

Ex. No. : 3a

Date : 07.02.2025

Register No. : 230701385

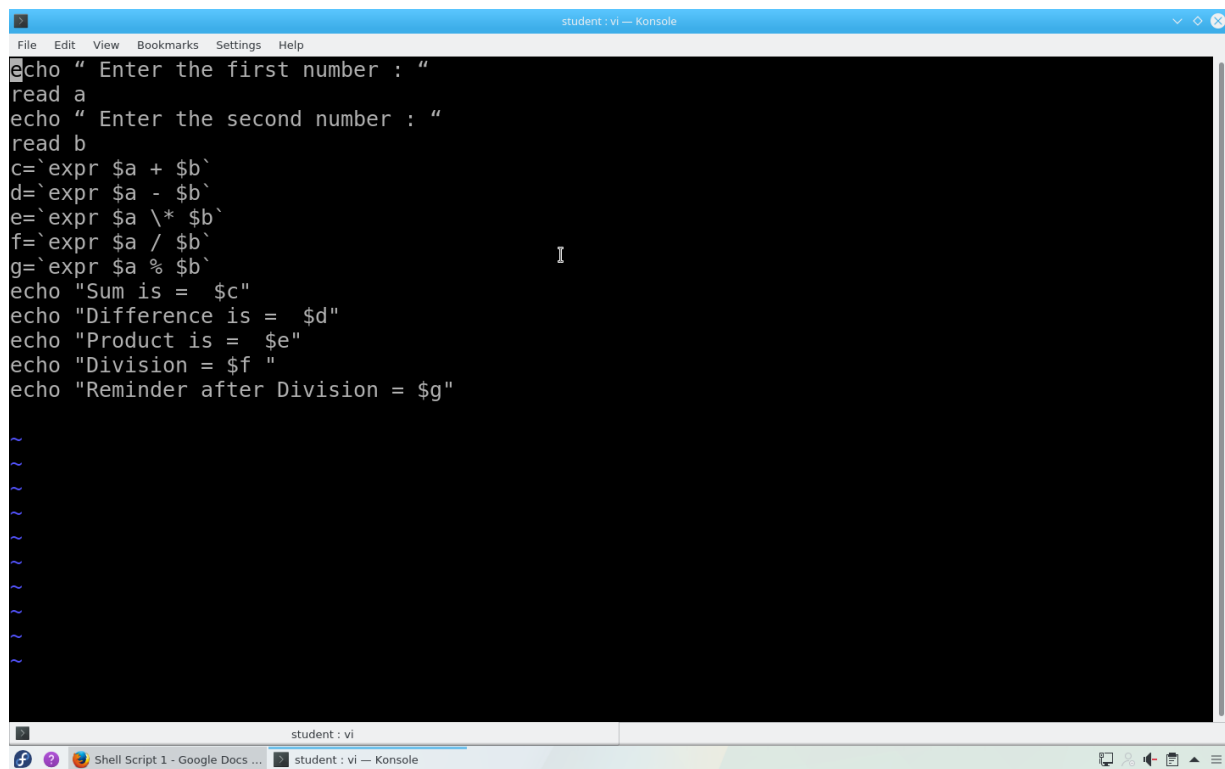
Name : VISHWAK S

Shell Script - Basic Calculator

Aim:

To write a Shell Script to display a basic calculator.

Program:

