Assignment Day 23 | 21st November 2020
By Sameer Kulkarni

QUESTION 1 : Create a script script1 which will read the name of the user and the number of lines to be displayed from C:\abcd.txt from the users interactively. You welcome the user and then display the top so many lines from C:\abcd.txt. If the number of lines entered is greater than the number of lines available, a proper message should be displayed. (Hint: Use Array)

Answer 1: The script is

```
$val1 = Read-Host 'Enter user'
$val2 = Read-Host 'Number of lines to be displayed from abcd.txt '
              write-output "Welcome $val1"
$val3 = Get-Content ".\abcd.txt" | select -First $val2
$val4 = (Get-Content ".\abcd.txt").Length
if ($val4 -gt $val2)
{Write-output $val3}
else
{write-output "Does not contain enough lines to display"}
```

QUESTION 2 : Read 2 numbers from the user. If these numbers are greater than zero then display the sum of these numbers.

Answer 2. The script is as follows:

```
$val1 = Read-Host 'Enter value 1'
$val2 = Read-Host 'Enter value 2'
if ($val1 -gt 0 -and $val2 -gt 0)
{$result = $val1 + $val2
write-output "The sum of numbers entered is $result"}
else{ Write-Output "You entered a number less than zero"}
```

QUESTION 3 : Read the name & password of the user. Check whether the username is DAD and password is 111 and give appropriate messages

Answer 3. The script is as follows:

```
$c = Get-Credential
if (($c.username -eq "DAD") -and ($c.GetNetworkCredential().password) -eq "111")
{write-output "You entered correct username and password"}
else
{write-output "Wrong Credentials"}
```

QUESTION 4 : Read a file name from the user. Check recursively whether it's a regular file and is empty and then remove the file. After verifying the removal, give a confirmation message.

Answer 4: `find dir -empty -type f -delete`

QUESTION 5 : Create a shell script which gives the following choices to the user

1. Gives number of running services (service name is not required, only count the numbers)
2. List the no of processes
3. Count of files in the current dir

Answer 5:

1. `systemctl --type=service`
2. `ps -e`
3. `ls -1 | wc -l`

QUESTION 6 : Create a shell script that backs up all the content of a specific directory, where the designation directory name is equal to the current date and time, when the script was run.

Answer 6: The script is as follows:

Not understood.

QUESTION 7 : You have a directory that contains files of different extensions. (say mp3, exe, pdf, doc etc.). You need to take all files with the same extension and copy it in a directory where the directory name is the extension of the file.

Answer 7: The script used for this purpose is as follows:

```
#!/bin/bash
for filename in *; do
  if [[ -f "$filename" ]]; then
      base=${filename%.*}
      ext=${filename#$base.}
    mkdir -p "${ext}"
    mv "$filename" "${ext}"
  fi
done
```

QUESTION 8 : You have a few files with the same content but different names. I mean they are duplicates. You have to search all duplicate files and delete all files keeping only one copy of the file.

Answer 8: To search for duplicate files, fdupes is used with -d for deleting duplicate files and -N for not prompting before deleting. It keeps only one file in a specific directory. The command is:

```
fdupes -d -N /root
```

QUESTION 9 : Find the system uptime and display it in format in red background and yellow text.

Answer 9: The script to display uptime as required is as follows

```
tput setaf 3
tput setab 1
uptime
```

```
[sameer_kuls@lucentossam ~]$ uptime
 10:24:23 up 12 days, 17:33,  1 user,  load average: 0.02, 0.06, 0.02
[sameer_kuls@lucentossam ~]$
```

Note: The system uptime is 5days 3hours 7 min