



Experiment 5: (Shell scripts)

Name: Satvam Thakkar

SAPID: 60009210196

Aim: To study and implement basic Shell scripting.

Theory:

Shell Script

Shell is a program which interprets user commands through CLI like Terminal. The Bourne shell, bash shell and the C shell are the most used shells in Unix. Unix commands may also be executed non-interactively in the form of a Shell program or a Shell Script. The script is a series of commands that will be run together.

It can combine lengthy and repetitive sequences of commands into a single and simple script, which can be stored and executed anytime. This reduces the effort required by the end user. Typical operations performed by shell scripts include file manipulation, program execution, and printing text.

Creating and executing a shell script

Steps in creating a Shell Script:

1. **Create a file using a vi editor** (or any other editor).
2. Name the script file with **extension .sh**
3. **Start** the script with **#!/bin/sh**
4. Write some code.
5. Save the script file as filename.sh
6. Give the shell permission to execute it.
7. For **executing** the script type **bash filename.sh**

An example shell script

The following example shows a simple shell script that lists the contents of the current directory and also shows the path of the current directory.

```
#!/bin/sh  
ls  
pwd
```



Lab Assignments to complete in this session

Implementation Instruction:

Objectives:

- To understand shell variables and shell programming.
- To develop shell scripts.

Outcomes: After study of this experiment, the student will be able to

- Develop shell scripts for simple tasks.

Prerequisite: UNIX shell.

Requirements: Personal Computer, Ubuntu OS, Text Editor, LibreOffice.

1. Write Shell script to copy files from one folder to another
2. Write Shell script Count number of words, characters and lines.
3. Write Shell script To describe files in different format.
4. Write Shell script to find factorial of given number using bash script
5. Display first 10 natural numbers using bash script
6. Display Fibonacci series using bash script
7. Find given number is prime or nor using bash script
8. Write shell script to find biggest of three numbers
9. Write shell script to reverse a given number
10. Write shell script to find Sum of individual digits (1234 => 1+2+3+4=10)
11. Write a shell script to display a list of users currently logged in.
12. Write a shell script to perform arithmetic operations.
13. Write a shell script to copy contents of one file to another.
14. Write a shell program to generate multiplication table of a number upto a givenrange.
15. Write a shell program to count the number of files in a directory.
16. to find the number of matched characters, words and lines ina file.
17. Write a script to find the number of characters, words and lines in a file.
18. Write a script to display list of files starting with particular letter in thedirectory.
19. Write a script to develop a Fibonacci series.
20. Write a shell script to replace the Nth occurrence of a pattern.
21. Write a shell script to convert temperature from Centigrade to Fahrenheit.
22. Write a shell script to compute the power of a given number.
23. Write a shell script to check whether the entered number is prime or not.



A.Y.: 2022-23

Class: S.Y.B.Tech

Sub: System Fundamentals

24. Write a shell script to check whether the year is leap year or not.
25. Write a shell script to check whether a number is even or odd.
26. When to use shell scripts?
27. Where is the bash program located on your system?
28. How to find the current shell which you are using in UNIX?

1.

```
satva@satvam MINGW64 ~  
$ #!/bin/bash  
src_dir="/c/Users/satva/folder1"  
dst_dir="/c/Users/satva/folder3"  
mkdir -p "$dst_dir"  
cp "$src_dir"/* "$dst_dir/"  
echo "Files copied successfully!"  
Files copied successfully!
```

2.

```
satva@satvam MINGW64 ~  
$ #!/bin/bash  
file="/c/Users/satva/f1.txt"  
c=$(cat "$file" | wc -c)  
w=$(cat "$file" | wc -w)  
l=$(grep -c "." "$file")  
echo "Number of characters in file is $c"  
echo "Number of words in file is $w"  
echo "Number of lines in file is $l"  
Number of characters in file is 29  
Number of words in file is 6  
Number of lines in file is 5
```



