

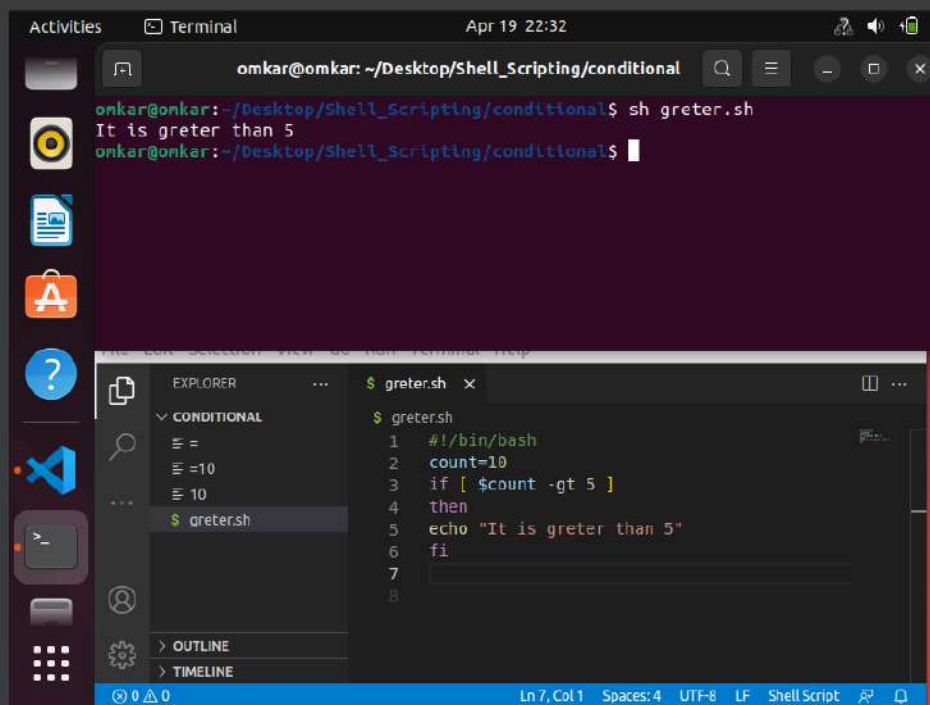
The screenshot shows a terminal window titled "Terminal" with a timestamp of "Apr 19 22:04". The terminal is running a shell script named "arg.sh" in the directory "~/Desktop/Shell_Scripting/expressions". The script's output is displayed as follows:

```
omkar@omkar: ~/Desktop/Shell_Scripting/expressions
omkar@omkar:~/Desktop/Shell_Scripting/expressions$ sh arg.sh ok okkk okkk
arg.sh,ok,okkk,okkk
ok okkk okkk
3
omkar@omkar:~/Desktop/Shell_Scripting/expressions$
```

Below the terminal window, a file editor shows the content of "arg.sh":

```
$ arg.sh
$ arg.sh
1 #!/bin/bash
2
3 echo $0,$1,$2,$3
4 echo $@
5 echo $#
```

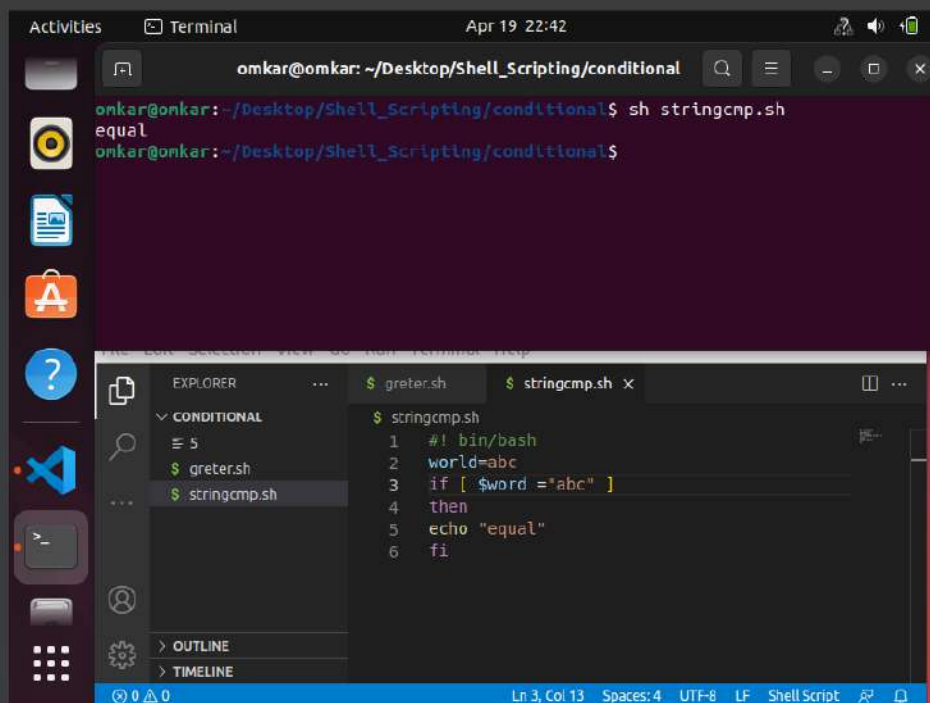
The status bar at the bottom of the file editor indicates "Ln 3, Col 17", "5spaces: 4", "UTF-8", "LF", and "Shell Script".