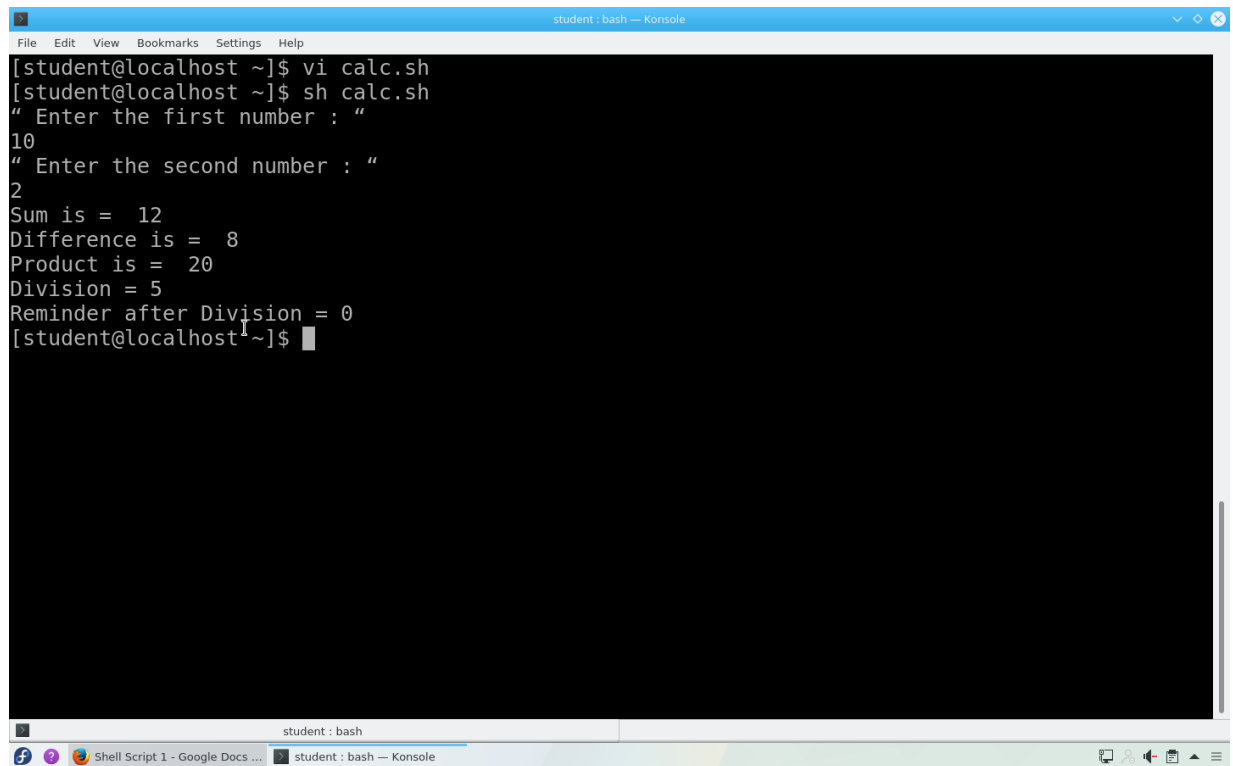


```
student: vi — Konsole
File Edit View Bookmarks Settings Help
echo " Enter the first number : "
read a
echo " Enter the second number : "
read b
c=`expr $a + $b`
d=`expr $a - $b`
e=`expr $a \* $b`
f=`expr $a / $b`
g=`expr $a % $b`
echo "Sum is = $c"
echo "Difference is = $d"
echo "Product is = $e"
echo "Division = $f "
echo "Reminder after Division = $g"

~
~
~
~
~
~
~
~
~
~
```

The screenshot shows a terminal window titled "student: vi — Konsole". The terminal displays a shell script for a basic calculator. The script prompts the user to enter two numbers, reads them into variables 'a' and 'b', and then calculates the sum, difference, product, division, and remainder using the 'expr' command. The results are displayed using 'echo'. The terminal also shows a list of tilde characters (~) at the bottom, indicating the user's home directory. The window's title bar includes a menu bar with options: File, Edit, View, Bookmarks, Settings, and Help. The bottom of the window shows a taskbar with icons for a web browser, a Google Docs document titled "Shell Script 1 - Google Docs ...", and the terminal window itself.

Output:



```
student : bash — Konsole
File Edit View Bookmarks Settings Help
[student@localhost ~]$ vi calc.sh
[student@localhost ~]$ sh calc.sh
" Enter the first number : "
10
" Enter the second number : "
2
Sum is = 12
Difference is = 8
Product is = 20
Division = 5
Reminder after Division = 0
[student@localhost ~]$
```

Result:

Hence the Shell Script to display a basic calculator has been successfully executed.

