

Shell Programming (bash)

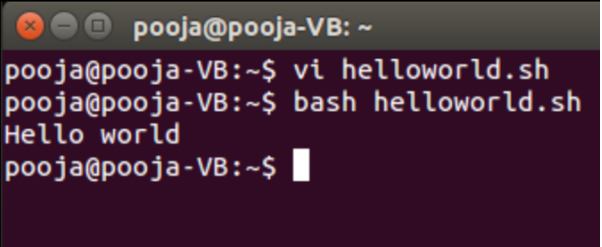
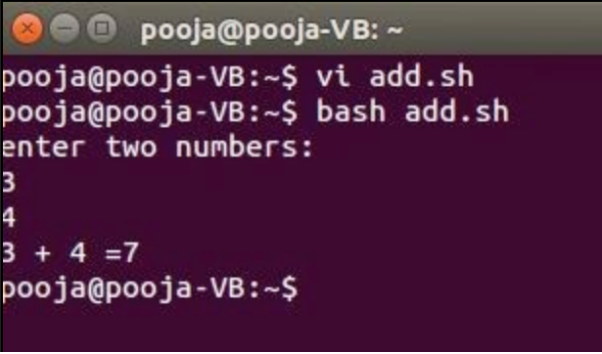
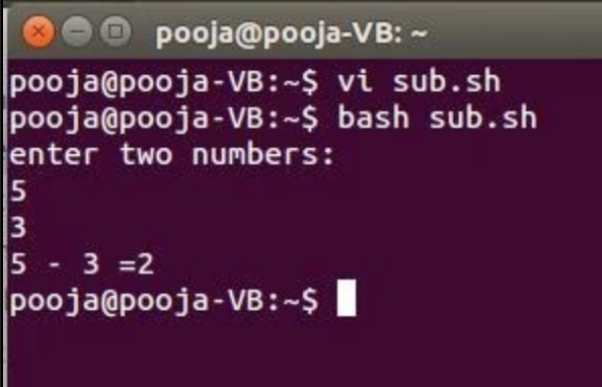
1. Exercises

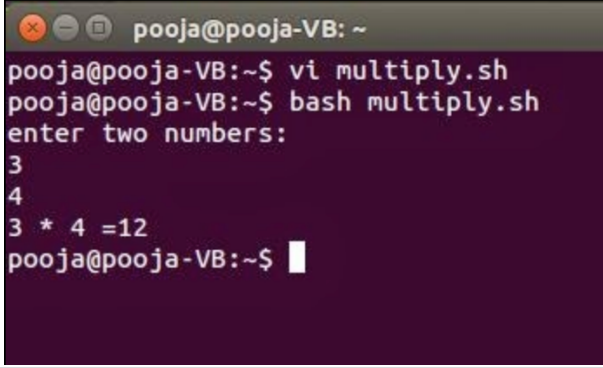
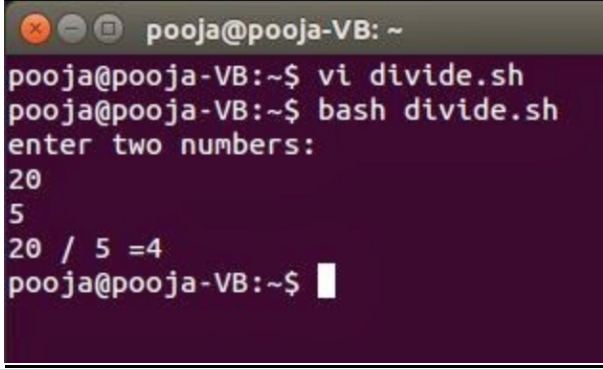
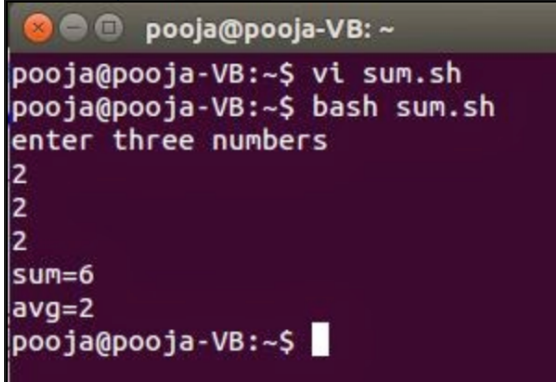
1. Hello, world!
2. Addition of two numbers
3. Subtraction of two numbers
4. Multiplication of two numbers
5. Division of two numbers
6. Sum and average of three numbers
7. Square and cube of the given number
8. Multiplication table of the given number
9. Print natural numbers up to n
10. Check whether the given number is even or odd
11. Swapping of two numbers with using third variable
12. Swapping of two numbers without using third variable
13. Factorial