The screenshot shows a terminal window titled "Terminal" with the date and time "Apr 19 22:32". The terminal prompt is "omkar@omkar: ~/Desktop/Shell_Scripting/conditional". The user has executed the command "sh greter.sh", and the output is "It is greter than 5". Below the terminal window, there is a file explorer window showing the contents of the "conditional" directory. The file explorer has a sidebar with "EXPLORER", "CONDITIONAL", "OUTLINE", and "TIMELINE". The "CONDITIONAL" directory is expanded, showing a file named "greter.sh". The file explorer also shows the contents of the "greter.sh" file, which is a shell script.

```
omkar@omkar: ~/Desktop/Shell_Scripting/conditional
omkar@omkar:~/Desktop/Shell_Scripting/conditional$ sh greter.sh
It is greter than 5
omkar@omkar:~/Desktop/Shell_Scripting/conditional$
```

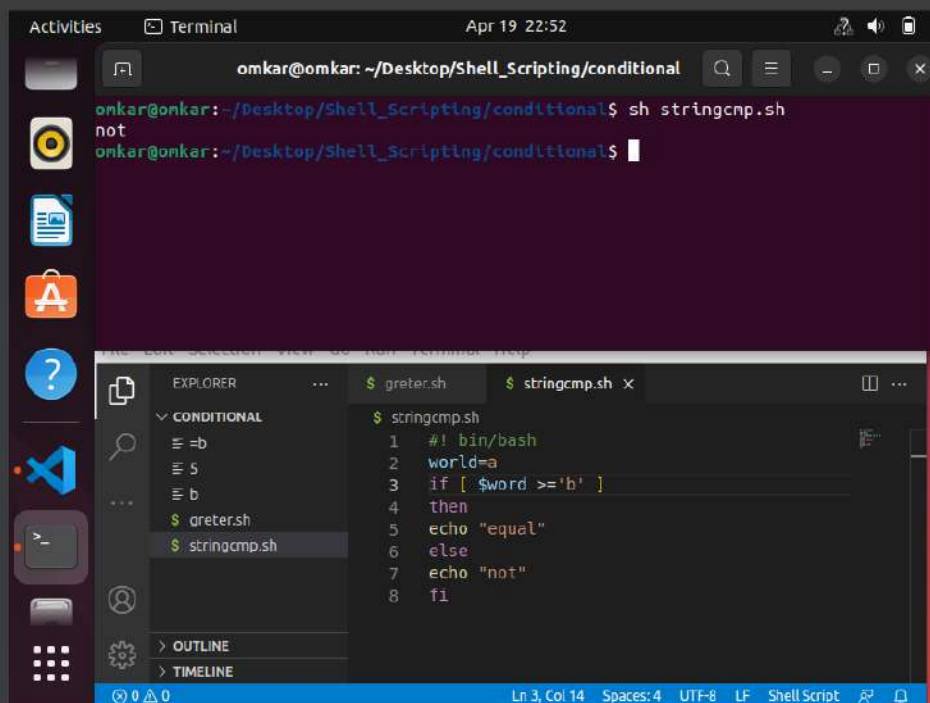
```
$ greter.sh
$ greter.sh
1  #!/bin/bash
2  count=10
3  if [ $count -gt 5 ]
4  then
5  echo "It is greter than 5"
6  fi
7
8
```



The screenshot shows a terminal window titled "Terminal" with the date and time "Apr 19 22:42". The terminal prompt is "omkar@omkar: ~/Desktop/Shell_Scripting/conditional". The user has executed the command "sh stringcmp.sh", which has produced the output "equal". Below the terminal window, there is a code editor window showing the contents of "stringcmp.sh". The code is as follows:

```
$ stringcmp.sh
1  #! bin/bash
2  world=abc
3  if [ $word = "abc" ]
4  then
5  echo "equal"
6  fi
```

The code editor window also shows a file explorer on the left with a tree view containing "CONDITIONAL", "greter.sh", and "stringcmp.sh". The status bar at the bottom of the code editor indicates "Ln 3, Col 13", "Spaces: 4", "UTF-8", "LF", and "Shell Script".



The screenshot shows a terminal window titled "Terminal" with the date and time "Apr 19 22:52". The terminal prompt is "omkar@omkar: ~/Desktop/Shell_Scripting/conditional". The user has executed the command "sh stringcmp.sh", and the output is "not". Below the terminal window, there is a file explorer view showing the contents of the "conditional" directory. The files listed are "greter.sh" and "stringcmp.sh". The "stringcmp.sh" file is selected, and its contents are displayed in the right pane. The script is a shell script that compares two words, "world" and "a", and prints "equal" if the first word is greater than or equal to the second, and "not" otherwise.