Date : 07.02.2025

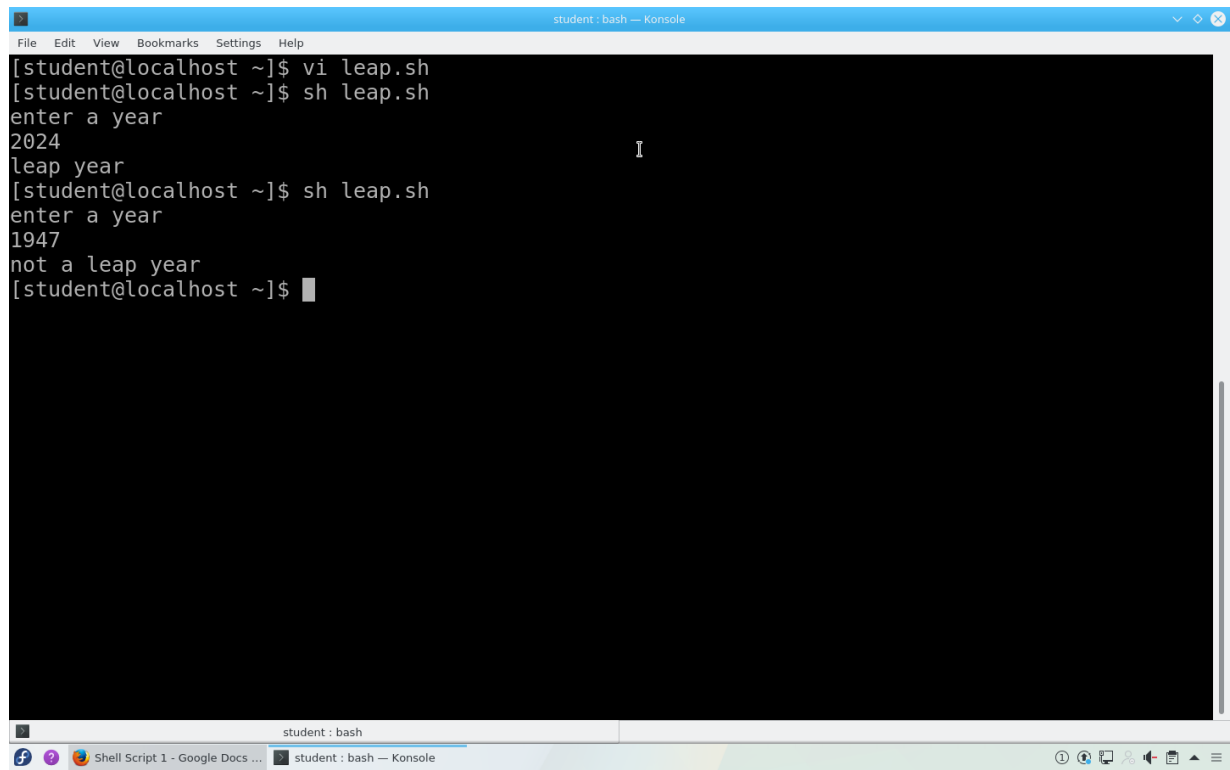
Name : VISHWAK S

A screenshot of a terminal window titled "student : vi — Konsole". The terminal displays a shell script for checking if a year is a leap year. The script uses `echo` for prompts and output, `read` for input, and an `if` statement with modulo arithmetic to determine leap years. The cursor is at the end of the last line of code. The bottom status bar shows the file name as "student : vi".

```
echo "enter a year"
read year
if [ $((year%4)) -eq 0 ]
then
    echo "leap year"
else
    echo "not a leap year"
fi
```

"leap.sh" 8L, 111C

Output:



```
[student@localhost ~]$ vi leap.sh
[student@localhost ~]$ sh leap.sh
enter a year
2024
leap year
[student@localhost ~]$ sh leap.sh
enter a year
1947
not a leap year
[student@localhost ~]$
```

The screenshot shows a terminal window titled "student : bash --- Konsole". The terminal displays the following sequence of commands and outputs:

- `[student@localhost ~]$ vi leap.sh`
- `[student@localhost ~]$ sh leap.sh`
- enter a year
- 2024
- leap year
- `[student@localhost ~]$ sh leap.sh`
- enter a year
- 1947
- not a leap year
- `[student@localhost ~]$`

The terminal window has a menu bar with "File", "Edit", "View", "Bookmarks", "Settings", and "Help". The bottom status bar shows the current directory as "~" and the shell as "bash".

