/ShellQuestions]
$ chmod u+x 07_delete_empty_lines.sh

(saurabh@ LenovoLegion5)~[~/ShellScripting/ShellQuestions]
$ cat 07_delete_empty_lines.sh
#!/bin/bash

filename=$1

if [ $# -eq 0 ]
then
    echo "Please provide the file name in command line."
else
    sed '/^$/d' $filename
fi

(saurabh@ LenovoLegion5)~[~/ShellScripting/ShellQuestions]
$ ./07_delete_empty_lines.sh file.txt
Hello, I am Saurabh.
I am from Maharashtra.
Recently I completed my CDAC course from MET Nashik.

(saurabh@ LenovoLegion5)~[~/ShellScripting/ShellQuestions]
$ cat file.txt
Hello, I am Saurabh.

I am from Maharashtra.

Recently I completed my CDAC course from MET Nashik.

(saurabh@ LenovoLegion5)~[~/ShellScripting/ShellQuestions]
$
```

Question9:

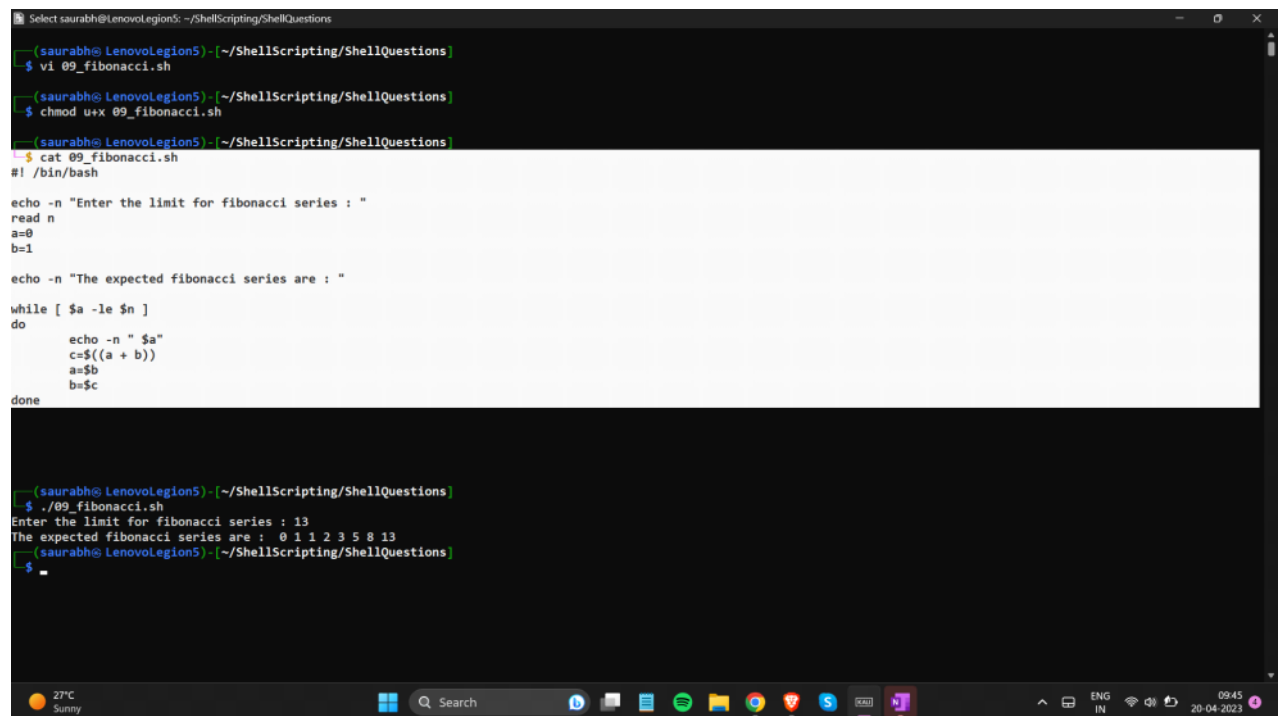
Description- script to read 'n' and generate Fibonacci numbers $\leq n$

Input- `bash 09_fibonacci.sh`

Enter limit for fibonacci series: 13

Output- The expected fibonacci series is:

0, 1, 1, 2, 3, 5, 8, 13,