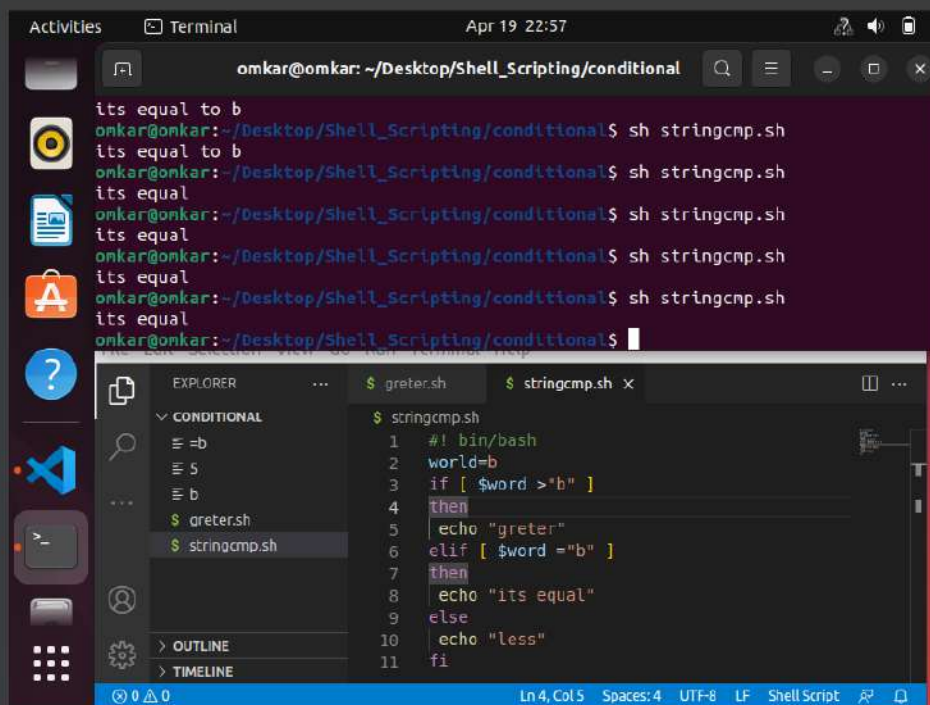
```
omkar@omkar: ~/Desktop/Shell_Scripting/conditional
omkar@omkar:~/Desktop/Shell_Scripting/conditional$ sh stringcmp.sh
not
omkar@omkar:~/Desktop/Shell_Scripting/conditional$
```

EXPLORER

- CONDITIONAL
 - ≡ =b
 - ≡ 5
 - ≡ b
 - ...\$ greter.sh
 - \$ stringcmp.sh
- OUTLINE
- TIMELINE

```
$ stringcmp.sh
1  #! bin/bash
2  world=a
3  if [ $word >= 'b' ]
4  then
5  echo "equal"
6  else
7  echo "not"
8  fi
```

Ln 3, Col 14 Spaces: 4 UTF-8 LF Shell Script



```
omkar@omkar: ~/Desktop/Shell_Scripting/conditional
its equal to b
omkar@omkar:~/Desktop/Shell_Scripting/conditional$ sh stringcmp.sh
its equal to b
omkar@omkar:~/Desktop/Shell_Scripting/conditional$ sh stringcmp.sh
its equal
omkar@omkar:~/Desktop/Shell_Scripting/conditional$ sh stringcmp.sh
its equal
omkar@omkar:~/Desktop/Shell_Scripting/conditional$ sh stringcmp.sh
its equal
omkar@omkar:~/Desktop/Shell_Scripting/conditional$ sh stringcmp.sh
its equal
omkar@omkar:~/Desktop/Shell_Scripting/conditional$
```

EXPLORER

- CONDITIONAL
 - =b
 - =5
 - =b
 - greter.sh
 - stringcmp.sh
- OUTLINE
- TIMELINE

```
$ stringcmp.sh
1  #!/ bin/bash
2  world=b
3  if [ $word >"b" ]
4  then
5  echo "greter"
6  elif [ $word ="b" ]
7  then
8  echo "its equal"
9  else
10 echo "less"
11 fi
```

Ln 4, Col 5 Spaces: 4 UTF-8 LF Shell Script

The screenshot shows a terminal window titled "omkar@omkar: ~/Desktop/Shell_Scripting/conditional" with a search bar and window controls. The terminal output shows two runs of a script named "age.sh". In the first run, the user enters "15" and the script outputs "you are not". In the second run, the user enters "25" and the script outputs "you are eligible to vote". Below the terminal, a code editor is open with a file explorer on the left showing a directory structure with files "greter.sh", "stringcmp.sh", "age.sh", "b", and "greter.sh". The main editor area shows the content of "age.sh" with 8 lines of code. The status bar at the bottom indicates "Ln 8, Col 3", "Spaces: 4", "UTF-8", "LF", and "Shell Script".

```
omkar@omkar: ~/Desktop/Shell_Scripting/conditional
omkar@omkar:~/Desktop/Shell_Scripting/conditional$ sh age.sh
enter your age
15
you are not
omkar@omkar:~/Desktop/Shell_Scripting/conditional$ sh age.sh
enter your age
25
you are eligible to vote
omkar@omkar:~/Desktop/Shell_Scripting/conditional$
```

EXPLORER

- CONDITIONAL
 - = b
 - = 5
 - age.sh
 - b
 - greter.sh
 - stringcmp.sh
- OUTLINE
- TIMELINE

```
$ age.sh
1 echo enter your age
2 read age
3 if [ $age -gt 18 ]
4 then
5 echo you are eligible to vote
6 else
7 echo you are not
8 fi
```