Result:

Hence the Shell Script to test given year is leap or not using conditional statement has been executed successfully.

Ex. No. : 3b

Date : 08.02.2025

Register No. : 230701385

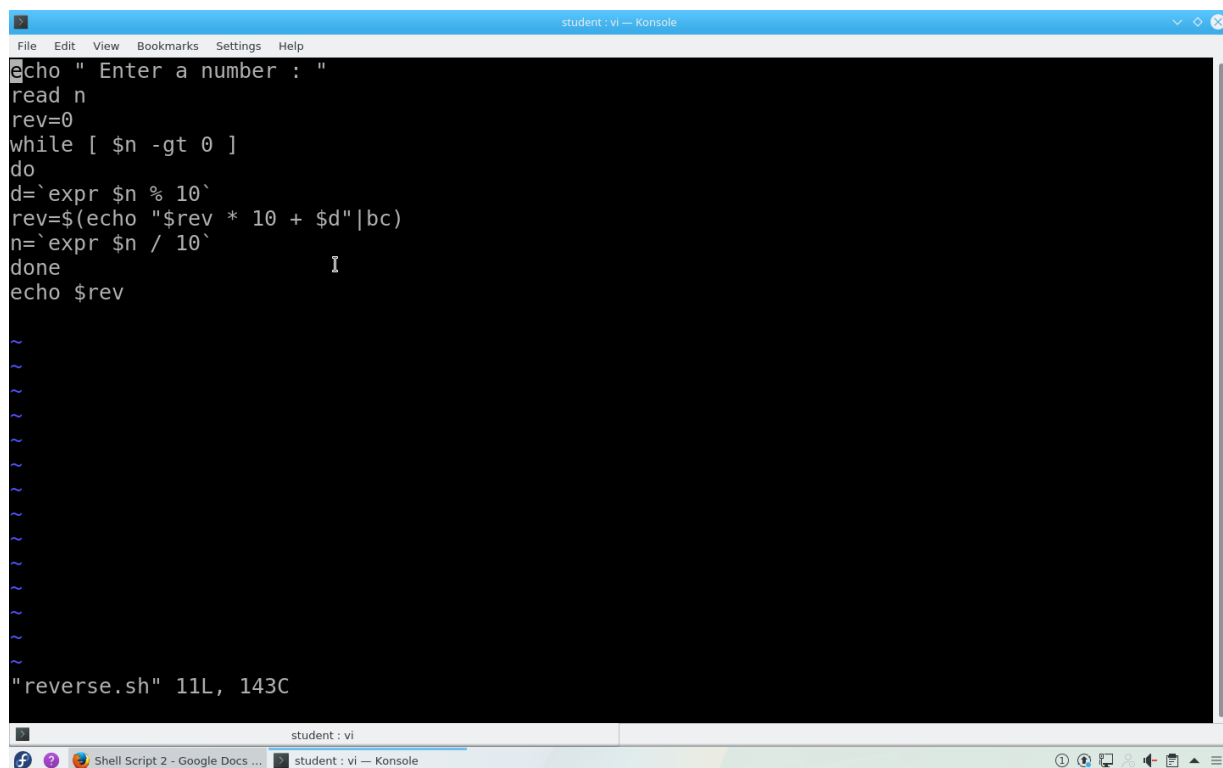
Name : VISHWAK S

Shell Script - Reverse the Number

Aim:

To write a Shell script to reverse a given digit using a looping statement.