```
Select saurabh@LenovoLegion5: ~/ShellScripting/ShellQuestions
(saurabh@ LenovoLegion5)~/ShellScripting/ShellQuestions
$ vi 09_fibonacci.sh
(saurabh@ LenovoLegion5)~/ShellScripting/ShellQuestions
$ chmod u+x 09_fibonacci.sh
(saurabh@ LenovoLegion5)~/ShellScripting/ShellQuestions
$ cat 09_fibonacci.sh
#!/bin/bash

echo -n "Enter the limit for fibonacci series : "
read n
a=0
b=1

echo -n "The expected fibonacci series are : "

while [ $a -le $n ]
do
    echo -n " $a"
    c=$((a + b))
    a=$b
    b=$c
done

(saurabh@ LenovoLegion5)~/ShellScripting/ShellQuestions
$ ./09_fibonacci.sh
Enter the limit for fibonacci series : 13
The expected fibonacci series are : 0 1 1 2 3 5 8 13
(saurabh@ LenovoLegion5)~/ShellScripting/ShellQuestions
$
```

Question11:

script to print chess board , black as 1 , white as 0

Input- `bash 11_chess_board.sh`

```
Select saurabh@LenovoLegion5: ~/ShellScripting/ShellQuestions
(saurabh@ LenovoLegion5) ~/ShellScripting/ShellQuestions
$ vi 11_chess_board.sh
(saurabh@ LenovoLegion5) ~/ShellScripting/ShellQuestions
$ cat 11_chess_board.sh
#!/bin/bash

for (( i=1; i<=8; i++ ))
do
    for (( j=1; j<=8; j++ ))
    do
        if (( (i + j) % 2 == 0 ));
        then
            echo -n "0 "
        else
            echo -n "1 "
        fi
    done
    echo ""
done

(saurabh@ LenovoLegion5) ~/ShellScripting/ShellQuestions
$ ./11_chess_board.sh
0 1 0 1 0 1 0 1
1 0 1 0 1 0 1 0
0 1 0 1 0 1 0 1
1 0 1 0 1 0 1 0
0 1 0 1 0 1 0 1
1 0 1 0 1 0 1 0
0 1 0 1 0 1 0 1
1 0 1 0 1 0 1 0

(saurabh@ LenovoLegion5) ~/ShellScripting/ShellQuestions
$
```

Question13:

Description- Script to print the following:

- Currently logged users
- Your shell directory
- Home directory
- OS name & version
- Current working directory
- Number of users logged in
- Show all available shells in your system
- Hard disk information
- CPU information.
- Memory information.
- File system information.
- Currently running process

```
Select saurabh@LenovoLegion5: ~/ShellScripting/ShellQuestions
echo "2. Your shell directory : "
echo "3. Home directory : "
echo "4. Os name & version : "
echo "5. Current working directory : "
echo "6. Number of users logged in : "
echo "7. Show all available shell in your sysstem : "
echo "8. Hard disk information : "
echo "9. CPU information : "
echo "10. Memory information : "
echo "11. File system information : "
echo "12. Currently running process : "
read n

case $n in
  1)
    sudo w;;
  2)
    which bash ;;
  3)
    cd ~ ;;
  4)
    uname -sr ;;
  5)
    pwd ;;
  6)
    who -q ;;
  7)
    cat /etc/shells ;;
  8)
    df -h ;;
  9)
    cat /proc/cpuinfo | head -n 10 ;;
  10)
    cat /proc/meminfo ;;
  11)
    df -h ;;
  12)
    ps ;;
  *)
    echo "Please enter a valid input."
esac

(saurabh@ LenovoLegion5) [~/ShellScripting/ShellQuestions]
```

```
Select saurabh@LenovoLegion5: ~/ShellScripting/ShellQuestions
cat /etc/shells ;;
df -h ;;
cat /proc/cpuinfo | head -n 10 ;;
cat /proc/meminfo ;;
df -h ;;
ps ;;
echo "Please enter a valid input."
esac