Ln 8, Col 3 Spaces: 4 UTF-8 LF Shell Script

Activities Terminal Apr 20 01:05

omkar@omkar: ~/Desktop/Shell_Scripting/conditional

```
15
evenNum.sh: 10: Syntax error: end of file unexpected (expecting "fi")
omkar@omkar:~/Desktop/Shell_Scripting/conditional$ sh evenNum.sh
enter the number
10
even
omkar@omkar:~/Desktop/Shell_Scripting/conditional$ sh evenNum.sh
enter the number
15
odd
omkar@omkar:~/Desktop/Shell_Scripting/conditional$
```

Restricted Mode is intended for safe code browsing. Trust this folder to enable all features. [Manage](#) [Learn More](#) X

EXPLORER ... \$ evenNum.sh X \$ highest.sh

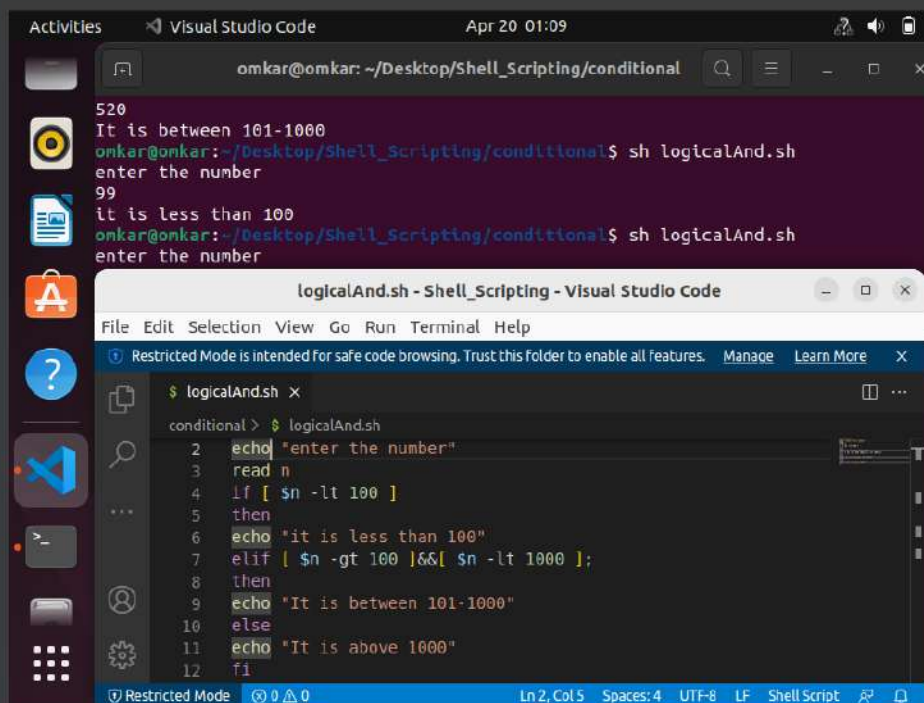
SHELL_SCRIPTING

- > Batsman
- > conditional
 - \$ age.sh
 - \$ evenNum.sh
 - \$ greter.sh
 - \$ highest.sh
 - \$ n.sh
 - \$ stringcmp.sh
- > OUTLINE
- > TIMELINE

conditional > \$ evenNum.sh

```
1  #!/bin/bash
2
3  echo "enter the number"
4  read num
5  if [ `expr $num % 2` -eq 0 ]
6  then
7  echo "even"
8  else
9  echo "odd"
10 fi
11
```

Ln 10, Col 3 Spaces: 4 UTF-8 LF Shell Script



Activities Visual Studio Code Apr 20 01:09

omkar@omkar: ~/Desktop/Shell_Scripting/conditional

```
520
It is between 101-1000
omkar@omkar:~/Desktop/Shell_Scripting/conditional$ sh logicalAnd.sh
enter the number
99
it is less than 100
omkar@omkar:~/Desktop/Shell_Scripting/conditional$ sh logicalAnd.sh
enter the number
```