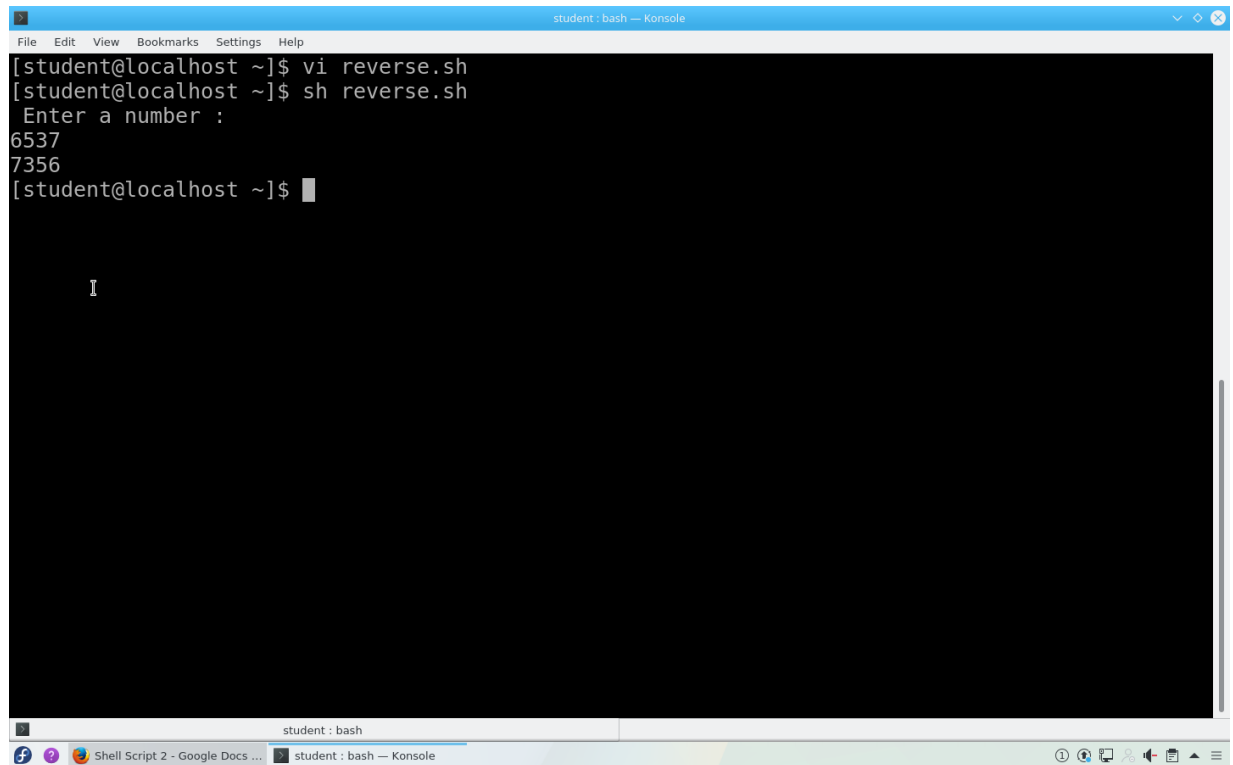
Program:

A screenshot of a terminal window titled "student : vi — Konsole". The window displays a shell script for reversing a number. The script uses a while loop to repeatedly extract the last digit of a number and build the reversed number. The script is as follows:

```
echo " Enter a number : "  
read n  
rev=0  
while [ $n -gt 0 ]  
do  
d=`expr $n % 10`  
rev=$(echo "$rev * 10 + $d"|bc)  
n=`expr $n / 10`  
done  
echo $rev
```

The terminal shows several tilde (~) characters representing input, and the command prompt at the bottom indicates the script is 11 lines long and 143 characters wide. The terminal window is part of a desktop environment with a taskbar at the bottom showing other open applications like "Shell Script 2 - Google Docs ...".

Output:



```
[student@localhost ~]$ vi reverse.sh
[student@localhost ~]$ sh reverse.sh
Enter a number :
6537
7356
[student@localhost ~]$
```

The screenshot shows a terminal window titled "student : bash -- Konsole". The user has created a file named "reverse.sh" using the "vi" editor and then executed it with "sh reverse.sh". The script prompts the user to "Enter a number :", and the user has entered "6537". The script has output "7356", which is the reverse of the input. The terminal prompt is now "[student@localhost ~]\$".