2. Tasks

1. Time Converter
2. Temperature C \leftrightarrow F Converter

1. Exercises

Nr	Task & Code	Remark & Output
1	<pre>#!/bin/bash echo "Hello, world!"</pre>	 <pre>pooja@pooja-VB: ~ pooja@pooja-VB:~\$ vi helloworld.sh pooja@pooja-VB:~\$ bash helloworld.sh Hello world pooja@pooja-VB:~\$</pre>
2	<pre>#!/bin/bash echo "Enter two numbers:" read a read b c=`expr "\$a" + "\$b" ` echo "\$a + \$b = "\$c</pre>	 <pre>pooja@pooja-VB: ~ pooja@pooja-VB:~\$ vi add.sh pooja@pooja-VB:~\$ bash add.sh enter two numbers: 3 4 3 + 4 =7 pooja@pooja-VB:~\$</pre>
3	<pre>#!/bin/bash echo "Enter two numbers:" read a read b c=`expr "\$a" - "\$b" ` echo "\$a - \$b = "\$c</pre>	 <pre>pooja@pooja-VB: ~ pooja@pooja-VB:~\$ vi sub.sh pooja@pooja-VB:~\$ bash sub.sh enter two numbers: 5 3 5 - 3 =2 pooja@pooja-VB:~\$</pre>

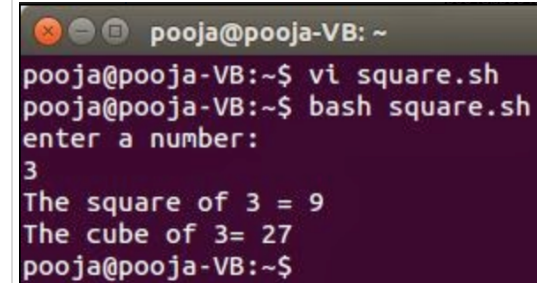
<p>4 Multiplication of two numbers</p> <pre>#!/bin/bash echo "Enter two numbers:" read a read b c=`expr "\$a" * "\$b" ` echo "\$a * \$b = "\$c</pre>	 <pre>pooja@pooja-VB: ~ pooja@pooja-VB:~\$ vi multiply.sh pooja@pooja-VB:~\$ bash multiply.sh enter two numbers: 3 4 3 * 4 =12 pooja@pooja-VB:~\$</pre>
<p>5 Division of two numbers</p> <pre>#!/bin/bash echo "Enter two numbers:" read a read b c=`expr "\$a" V "\$b" ` echo "\$a / \$b = "\$c</pre>	 <pre>pooja@pooja-VB: ~ pooja@pooja-VB:~\$ vi divide.sh pooja@pooja-VB:~\$ bash divide.sh enter two numbers: 20 5 20 / 5 =4 pooja@pooja-VB:~\$</pre>
<p>6 Sum and average of three numbers</p> <pre>#!/bin/bash echo "Enter three numbers" read a read b read c sum=`expr "\$a" + "\$b" + "\$c"` avg=`expr "\$sum" V 3 ` echo "sum="\$sum echo "avg="\$avg</pre>	 <pre>pooja@pooja-VB: ~ pooja@pooja-VB:~\$ vi sum.sh pooja@pooja-VB:~\$ bash sum.sh enter three numbers 2 2 2 sum=6 avg=2 pooja@pooja-VB:~\$</pre>