3.

```
satvam@cseds-OptiPlex-5400-cv110:~/sf$  
#!/bin/bash  
echo "Enter file name: "  
read file  
if [ -e "$file" ]  
then  
echo "File Name: $file"  
echo "Size: $(du -h $file | cut -f1)"  
echo "Type: $(file -b $file)"  
echo "Permissions: $(ls -l $file | cut -d" " -f1)"  
else  
echo "File does not exist."  
fi  
Enter file name:  
test.txt  
File Name: test.txt  
Size: 4.0K  
Type: ASCII text  
Permissions: -rw-rw-r--
```

4.

```
satvam@cseds-OptiPlex-5400-cv110:~/sf$  
#!/bin/bash  
echo "Enter a number:"  
read num  
fact=1  
for (( i=1; i<=$num; i++ ))  
do  
fact=$((fact * i))  
done  
echo "Factorial of $num is: $fact"  
Enter a number:  
6  
Factorial of 6 is: 720
```



5.

```
satvam@cseds-OptiPlex-5400-cv110:~/sf$  
echo "Enter a integer number"  
read n  
a=1  
echo "Printing numbers:"  
while [ $a -le $n ]  
do  
echo "$a"  
a=`expr $a + 1`  
done  
Enter a integer number  
11  
Printing numbers:  
1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11
```

6.

```
satvam@cseds-OptiPlex-5400-cv110:~/sf$  
read -p "Enter end point for Fibonacci series: " n  
a=0  
b=1  
c=0  
for i in $(seq 1 $n); do  
echo -n "$a "  
c=$((a+b))  
a=$b  
b=$c  
done  
echo  
Enter end point for Fibonacci series: 4  
0 1 1 2
```



7.

```
satvam@cseds-OptiPlex-5400-cv110:~/sf$  
#!/bin/bash  
read -p "Enter a number: " num  
flag=0  
if [ $num -eq 1 ]; then  
echo "1 is neither prime nor composite."  
else  
for ((i=2;i<=$num/2;i++))  
do  
if [ $((($num%i)) -eq 0 ); then  
flag=1  
break  
fi  
done  
if [ $flag -eq 1 ]; then  
echo "$num is not a prime number."  
else  
echo "$num is a prime number."  
fi  
fi  
Enter a number: 17  
17 is a prime number.
```

8.

```
satvam@cseds-OptiPlex-5400-cv110:~/sf$  
echo  
echo "==== Greatest of 3 Numbers ===="
```

```
read -p "Please enter 3 numbers: " a b c  
if [ $a = $b -a $a = $c ]; then  
echo "All numbers are equal!"  
elif [ $a -gt $b -a $a -gt $c ]; then  
echo "$a is the greatest number!"  
elif [ $b -gt $a -a $b -gt $c ]; then  
echo "$b is the greatest number!"  
else  
echo "$c is the gratest number!"  
fi
```

```
==== Greatest of 3 Numbers ====  
Please enter 3 numbers: 22 3 90  
90 is the gratest number!
```




9.

```
satvam@cseds-OptiPlex-5400-cv110:~/sf$  
#!/bin/bash  
read -p "Enter a Number:" n  
rev=0  
while [ $n -gt 0 ]  
do  
rem=$(( $n % 10 ))  
rev=$(( $rev * 10 + $rem ))  
n=$(( $n / 10 ))  
done  
echo "Reverse of num is $rev"  
Enter a Number:4321  
Reverse of num is 1234
```

10.

```
satvam@cseds-OptiPlex-5400-cv110:~/sf$  
read -p "Enter a 5 Digit Number: " n  
add=0  
while [ $n -gt 0 ]; do  
rem=$((n%10))  
add=$((add+rem))  
n=$((n/10))  
done  
echo "Addition = $add"  
Enter a 5 Digit Number: 12221  
Addition = 8
```

11.

```
satvam@cseds-OptiPlex-5400-cv110:~/sf$  
echo "List of users currently logged in:"  
who  
List of users currently logged in:  
cseds-student :1          2023-05-04 10:21 (:1)  
satvam    tty3          2023-05-04 12:11 (tty3)
```