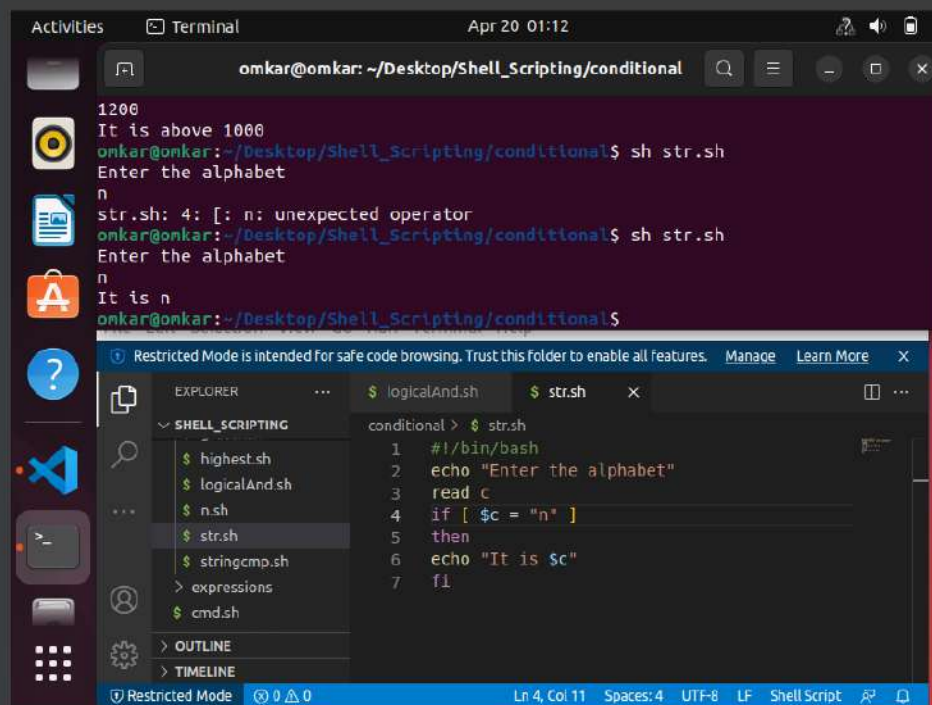
logicalAnd.sh - Shell_Scripting - Visual Studio Code

File Edit Selection View Go Run Terminal Help

Restricted Mode is intended for safe code browsing. Trust this folder to enable all features. [Manage](#) [Learn More](#) X

```
$ logicalAnd.sh X
conditional > $ logicalAnd.sh
2 echo "enter the number"
3 read n
4 if [ $n -lt 100 ]
5 then
6 echo "it is less than 100"
7 elif [ $n -gt 100 ]&&[ $n -lt 1000 ];
8 then
9 echo "It is between 101-1000"
10 else
11 echo "It is above 1000"
12 fi
```

Ln 2, Col 5 Spaces: 4 UTF-8 LF Shell Script



```
1200  
It is above 1000  
omkar@omkar:~/Desktop/Shell_Scripting/conditional$ sh str.sh  
Enter the alphabet  
n  
str.sh: 4: [: n: unexpected operator  
omkar@omkar:~/Desktop/Shell_Scripting/conditional$ sh str.sh  
Enter the alphabet  
n  
It is n  
omkar@omkar:~/Desktop/Shell_Scripting/conditional$
```

Restricted Mode is intended for safe code browsing. Trust this folder to enable all features. [Manage](#) [Learn More](#) [X](#)

EXPLORER

- ✓ SHELL_SCRIPTING
 - \$ highest.sh
 - \$ logicalAnd.sh
 - \$ n.sh
 - \$ str.sh
 - \$ stringcmp.sh
 - > expressions
 - \$ cmd.sh
- > OUTLINE
- > TIMELINE

conditional > \$ str.sh

```
1  #!/bin/bash  
2  echo "Enter the alphabet"  
3  read c  
4  if [ $c = "n" ]  
5  then  
6  echo "It is $c"  
7  fi
```

Ln 4, Col 11 Spaces: 4 UTF-8 LF Shell Script

Activities Terminal Apr 20 01:16

omkar@omkar: ~/Desktop/Shell_Scripting/conditional

```
File exists
-----
not
omkar@omkar:~/Desktop/Shell_Scripting/conditional$ ls
age.sh  evenNum.sh  greter.sh  logicalAnd.sh  stringcnp.sh
Batsman file-f.sh  highest.sh  n.sh          str.sh
omkar@omkar:~/Desktop/Shell_Scripting/conditional$ sh file-f.sh
File exists
-----
directory exists
omkar@omkar:~/Desktop/Shell_Scripting/conditional$
```

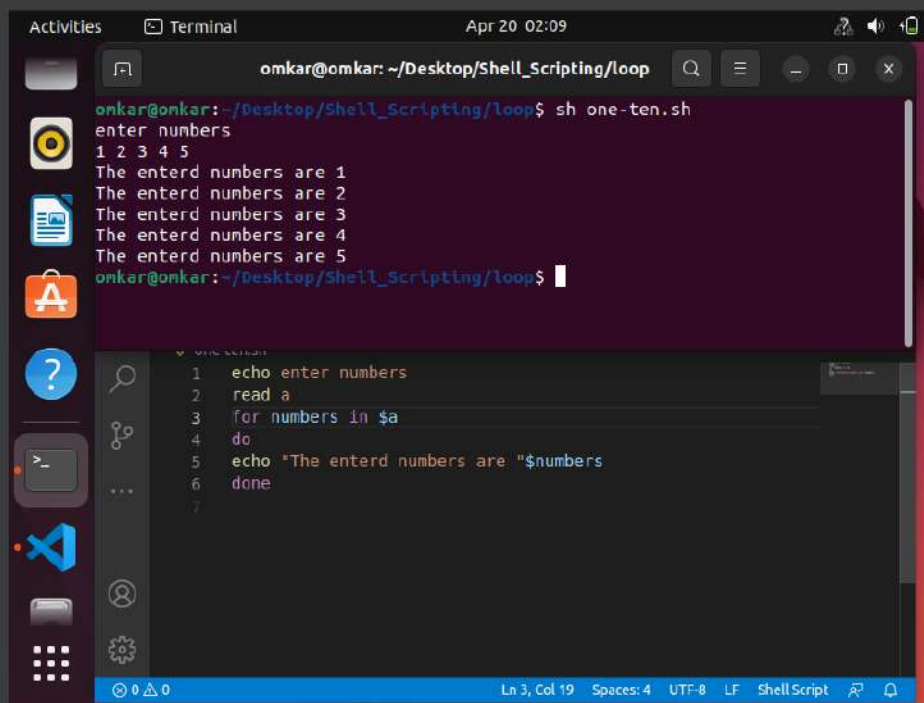
Restricted Mode is intended for safe code browsing. Trust this folder to enable all features. [Manage](#) [Learn More](#) X