7 Square and cube of the given number

```
#!/bin/bash

echo -n "Enter a number:"
read a
sq=`expr "$a" \* "$a" `
cube=`expr "$a" \* "$a" \* "$a" `

echo "The square of $a =" $sq
echo "The cube of $a=" $cube
```



```
pooja@pooja-VB: ~
pooja@pooja-VB:~$ vi square.sh
pooja@pooja-VB:~$ bash square.sh
enter a number:
3
The square of 3 = 9
The cube of 3= 27
pooja@pooja-VB:~$
```

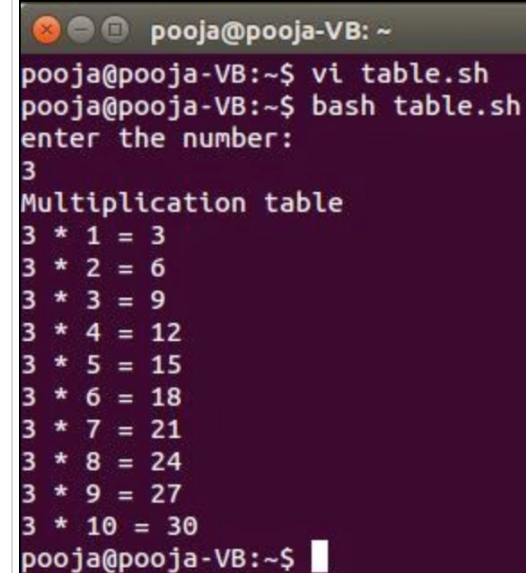
8 Multiplication table of the given number

```
#!/bin/bash

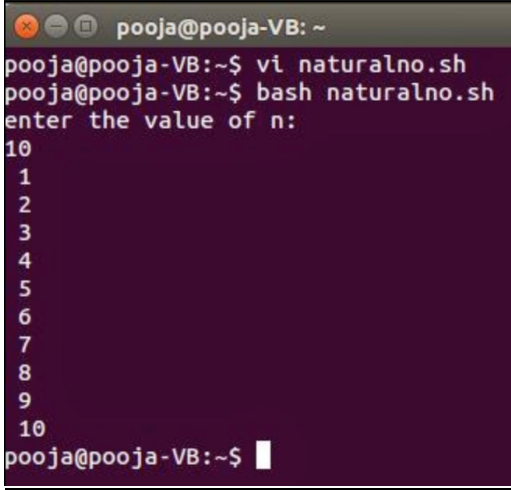
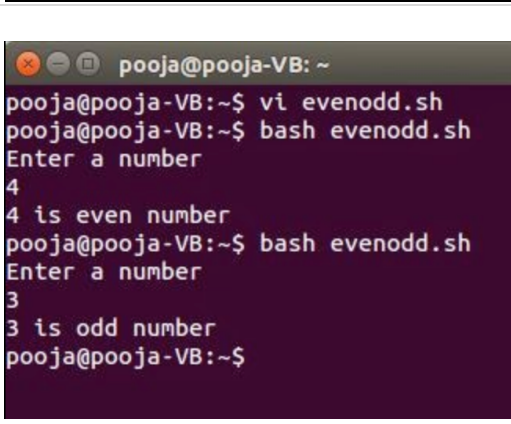

echo -n "Enter the number:"
read a

echo "Multiplication table"

for((i=1;i<=10;i++))
do
echo "$a * $i = `expr $a \* $i`"
done
```



```
pooja@pooja-VB: ~
pooja@pooja-VB:~$ vi table.sh
pooja@pooja-VB:~$ bash table.sh
enter the number:
3
Multiplication table
3 * 1 = 3
3 * 2 = 6
3 * 3 = 9
3 * 4 = 12
3 * 5 = 15
3 * 6 = 18
3 * 7 = 21
3 * 8 = 24
3 * 9 = 27
3 * 10 = 30
pooja@pooja-VB:~$
```