12.



```
satvam@cseds-OptiPlex-5400-cv110:~/sf$  
read -p "Enter 2 numbers: " a b  
op=1  
while [ $op -lt 5 ]; do  
echo  
echo "1. Addition"  
echo "2. Subtraction"  
echo "3. Multiplication"  
echo "4. Division"  
echo "5. Exit"  
read -p "Enter the Operation to be performed: " op  
echo  
case $op in  
1)  
ans=$((a+b))  
echo "Addition: $ans";;  
2)  
ans=$((a-b))  
echo "Subtraction: $ans";;  
3)  
ans=$((a*b))  
echo "Multiplication: $ans";;  
4)  
ans=$((a/b))  
echo "Division: $ans";;  
5)  
;;  
*)  
echo "Invalid Input! Exiting Code.."  
op=5;;  
esac  
done  
Enter 2 numbers: 8 2  
  
1. Addition  
2. Subtraction  
3. Multiplication  
4. Division  
5. Exit  
Enter the Operation to be performed: 1  
  
Addition: 10  
  
1. Addition  
2. Subtraction  
3. Multiplication  
4. Division  
5. Exit  
Enter the Operation to be performed: 5
```




13.

```
satva@satvam MINGW64 ~/folder1
$ src_file="/c/Users/satva/folder1/file1.txt"
dest_file="/c/Users/satva/folder1/file2.txt"
cp "$src_file" "$dest_file"
echo "Contents have been copied "
echo
echo "File2 content's are :- "
cat "$dest_file"
Contents have been copied

File2 content's are :-
I have executed this code successfully !!!!
```

14.

```
satvam@cseds-OptiPlex-5400-cv110:~/sf$
read -p "Enter a number to generate multiplication table : " num
read -p "Enter the range for the multiplication table : " range
for (( i=1; i<=range; i++ ))
do
echo "$num x $i = $((num*i))"
done
Enter a number to generate multiplication table : 9
Enter the range for the multiplication table : 5
9 x 1 = 9
9 x 2 = 18
9 x 3 = 27
9 x 4 = 36
9 x 5 = 45
```

15.

```
satvam@cseds-OptiPlex-5400-cv110:~/sf$
dir="/home/satvam/sf"
num_files=$(ls -1 "$dir" | wc -l)
echo "Number of files in $dir: $num_files"
Number of files in /home/satvam/sf: 5
```



16.

```
satvam@cseds-OptiPlex-5400-cv110:~/sf$  
#!/bin/bash  
read -p "Enter the file name : " file_name  
read -p "Enter the search string : " search_string  
char_count=$(grep -o "$search_string" "$file_name" | wc -m)  
word_count=$(grep -o "$search_string" "$file_name" | wc -w)  
line_count=$(grep -c "$search_string" "$file_name")  
echo "Number of matched characters: $char_count"  
echo "Number of matched words: $word_count"  
echo "Number of matched lines: $line_count"  
Enter the file name : sample.txt  
Enter the search string : My Name Is Satvam  
Number of matched characters: 18  
Number of matched words: 4  
Number of matched lines: 1
```

17.

```
satvam@cseds-OptiPlex-5400-cv110:~/sf$  
#!/bin/bash  
read -p "Enter file name : " filename  
if [ ! -f $filename ]; then  
echo "File not found."  
exit 1  
fi  
num_chars=$(wc -c $filename | awk '{print $1}')num_words=$(wc -w $filename | awk '{print $1}')num_lines=$(wc -l $filename | awk '{print $1}')echo "Number of characters: $num_chars"  
echo "Number of words: $num_words"  
echo "Number of lines: $num_lines"  
Enter file name : example.txt  
Number of characters: 674  
Number of words: 119  
Number of lines: 1
```