EXPLORER ... \$ logicalAnd.sh \$ str.sh \$ file-f.sh X ...

SHELL_SCRIPTING conditional > \$ file-f.sh

```
5 echo "File exists"
6 fi
7 echo "-----"
8 Dir=Batsman
9 if [ -d $Dir ]
10 then
11 echo "directory exists"
12 else
13 echo "not"
14 fi
```

Ln 8, Col 9 Spaces: 4 UTF-8 LF Shell Script



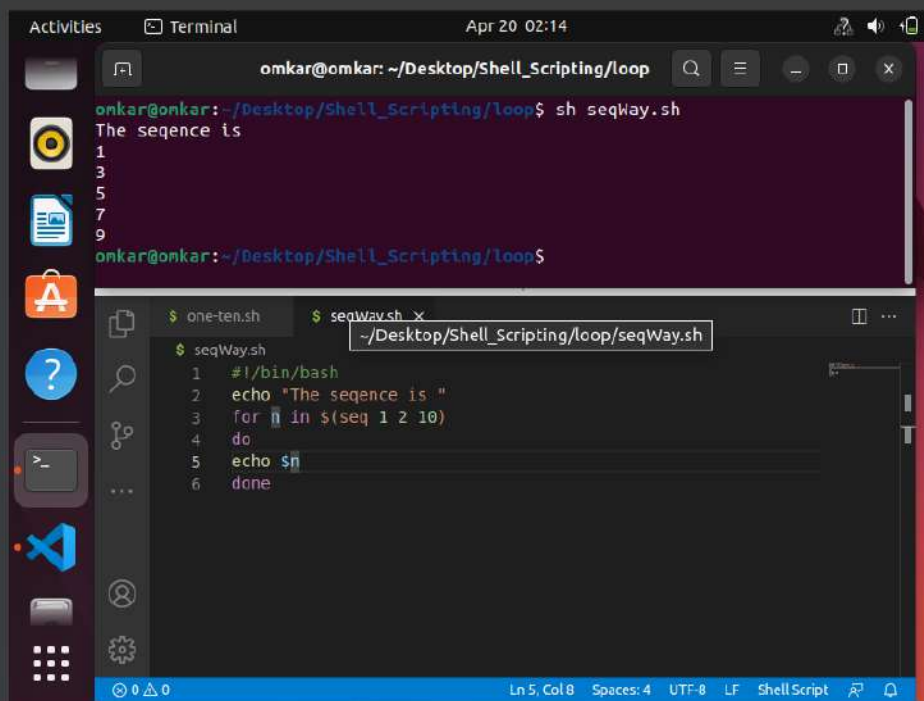
The screenshot shows a terminal window titled "Terminal" with a date and time of "Apr 20 02:09". The terminal is running a shell script named "one-ten.sh". The script prompts the user to "enter numbers" and then displays the numbers 1 through 5. The output of the script is "The enterd numbers are 1", "The enterd numbers are 2", "The enterd numbers are 3", "The enterd numbers are 4", and "The enterd numbers are 5". The terminal prompt is "onkar@onkar: ~/Desktop/Shell_Scripting/loop\$".

```
onkar@onkar: ~/Desktop/Shell_Scripting/loop$ sh one-ten.sh
enter numbers
1 2 3 4 5
The enterd numbers are 1
The enterd numbers are 2
The enterd numbers are 3
The enterd numbers are 4
The enterd numbers are 5
onkar@onkar: ~/Desktop/Shell_Scripting/loop$
```

The terminal window also shows the source code of the script "one-ten.sh" in a separate pane:

```
1 echo enter numbers
2 read a
3 for numbers in $a
4 do
5 echo "The enterd numbers are "$numbers
6 done
7
```