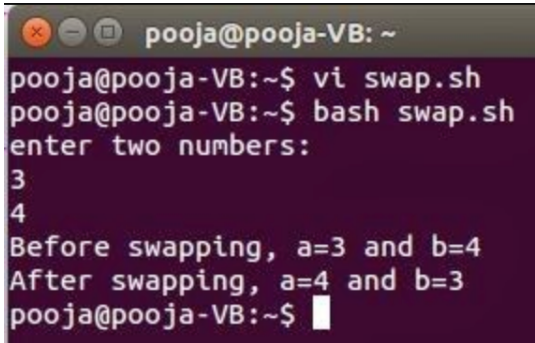
9	<p>Print natural numbers up to n</p> <pre>#!/bin/bash echo -n "Enter the value of n:" read n for((i=1;i<=n;i++)) do echo " \$i" done</pre>	 <pre>pooja@pooja-VB: ~ pooja@pooja-VB:~\$ vi naturalno.sh pooja@pooja-VB:~\$ bash naturalno.sh enter the value of n: 10 1 2 3 4 5 6 7 8 9 10 pooja@pooja-VB:~\$</pre>
10	<p>Check whether the given number is even or odd.</p> <pre>#!/bin/bash echo -n "Enter a number" read a if test `expr \$a % 2` -eq 0 then echo "\$a is even number" else echo "\$a is odd number" fi</pre>	 <pre>pooja@pooja-VB: ~ pooja@pooja-VB:~\$ vi evenodd.sh pooja@pooja-VB:~\$ bash evenodd.sh Enter a number 4 4 is even number pooja@pooja-VB:~\$ bash evenodd.sh Enter a number 3 3 is odd number pooja@pooja-VB:~\$</pre>
11	<p>Swapping of two numbers using third variable</p> <pre>#!/bin/bash echo "Enter two numbers:" read a read b echo "Before swapping, a=\$a and b=\$b" z=\$a a=\$b b=\$z echo "After swapping, a=\$a and b=\$b"</pre>	 <pre>pooja@pooja-VB: ~ pooja@pooja-VB:~\$ vi swap1.sh pooja@pooja-VB:~\$ bash swap1.sh enter two numbers: 4 5 Before swapping, a=4 and b=5 After swapping, a=5 and b=4 pooja@pooja-VB:~\$</pre>

12 Swapping of two numbers without using third variable

```
#!/bin/bash

echo "Enter two numbers:"
read a
read b
echo "Before swapping, a=$a and b=$b"
a=$((a+b))
b=$((a-b))
a=$((a-b))

echo "After swapping, a=$a and b=$b"
```



A terminal window titled 'pooja@pooja-VB: ~' showing the execution of a shell script named 'swap.sh'. The user enters 'vi swap.sh' to edit the script, then 'bash swap.sh' to run it. The script prompts for two numbers, '3' and '4'. It then displays the output: 'Before swapping, a=3 and b=4' and 'After swapping, a=4 and b=3'. The prompt returns to the user.

13 Factorial 0..20

```
#!/bin/bash

echo -n "Enter the value of n:"
read n

if test $n -eq 0
then
  f=1
else
  f=1
  for((i=1;i<=n;i++))
  do
    f=`expr $f \* $i`
  done
fi

echo "$n! = $f"
```