18.

```
satvam@cseds-OptiPlex-5400-cv110:~/sf$  
#!/bin/bash  
read -p "Enter a letter :" letter  
echo "Files starting with '$letter':"  
ls -l $letter*  
Enter a letter :f  
Files starting with 'f':  
file_path:  
total 4  
drwxrwxr-x 2 satvam satvam 4096 May  4 12:20 folder3  
  
folder1:  
total 0  
  
folder3:  
total 0
```

19. Already Done Above



20.

```
satvam@cseds-OptiPlex-5400-cv110:~/sf$  
read -p "Enter the file name : " filename  
read -p "Enter the pattern to replace : " pattern  
read -p "Enter the replacement string : " replacement  
read -p "Enter the Nth occurrence to replace : " Nth_occurrence  
if [ -f "$filename" ]  
then  
count=$(grep -o "$pattern" "$filename" | wc -l)  
if [ "$count" -ge "$Nth_occurrence" ]  
then  
sed -i "${Nth_occurrence}s/$pattern/$replacement/" "$filename"  
echo "Nth occurrence of $pattern has been replaced with $replacement"  
else  
echo "Nth occurrence does not exist in the file"  
fi  
else  
echo "File not found"  
fi  
Enter the file name : sample.txt  
Enter the pattern to replace : My  
Enter the replacement string : Your  
Enter the Nth occurrence to replace : 1  
Nth occurrence of My has been replaced with Your  
satvam@cseds-OptiPlex-5400-cv110:~/sf$ cat sample.txt  
Your Name Is Satvam
```

21.

```
satvam@cseds-OptiPlex-5400-cv110:~/sf$  
#!/bin/bash  
read -p "Enter degree celsius temperature: " celsius  
fahrenheit=`echo "scale=4; $celsius*1.8 + 32" | bc`  
echo "$celsius degree celsius is equal to $fahrenheit degree fahrenheit"  
Enter degree celsius temperature: 45  
45 degree celsius is equal to 113.0 degree fahrenheit
```



22.

```
satvam@cseds-OptiPlex-5400-cv110:~/sf$  
#!/bin/bash  
read -p "Enter a number :" base  
read -p "Enter a power :" power  
result=1  
while [ $power -gt 0 ]  
do  
result=$((result * base))  
power=$((power - 1))  
done  
echo "$result"  
Enter a number :5  
Enter a power :3  
125
```

23. Already Done Above

24.

```
satvam@cseds-OptiPlex-5400-cv110:~/sf$  
#!/bin/bash  
read -p "Enter a year : " year  
if [ $((year % 4)) -eq 0 ] && [ $((year % 100)) -ne 0 ] || [ $((year % 400)) -eq  
0 ]  
then  
echo "$year is a leap year"  
else  
echo "$year is not a leap year"  
fi  
Enter a year : 2023  
2023 is not a leap year
```

25.

```
satvam@cseds-OptiPlex-5400-cv110:~/sf$  
#!/bin/bash  
read -p "Enter a number : " num  
if ((num%2==0)); then  
echo "$num is even"  
else  
echo "$num is odd"  
fi  
Enter a number : 296  
296 is even
```



26.

Shell scripts are useful in situations where you need to automate repetitive tasks, perform system administration tasks, or execute commands in a specific order. They can also be used to create simple programs or utilities that perform a specific task.

Some examples of when you might use a shell script include:

- Automating backups of important files or databases
- Running system maintenance tasks, such as clearing log files or removing temporary files
- Performing batch operations on files, such as renaming or converting them
- Setting up or configuring new software installations
- Running tests or performing other quality assurance tasks
- Automating deployments or other routine tasks in a software development environment

Overall, shell scripts can help you save time and reduce errors by automating repetitive tasks and ensuring that commands are executed consistently and correctly.

27.

```
satvam@cseds-OptiPlex-5400-cv110:~/sf$  
which bash  
/usr/bin/bash
```

28.

```
satvam@cseds-OptiPlex-5400-cv110:~/sf$  
echo $SHELL  
/bin/bash
```