Result:

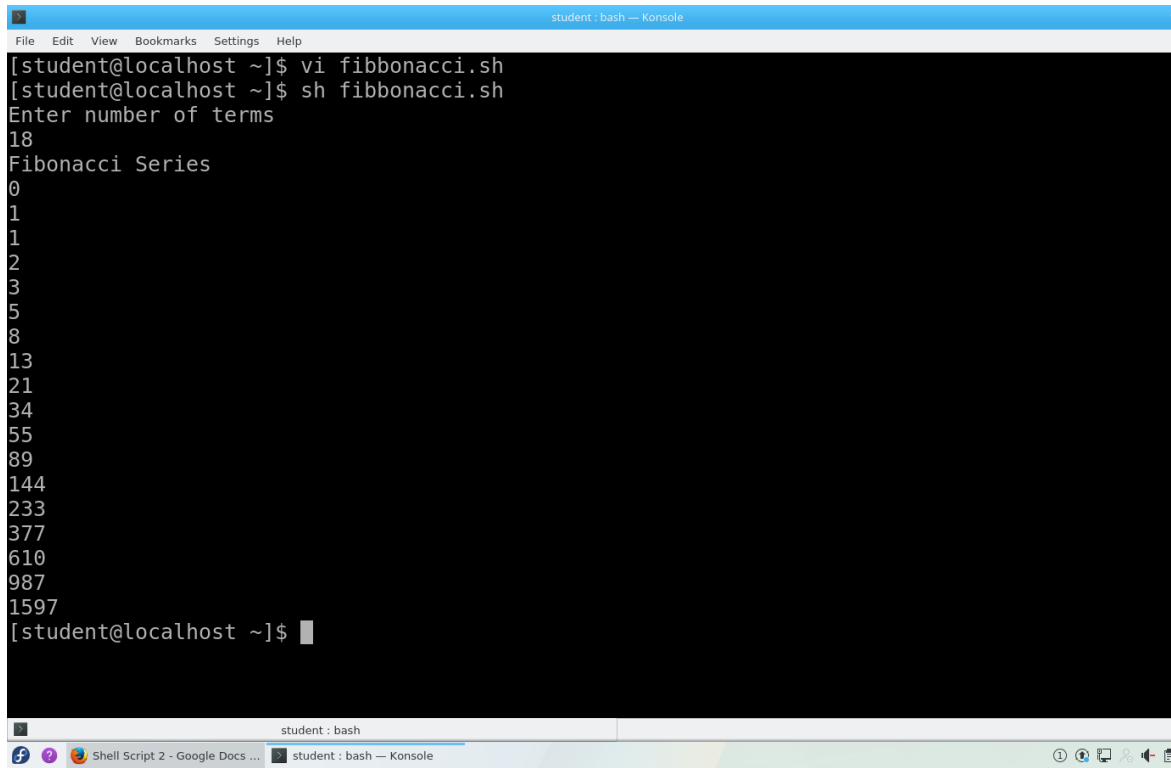
Hence the Shell script to reverse a given digit using a looping statement has been executed successfully.

Date : 08.02.2025

Name : VISHWAK S

[illegible]

Output:



The screenshot shows a terminal window titled "student : bash — Konsole". The user enters the command `vi fibbonacci.sh` to edit a script, followed by `sh fibbonacci.sh` to execute it. The script prompts for the number of terms, and the user enters `18`. The script then outputs the Fibonacci series: `Fibonacci Series` followed by the numbers 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987, and 1597. The terminal window has a menu bar with "File", "Edit", "View", "Bookmarks", "Settings", and "Help". The bottom of the window shows a taskbar with icons for Facebook, a help icon, a Google Docs document titled "Shell Script 2 - Google Docs ...", and the terminal window itself.

```
[student@localhost ~]$ vi fibbonacci.sh
[student@localhost ~]$ sh fibbonacci.sh
Enter number of terms
18
Fibonacci Series
0
1
1
2
3
5
8
13
21
34
55
89
144
233
377
610
987
1597
[student@localhost ~]$
```

Result:

Hence the Shell script to generate a Fibonacci series using for loop has been executed successfully.