(saurabh@ LenovoLegion5) [~/ShellScripting/ShellQuestions]
$ ./15_system_info.sh
Enter the choice :
1. Currently logged users :
2. Your shell directory :
3. Home directory :
4. Os name & version :
5. Current working directory :
6. Number of users logged in :
7. Show all available shell in your sysstem :
8. Hard disk information :
9. CPU information :
10. Memory information :
11. File system information :
12. Currently running process :
7
# /etc/shells: valid login shells
/bin/sh
/bin/bash
/usr/bin/bash
/bin/rbash
/usr/bin/rbash
/bin/dash
/usr/bin/dash
/usr/bin/sh

(saurabh@ LenovoLegion5) [~/ShellScripting/ShellQuestions]
$
```

Question15

Description- Script to rename current working directory with given name.

Input- bash 15_rename_cur_dir.sh siddaling

Output- Before running the script:

Name of current directory- siddaling1

After running the script:

Name of current directory- siddaling

```
Select saurabh@LenovoLegion5: ~/ShellScripting/ShellQues
(saurabh@ LenovoLegion5) [~/ShellScripting/ShellQues]
$ vi 15_rename_cur_dir.sh
```

```
Select saurabh@LenovoLegion5: ~/ShellScripting/ShellQues

(saurabh@LenovoLegion5)~/ShellScripting/ShellQues
$ vi 15_rename_cur_dir.sh

(saurabh@LenovoLegion5)~/ShellScripting/ShellQues
$ chmod u+x 15_rename_cur_dir.sh

(saurabh@LenovoLegion5)~/ShellScripting/ShellQues
$ cat 15_rename_cur_dir.sh
#!/bin/bash

current_dir=$(pwd)
new_dir_name=$1

echo "Before running the script : $current_dir"

#getting the parent directory
parent_dir=$(dirname "$current_dir")

#Changing the directory name
mv "$current_dir" "$parent_dir/$new_dir_name"

echo "After running the script : $parent_dir/$new_dir_name "
```

```
(saurabh@LenovoLegion5)~/ShellScripting/ShellQues
$ ./15_rename_cur_dir.sh ShellQuestions
Before running the script : /home/saurabh/ShellScripting/ShellQues
After running the script : /home/saurabh/ShellScripting/ShellQuestions

(saurabh@LenovoLegion5)~/ShellScripting/ShellQues
$
```

Question16:

Description- Script to rename all .jpg files by replacing prefix which is given by user

Input- bash 16_rename_album.sh myday

Output- Before running the script

```
ls
```

```
16_rename_album.sh DSN001.jpg DSN002.jpg DSN003.jpg DSN004.jpg DSN005.jpg
DSN006.jpg DSN007.jpg
```

After running the script

```
$ ls
```

```
16_rename_album.sh myday_001.jpg myday_002.jpg myday_003.jpg myday_004.jpg
myday_005.jpg myday_006.jpg myday_007.jpg
```

Question17:

Description- Script to print contents of file from given line number to next given number of lines.

Input- bash 17_print_lines.sh 5 4

Output-

```
line 5
line 6
line 7
line 8
```



```
Select saurabh@LenovoLegion5: ~/ShellScripting/ShellQuestions

(saurabh@LenovoLegion5)~/ShellScripting/ShellQuestions
$ vi 17_print_lines.sh

(saurabh@LenovoLegion5)~/ShellScripting/ShellQuestions
$ chmod u+x 17_print_lines.sh

(saurabh@LenovoLegion5)~/ShellScripting/ShellQuestions
$ cat 17_print_lines.sh
#!/bin/bash

if [ $# -eq 2 ]
then
    start_line=$1
    number_of_lines=$(( $2 - 1 ))

    sed -n $start_line,+$number_of_lines p demofile.txt
else
    echo "Please provide the command line arguments."
fi

(saurabh@LenovoLegion5)~/ShellScripting/ShellQuestions
$ ./17_print_lines.sh 5 4
Line 5
Line 6
Line 7
Line 8

(saurabh@LenovoLegion5)~/ShellScripting/ShellQuestions
$
```

Question18:

Description- Script to display the longest and shortest user-names on the system.

Input- bash 18_largest_username.sh

Output- The user with longest name is gnome-initial-setup

The user with shortest name is lp