2. Tasks.

Nr	Task	Задание
1	<p>Time Converter Create a shell program to convert time in seconds to human format Days-HH:MM:SS. When given seconds, the script will convert them to Days-HH:MM:SS.</p>	<p>Time Converter Создайте программу-оболочку для преобразования времени в секундах в человеческий формат Дни-ЧЧ:ММ:СС. Получая секунды, скрипт преобразует их в Days-HH:MM:SS.</p>
2	<p>Temperature C<->F Converter Create a shell program using the following formulas that effect temperature conversion. Given degrees Celsius, he converts them to Fahrenheit and vice versa. $C = 5 * (F - 32) / 9$ $F = 9 * C / 5 + 32$</p>	<p>Temperature C<->F Converter Создайте программу-оболочку, используя следующие формулы для преобразования температуры. Получив градусы Цельсия, он конвертирует их в градусы Фаренгейта и наоборот. $C = 5 * (F - 32) / 9$ $F = 9 * C / 5 + 32$</p>
3	<p>Develop a script that allows you to enter a decimal integer and a number base. Then it translates the number to the specified base.</p>	<p>Разработайте скрипт, который позволяет вводить целое десятичное число и числовую базу. Затем делает перевод числа на указанную базу.</p>
4	<p>Create a program that accepts any number of simple digits as arguments and shows for each digit whether it is even (odd) or odd (odd). An alphabetic or zero character will imply the invalid entry response (invalid). This program can be written with a for and a case operands. Example: <pre>\$ command 1 3 5 2 5 8 a 1 is odd 3 is odd 5 is odd 2 is even 5 is odd 8 is even a is invalid</pre></p>	<p>Создайте программу, которая принимает любое количество простых цифр в качестве аргументов и показывает для каждой цифры, является ли она четной (even) или нечетной (odd). Буквенный или нулевой символ будет означать неверный ответ на вход (invalid). Эта программа может быть написана с использованием for и case операторов. Пример: <pre>\$ command 1 3 5 2 5 8 a 1 is odd 3 is odd 5 is odd 2 is even 5 is odd 8 is even a is invalid</pre></p>
5	<p>Write a shell program that accepts a single file_name as an argument.</p>	<p>Напишите программу оболочки, которая принимает одно имя_файла в качестве аргумента.</p>

	<p>Test to verify if file is an executable. If the test is true (the file is executable), then the program should print the message "The argument file_name is already executable" to the terminal. If the test is false (the file is not executable), the program should change the permissions of the file to "rwxr-xr-x" and print the message "The argument file_name has become executable."</p>	<p>Проверьте, является ли файл исполняемым. Если тест верен (файл исполняемый), то программа должна напечатать на терминал сообщение «Аргумент имя_файла уже исполняемый». Если тест ложен (файл не является исполняемым), программа должна изменить права доступа к файлу на «rwxr-xr-x» и вывести сообщение «Аргумент имя_файла стал исполняемым».</p>
6	<p>Write a program that takes a file_name provided as an argument. Determine its file type. If the file type is a script text file, or an ASCII text file, or an English text file, show file with command less or pg. If the file is of another type, have the message display "The file cannot be printed". If the file does not exist, program should show "The file does not exist".</p>	<p>Напишите программу, которая принимает имя файла, предоставленное в качестве аргумента. Определите тип файла. Если тип является текстовым файлом сценария, или текстовым файлом ASCII или текстовым файлом на английском языке, покажи файл с помощью команды less или pg. Если файл другого типа, отобразите сообщение «Файл не может быть напечатан». Если файл не существует, программа должна показать «Файл не существует».</p>
7	<p>Write a program that takes a file_name and a directory_name. Have the script perform a search starting with the name of the directory provided. The information resulting from the search will not be sent to the terminal. The program will print "The file file_name can be found at: location" or "The file file_name is not in the directory directory_name".</p>	<p>Напишите программу, которая принимает имя_файла и имя_каталога. Пусть скрипт выполнит поиск, начиная с имени предоставленного каталога. Информация, полученная в результате поиска, не будет отправлена на терминал. Программа напечатает «Файл имя_файла можно найти по адресу: location» или «Файл имя_файла отсутствует в каталоге имя_каталога».</p>
8	<p>Write a program that accepts a file name and a destination directory for a copy of the file. If the destination_file already exists, ask the user if they overwrite or rename file. If the destination_file is renamed, rename the original with the number 1 added to the file name (that is, the destination_file is transformed into destination_file1).</p>	<p>Напишите программу, которая принимает имя файла и каталог назначения для копирования файла. Если целевой_файл уже существует, спросите пользователя хочет ли он перезаписать или переименовать файл. Если целевой_файл переименовывается, то переименуйте оригинал с номером 1, добавленным к имени файла (то есть целевой_файл преобразуется в целевой_файл1).</p>