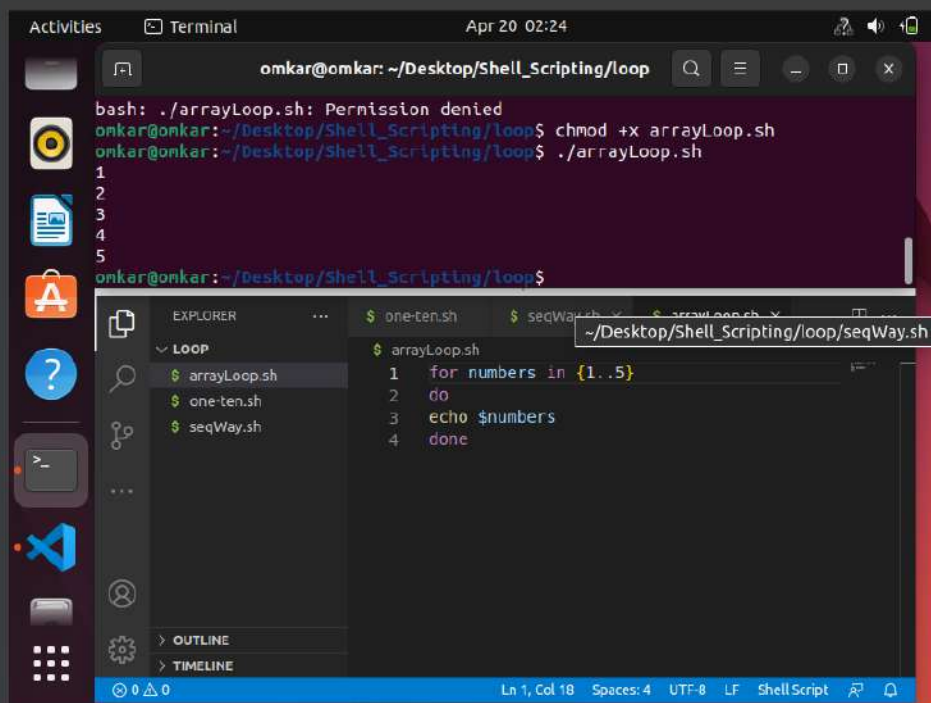
The status bar at the bottom of the terminal window indicates "Ln 3, Col 19 Spaces: 4 UTF-8 LF Shell Script".



The screenshot shows a terminal window titled "Terminal" with a timestamp of "Apr 20 02:14". The terminal is running a shell script named "seqWay.sh". The prompt is "onkar@onkar: ~/Desktop/Shell_Scripting/loop". The script output is "The sequence is" followed by the numbers 1, 3, 5, 7, and 9. Below the terminal window, there is a code editor showing the contents of "seqWay.sh". The code is as follows:

```
1 #!/bin/bash
2 echo "The sequence is "
3 for i in $(seq 1 2 10)
4 do
5 echo $i
6 done
```

The status bar at the bottom of the code editor shows "Ln 5, Col 8", "Spaces: 4", "UTF-8", "LF", "Shell Script", and a bell icon.



The screenshot shows a terminal window titled "omkar@omkar: ~/Desktop/Shell_Scripting/loop" with a search bar and window controls. The terminal output shows a permission error for `./arrayLoop.sh`, followed by a successful `chmod +x` command and the execution of `./arrayLoop.sh`, which prints the numbers 1 through 5. Below the terminal, a file explorer view shows a directory named "LOOP" containing `arrayLoop.sh`, `one-ten.sh`, and `seqWay.sh`. The `seqWay.sh` file is selected, and its contents are displayed in the editor: a `for` loop over `{1..5}` that echoes each number. The status bar at the bottom indicates the cursor is at line 1, column 18, with 4 spaces, in UTF-8 encoding, LF line endings, and Shell Script mode.

```
omkar@omkar: ~/Desktop/Shell_Scripting/loop
bash: ./arrayLoop.sh: Permission denied
omkar@omkar:~/Desktop/Shell_Scripting/loop$ chmod +x arrayLoop.sh
omkar@omkar:~/Desktop/Shell_Scripting/loop$ ./arrayLoop.sh
1
2
3
4
5
omkar@omkar:~/Desktop/Shell_Scripting/loop$
```