```
Select saurabh@LenovoLegion5: ~/ShellScripting/ShellQuestions

(saurabh@LenovoLegion5)~/ShellScripting/ShellQuestions
$ vi 18_largest_username.sh

(saurabh@LenovoLegion5)~/ShellScripting/ShellQuestions
$ chmod u+x 18_largest_username.sh

(saurabh@LenovoLegion5)~/ShellScripting/ShellQuestions
$ cat 18_largest_username.sh
#!/bin/bash

usernames=$(cat /etc/passwd | awk -F: '{print $1}')

#variables to store longest and shortest usernames.
shortest=$(cat /etc/passwd | awk -F: '{print $1}' | head -n 1)
longest=""

for user in $usernames
do
    if [ ${#user} -gt ${#longest} ];
    then
        longest=$user
    fi
    if [ ${#user} -lt ${#shortest} ];
    then
        shortest=$user
    fi
done

echo "Longest Username is : $longest"
echo "Shortest Username is : $shortest"

(saurabh@LenovoLegion5)~/ShellScripting/ShellQuestions
$ ./18_largest_username.sh
Longest Username is : speech-dispatcher
Shortest Username is : lp

(saurabh@LenovoLegion5)~/ShellScripting/ShellQuestions
$
```

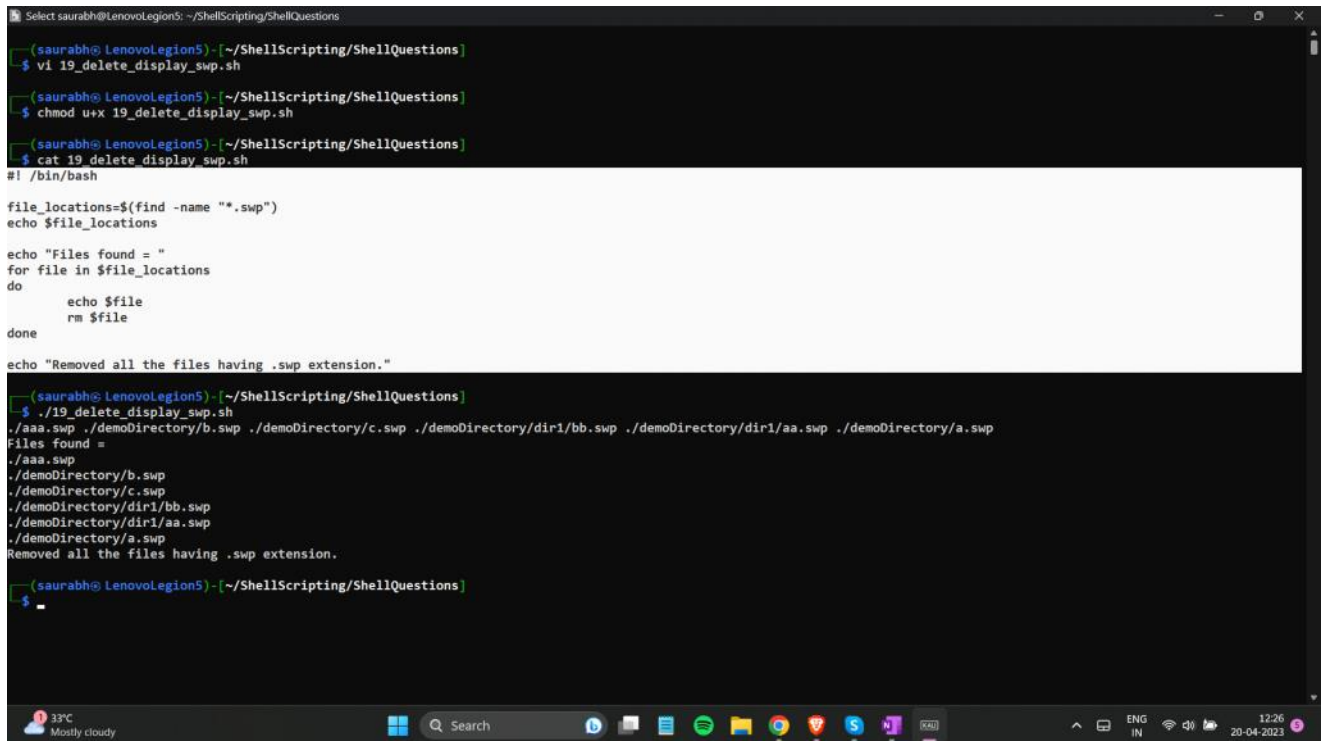
Question19:

Description- Script to delete all the .swp files found in your system or directory.

Input- bash 19_delete_display_swp.sh

Output- swp files found:

