

Experiment [5] : [Shell Programming]

Name: Prapti Uniyal Roll.: 590028360 Date: 10-09-2025

AIM:

- [To Learn Basic Conditional Statements in Bash Scripting]

Requirements:

- [Any Linux Distro, any kind of text editor (vs code, vim, nano, etc)]

Theory:

- [Basic usage of conditions and arrays in bash scripting.]

1- ARRAYS IN BASH

```
prapti1011@asus:~$ vim arr.sh
prapti1011@asus:~$ cat arr.sh
arr=(10 20 30 40)
echo "First: ${arr[0]}"
echo "All: ${arr[@]}"

prapti1011@asus:~$ ./arr.sh
-bash: ./arr.sh: Permission denied
prapti1011@asus:~$ chmod +x arr.sh
prapti1011@asus:~$ ./arr.sh
First: 10
All: 10 20 30 40
```

2- CONDITIONAL STATEMENTS

if, else, elif, case, etc.

Procedure & Observations:

Task 1: [Prime Number Check]

Task Statement:

- [To check if the number given by the user is a prime number or not.]

Explanation:


- [using if else loop wap to check if the number is a prime number or not.]

Command(s):

```
#!/bin/bash
echo "Enter a number: "
read num
flag=0
for ((i=2; i<=num/2; i++))
do
    if [ $((num % i)) -eq 0 ]
    then
        flag=1
        break
    fi
done

if [ $flag -eq 0 ]
then
    echo "$num is a prime number."
else
    echo "$num is not a prime number."
fi
```

Output:

A terminal window with a black background. The prompt is 'prapti1011@asus:~\$' in green. The command 'vim primecheck.sh' is entered in white text.

```
prapti1011@asus:~$ vim primecheck.sh
```

```
prapti1011@asus:~$ ./primecheck.sh
enter a number:
7
7 is a prime number.
prapti1011@asus:~$ ./primecheck.sh
enter a number:
54
54 is not a prime number.
```

Task 2: [Sum of Digits]

Task Statement:

- [Take input from user and give the sum of two digits.]

Explanation:

- [This script will take input from user and will give the following output.]

Command(s):

```
#!/bin/bash
echo "Enter a number: "
read num
sum=0
while [ $num -gt 0 ]
do
    digit=$((num % 10))
    sum=$((sum + digit))
    num=$((num / 10))
done

echo "Sum of digits: $sum"
```

Output:

```
prapti1011@asus:~$ vim sumcheck.sh
prapti1011@asus:~$ ./sumcheck.sh
-bash: ./sumcheck.sh: Permission denied
prapti1011@asus:~$ chmod +x sumcheck.sh
prapti1011@asus:~$ ./sumcheck.sh
enter a number:
1234
Sum of digits: 10
prapti1011@asus:~$ |
```

Task 3: [Armstrong Number Check]

Task Statement:

- [Take input user and give the sum of Armstrong number of n digits is a number equal to the sum of its digits raised to the power n. Example: $153 = 1^3 + 5^3 + 3^3$]

Explanation:

- [This script will tell if the number entered by the user is an armstrong number or not.]

Command(s):

```
#!/bin/bash
echo "Enter a number: "
read num
temp=$num
n=${#num}    # number of digits
sum=0

while [ $temp -gt 0 ]
do
    digit=$((temp % 10))
    sum=$((sum + digit**n))
    temp=$((temp / 10))
done

if [ $sum -eq $num ]
then
    echo "$num is an Armstrong number."
else
    echo "$num is not an Armstrong number."
fi
```

Output:

```
prapti1011@asus:~$ vim armcheck.sh
prapti1011@asus:~$ ./armcheck.sh
-bash: ./armcheck.sh: Permission denied
prapti1011@asus:~$ chmod +x armcheck.sh
]prapti1011@asus:~$ ./armcheck.sh
enter a number:
153
153 is an Armstrong number.
prapti1011@asus:~$ ./armcheck.sh
enter a number:
123
123 is not an Armstrong number.
prapti1011@asus:~$ |
```

Result:

- The tasks were successfully completed for Basic Shell Programming.