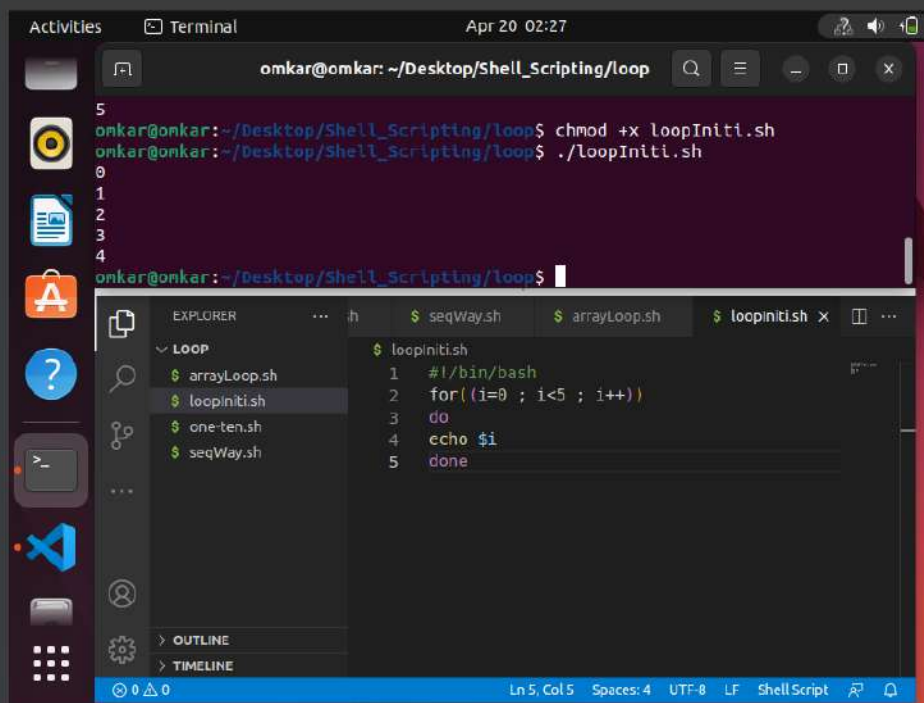
EXPLORER

- LOOP
 - arrayLoop.sh
 - one-ten.sh
 - seqWay.sh

OUTLINE

TIMELINE

Ln 1, Col 18 Spaces: 4 UTF-8 LF Shell Script



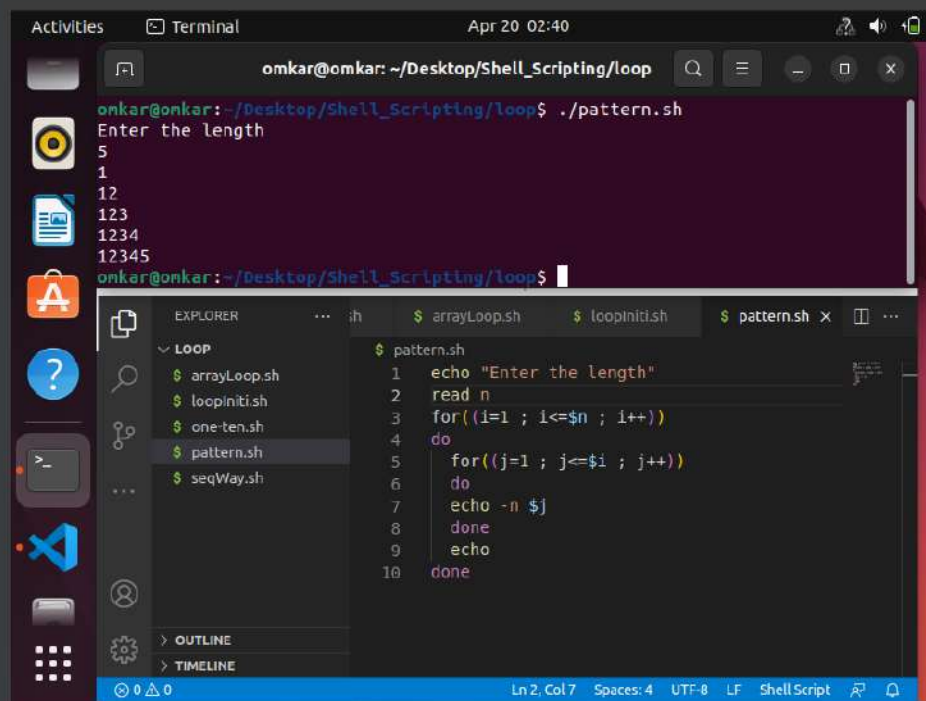
The screenshot shows a terminal window titled "omkar@omkar: ~/Desktop/Shell_Scripting/loop" with a search bar and window controls. The terminal output is as follows:

```
5
onkar@omkar:~/Desktop/Shell_Scripting/loop$ chmod +x loopIniti.sh
onkar@omkar:~/Desktop/Shell_Scripting/loop$ ./loopIniti.sh
0
1
2
3
4
onkar@omkar:~/Desktop/Shell_Scripting/loop$
```

Below the terminal, a code editor is open with the file "loopinitish" selected. The code in the editor is:

```
$ loopinitish
1 #!/bin/bash
2 for((i=0 ; i<5 ; i++))
3 do
4 echo $i
5 done
```

The code editor's interface includes an "EXPLORER" sidebar on the left showing a file tree with "LOOP" expanded, containing "arrayLoop.sh", "loopinitish", "one-ten.sh", and "seqWay.sh". The "OUTLINE" and "TIMELINE" sections are collapsed. The status bar at the bottom indicates "Ln 5, Col 5", "Spaces: 4", "UTF-8", "LF", and "Shell Script".



The screenshot shows a terminal window titled "omkar@omkar: ~/Desktop/Shell_Scripting/loop". The terminal output shows the execution of a script named "pattern.sh". The script prompts the user to "Enter the length" and the user enters "5". The script then prints the numbers 1, 12, 123, 1234, and 12345, each on a new line. The terminal window is part of an IDE with a sidebar showing a file explorer with a folder named "LOOP" containing files "arrayLoop.sh", "loopinit.sh", "one-ten.sh", "pattern.sh", and "seqWay.sh". The "pattern.sh" file is selected, and its contents are displayed in the main editor. The status bar at the bottom indicates "Ln 2, Col 7 Spaces: 4 UTF-8 LF Shell Script".

```
omkar@omkar: ~/Desktop/Shell_Scripting/loop$ ./pattern.sh
Enter the length
5
1
12
123
1234
12345
omkar@omkar: ~/Desktop/Shell_Scripting/loop$
```

```
$ pattern.sh
1 echo "Enter the length"
2 read n
3 for((i=1 ; i<=$n ; i++))
4 do
5     for((j=1 ; j<=$i ; j++))
6     do
7         echo -n $j
8     done
9     echo
10 done
```