```
./b.swp
./siddaling/b.swp
./siddaling/c.swp
./siddaling/test/b.swp
./siddaling/test/c.swp
./siddaling/test/d.swp
./siddaling/test/a.swp
./siddaling/test/e.swp
./siddaling/d.swp
./siddaling/a.swp
./a.swp
```



```
Select saurabh@LenovoLegion5: ~/ShellScripting/ShellQuestions
(saurabh@LenovoLegion5)~/ShellScripting/ShellQuestions
$ vi 19_delete_display_swp.sh
(saurabh@LenovoLegion5)~/ShellScripting/ShellQuestions
$ chmod u+x 19_delete_display_swp.sh
(saurabh@LenovoLegion5)~/ShellScripting/ShellQuestions
$ cat 19_delete_display_swp.sh
#!/bin/bash

file_locations=$(find -name "*.swp")
echo $file_locations

echo "Files found = "
for file in $file_locations
do
    echo $file
    rm $file
done

echo "Removed all the files having .swp extension."

(saurabh@LenovoLegion5)~/ShellScripting/ShellQuestions
$ ./19_delete_display_swp.sh
./aaa.swp ./demoDirectory/b.swp ./demoDirectory/c.swp ./demoDirectory/dir1/bb.swp ./demoDirectory/dir1/aa.swp ./demoDirectory/a.swp
Files found =
./aaa.swp
./demoDirectory/b.swp
./demoDirectory/c.swp
./demoDirectory/c.swp
./demoDirectory/dir1/bb.swp
./demoDirectory/dir1/aa.swp
./demoDirectory/a.swp
Removed all the files having .swp extension.

(saurabh@LenovoLegion5)~/ShellScripting/ShellQuestions
$
```

Question20:

Description- script for generating random 8-character passwords including alpha numeric characters.

Input- bash 20_random_password.sh

Output- The generated 8 random passwords are:

```
J,kwAc{0
Pft4Get*
\2('pKlr
!,(VwY%
^X|Z+}u8
y]\(2P|R
B#6'V=#D
8\Lrz{bc
```

Question28:

Description- Script to locks file permissions for a particular directory for groups and others

Input- bash 28_lock_permissions.sh test

```
saaurabh@LenovoLegion5: ~/ShellScripting/ShellQuestions
(saurabh@ LenovoLegion5)~/ShellScripting/ShellQuestions
$ vi 28_lock_permission.sh

(saurabh@ LenovoLegion5)~/ShellScripting/ShellQuestions
$ chmod u+x 28_lock_permission.sh

(saurabh@ LenovoLegion5)~/ShellScripting/ShellQuestions
$ cat 28_lock_permission.sh
#!/bin/bash

echo -e "Before locking the permissions for group and other : \n"
ls -l

filenames=$( ls -l | awk '{print $NF}')

echo -e "After locking the permissions for group and others : \n"

for file in $filenames
do
    chmod 700 $file
    ls -l $file
done

(saurabh@ LenovoLegion5)~/ShellScripting/ShellQuestions
$
```

```
saaurabh@LenovoLegion5: ~/ShellScripting/ShellQuestions
-rwxr-xr-x 1 saurabh saurabh 115 Apr 19 13:40 06_reverse.sh
-rwxr-xr-x 1 saurabh saurabh 134 Apr 19 15:52 07_delete_empty_lines.sh
-rwxr-xr-x 1 saurabh saurabh 97 Apr 19 18:59 08_operator_dependent.sh
-rwxr-xr-x 1 saurabh saurabh 205 Apr 20 09:44 09_fibonacci.sh
-rwxr-xr-x 1 saurabh saurabh 15 Apr 20 09:48 10_string_length.sh
-rwxr-xr-x 1 saurabh saurabh 173 Apr 19 19:21 11_chess_board.sh
-rwxr-xr-x 1 saurabh saurabh 83 Apr 20 10:01 12_sorting.sh
-rwxr-xr-x 1 saurabh saurabh 785 Apr 20 10:05 13_system_info.sh
-rwxr-xr-x 1 saurabh saurabh 310 Apr 20 11:41 15_rename_cur_dir.sh
-rwxr-xr-x 1 saurabh saurabh 194 Apr 20 11:26 17_print_lines.sh
-rwxr-xr-x 1 saurabh saurabh 427 Apr 20 12:18 18_largest_username.sh
-rwxr-xr-x 1 saurabh saurabh 207 Apr 20 12:24 19_delete_display_swp.sh
-rwxr-xr-x 1 saurabh saurabh 15 Apr 20 12:40 22_upper_lower.sh
-rwxr-xr-x 1 saurabh saurabh 17 Apr 20 12:46 25_recursion.sh
-rwxr-xr-x 1 saurabh saurabh 257 Apr 20 12:59 28_lock_permission.sh
-rwxr-xr-x 3 saurabh saurabh 4096 Apr 20 12:25 demoDirectory
-rwxr-xr-x 1 saurabh saurabh 71 Apr 20 11:26 demoFile.txt
-rwxr-xr-x 1 saurabh saurabh 100 Apr 19 15:52 file.txt
After locking the permissions for group and others :

chmod: cannot access '92': No such file or directory
ls: cannot access '92': No such file or directory
-rwxr-xr-x 1 saurabh saurabh 159 Apr 19 12:47 01_pattern.sh
-rwxr-xr-x 1 saurabh saurabh 162 Apr 19 12:55 02_pattern.sh
-rwxr-xr-x 1 saurabh saurabh 148 Apr 19 13:17 03_real_add.sh
-rwxr-xr-x 1 saurabh saurabh 113 Apr 19 13:22 04_calculator.sh
-rwxr-xr-x 1 saurabh saurabh 267 Apr 19 13:33 05_largest.sh
-rwxr-xr-x 1 saurabh saurabh 115 Apr 19 13:40 06_reverse.sh
-rwxr-xr-x 1 saurabh saurabh 134 Apr 19 15:52 07_delete_empty_lines.sh
-rwxr-xr-x 1 saurabh saurabh 97 Apr 19 18:59 08_operator_dependent.sh
-rwxr-xr-x 1 saurabh saurabh 205 Apr 20 09:44 09_fibonacci.sh
-rwxr-xr-x 1 saurabh saurabh 15 Apr 20 09:48 10_string_length.sh
-rwxr-xr-x 1 saurabh saurabh 173 Apr 19 19:21 11_chess_board.sh
-rwxr-xr-x 1 saurabh saurabh 83 Apr 20 10:01 12_sorting.sh
-rwxr-xr-x 1 saurabh saurabh 785 Apr 20 10:05 13_system_info.sh
-rwxr-xr-x 1 saurabh saurabh 310 Apr 20 11:41 15_rename_cur_dir.sh
-rwxr-xr-x 1 saurabh saurabh 194 Apr 20 11:26 17_print_lines.sh
-rwxr-xr-x 1 saurabh saurabh 427 Apr 20 12:18 18_largest_username.sh
-rwxr-xr-x 1 saurabh saurabh 207 Apr 20 12:24 19_delete_display_swp.sh
-rwxr-xr-x 1 saurabh saurabh 15 Apr 20 12:40 22_upper_lower.sh
-rwxr-xr-x 1 saurabh saurabh 17 Apr 20 12:46 25_recursion.sh
-rwxr-xr-x 1 saurabh saurabh 257 Apr 20 12:59 28_lock_permission.sh
total 4
drwxr-xr-x 2 saurabh saurabh 4096 Apr 20 12:25 dir1
-rwxr-xr-x 1 saurabh saurabh 71 Apr 20 11:26 demoFile.txt
-rwxr-xr-x 1 saurabh saurabh 100 Apr 19 15:52 file.txt

(saurabh@ LenovoLegion5)~/ShellScripting/ShellQuestions
$
```

Question29:

Description- Script to display the names of any file-system which have less than 10% free space available

Input- bash 29_free_space.sh

```
Select saurabh@LenovoLegion5: ~/ShellScripting/ShellQuestions
(saurabh@LenovoLegion5)~/ShellScripting/ShellQuestions
$ vi 29_free_space.sh
(saurabh@LenovoLegion5)~/ShellScripting/ShellQuestions
$ chmod u+x 29_free_space.sh
(saurabh@LenovoLegion5)~/ShellScripting/ShellQuestions
$ cat 29_free_space.sh
#!/bin/bash

echo "These are the files whose usage is more than 60 % : "

df -h | awk '{if($5>60) print $0}'

(saurabh@LenovoLegion5)~/ShellScripting/ShellQuestions
$ ./29_free_space.sh
These are the files whose usage is more than 60 % :
Filesystem      Size  Used Avail Use% Mounted on
tools           238G  156G   82G   66% /init
drivers          238G  156G   82G   66% /usr/lib/usb/drivers
lib             238G  156G   82G   66% /usr/lib/usb/lib
drvfs           238G  156G   82G   66% /mnt/c

(saurabh@LenovoLegion5)~/ShellScripting/ShellQuestions
$
```

Question30:

Description- Script to count the number of users with user IDs between 500 and 10000 on the system.

```
Select saurabh@LenovoLegion5: ~/ShellScripting/ShellQuestions
(saurabh@LenovoLegion5)~/ShellScripting/ShellQuestions
$ vi 30_print_userId.sh
(saurabh@LenovoLegion5)~/ShellScripting/ShellQuestions
$ chmod u+x 30_print_userId.sh
(saurabh@LenovoLegion5)~/ShellScripting/ShellQuestions
$ ./30_print_userId.sh
Please provide start and end limit of user Id's.
(saurabh@LenovoLegion5)~/ShellScripting/ShellQuestions
$ ./30_print_userId.sh 500 10000
Total no of userId between 500 and 10000 :
4
(saurabh@LenovoLegion5)~/ShellScripting/ShellQuestions
$ cat 30_print_userId.sh
#!/bin/bash

if [ $# -eq 2 ]
then
    start_limit=$1
    end_limit=$2

    echo "Total no of userId between $start_limit and $end_limit : "
    cat /etc/passwd | awk -F : '{if($3 > 500 && $3 < 10000) print $0}' | wc -l
else
    echo "Please provide start and end limit of user Id's."
fi

(saurabh@LenovoLegion5)~/ShellScripting/ShellQuestions
$
```

Question31:

Description- Script to search a user present in the system.

Input- bash 32_user_present.sh root

```
Select saurabh@LenovoLegion5: ~/ShellScripting/ShellQuestions
(saurabh@LenovoLegion5)~/ShellScripting/ShellQuestions
$ vi 32_user_present.sh
(saurabh@LenovoLegion5)~/ShellScripting/ShellQuestions
$ chmod u+x 32_user_present.sh
(saurabh@LenovoLegion5)~/ShellScripting/ShellQuestions
$ ./32_user_present.sh saurabh
saurabh
User saurabh is present.
(saurabh@LenovoLegion5)~/ShellScripting/ShellQuestions
$ car 32_user_present.sh
-bash: car: command not found
(saurabh@LenovoLegion5)~/ShellScripting/ShellQuestions
$ cat 32_user_present.sh
#!/bin/bash

if [ $# -eq 1 ]
then
    username=$1
    present=$(cat /etc/passwd | awk -F: '{print $1}' | grep $username )
    echo $present
    if [ "$present" = "$username" ];
    then
        echo "User $username is present."
    else
        echo "User $username is not present."
    fi
else
    echo "Please provide the username in command line."
fi

(saurabh@LenovoLegion5)~/ShellScripting/ShellQuestions
$ ./32_user_present.sh root
root
User root is present.
(saurabh@LenovoLegion5)~/ShellScripting/ShellQuestions
$
```