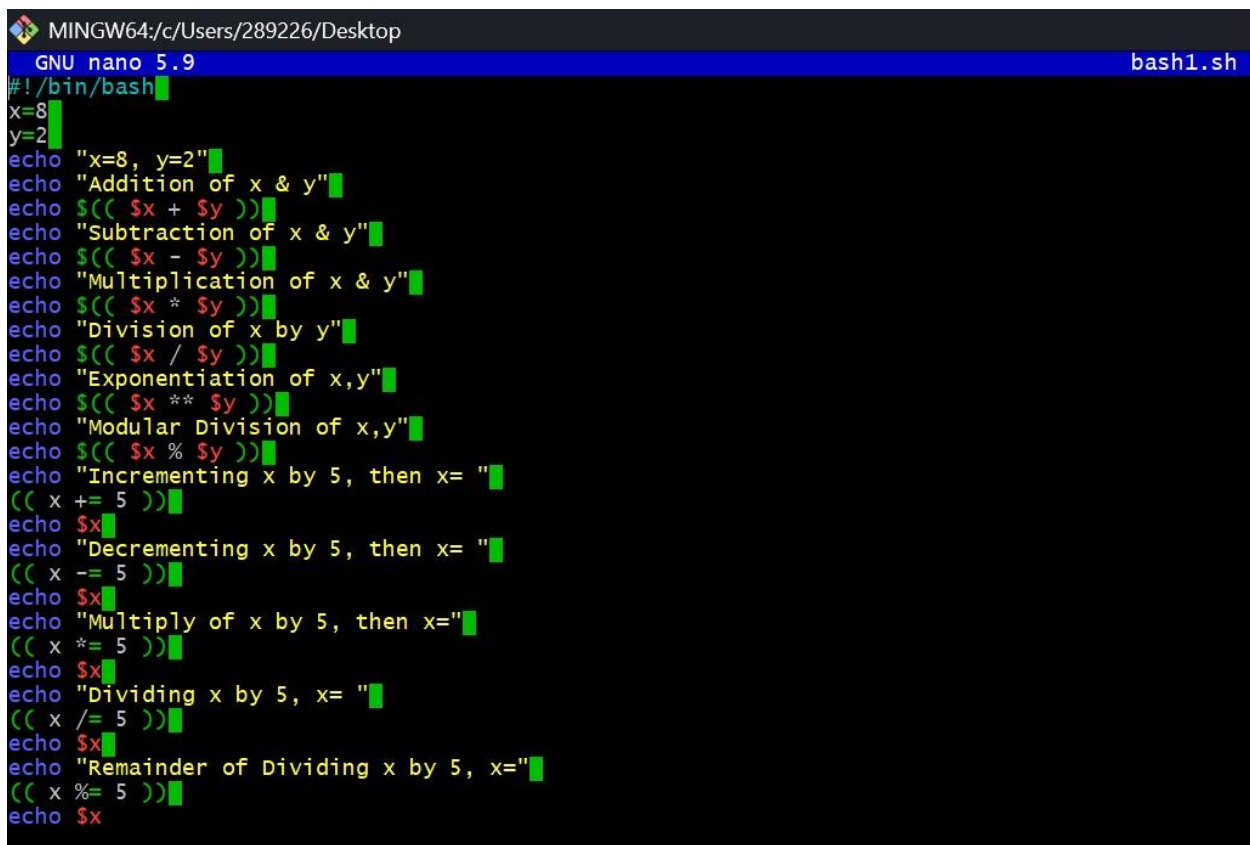


# BASH SCRIPT ASSIGNMENT

## Script 1: Arithmetic Operations



The screenshot shows a terminal window with the title bar "MINGW64:/c/Users/289226/Desktop". The terminal is running GNU nano 5.9, editing a file named "bash1.sh". The script content is as follows:

```
#!/bin/bash
x=8
y=2
echo "x=8, y=2"
echo "Addition of x & y"
echo $(( $x + $y ))
echo "Subtraction of x & y"
echo $(( $x - $y ))
echo "Multiplication of x & y"
echo $(( $x * $y ))
echo "Division of x by y"
echo $(( $x / $y ))
echo "Exponentiation of x,y"
echo $(( $x ** $y ))
echo "Modular Division of x,y"
echo $(( $x % $y ))
echo "Incrementing x by 5, then x= "
(( x += 5 ))
echo $x
echo "Decrementing x by 5, then x= "
(( x -= 5 ))
echo $x
echo "Multiply of x by 5, then x="
(( x *= 5 ))
echo $x
echo "Dividing x by 5, x= "
(( x /= 5 ))
echo $x
echo "Remainder of Dividing x by 5, x="
(( x %= 5 ))
echo $x
```

PRATHEEK U B

289226

```
USTR+289226@J4DR353 MINGW64 ~/Desktop  
$ chmod 777 bash1.sh
```

```
USTR+289226@J4DR353 MINGW64 ~/Desktop
```

```
$ . bash1.sh
```

```
x=8, y=2
```

```
Addition of x & y
```

```
10
```

```
Subtraction of x & y
```

```
6
```

```
Multiplication of x & y
```

```
16
```

```
Division of x by y
```

```
4
```

```
Exponentiation of x,y
```

```
64
```

```
Modular Division of x,y
```

```
0
```

```
Incrementing x by 5, then x=
```

```
13
```

```
Decrementing x by 5, then x=
```

```
8
```

```
Multiply of x by 5, then x=
```

```
40
```

```
Dividing x by 5, x=
```

```
8
```

```
Remainder of Dividing x by 5, x=
```

```
3
```

## Script 2: Let Command

```
MINGW64:/c/Users/289226/Desktop
GNU nano 5.9 bash2.sh
#!/bin/bash
x=10
y=6
z=0
echo "Addition"
let "z = $(( x + y ))"
echo "z= $z"

echo "Subtraction"
let "z = $((x - y ))"
echo "z= $z"

echo "Multiplication"
let "z = $(( x * y ))"
echo "z = $z"

echo "Division"
let "z = $(( x / y ))"
echo "z = $z"

echo "Exponentiation"
let "z = $(( x ** y ))"
echo "z = $z"

echo "Modular Division"
let "z = $(( x % y ))"
echo "z = $z"
let "x += 5"
echo "Incrementing x by 5, then x= "
echo $x

let "x -= 5"
echo "Decrementing x by 5, then x= "
echo $x

let "x *=5"
echo "Multiply of x by 5, then x="
echo $x

let "x /= 5"
echo "Dividing x by 5, x= "
echo $x

let "x %= 5"
echo "Remainder of Dividing x by 5, x="
echo $x

^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify    ^_ Go To Line
```

PRATHEEK U B

289226

```
USTR+289226@J4DR353 MINGW64 ~/Desktop
$ nano bash2.sh

USTR+289226@J4DR353 MINGW64 ~/Desktop
$ . bash2.sh
Addition
z= 16
Substraction
z= 4
Multiplication
z = 60
Division
z = 1
Exponentiation
z = 1000000
Modular Division
z = 4
Incrementing x by 5, then x=
15
Decrementing x by 5, then x=
10
Multiply of x by 5, then x=
50
Dividing x by 5, x=
10
Remainder of Dividing x by 5, x=
0

USTR+289226@J4DR353 MINGW64 ~/Desktop
$
```

### Script 3: Expr Command

```
MINGW64:/c/Users/289226/Desktop
GNU nano 5.9 bash3.sh
#!/bin/bash
#Basic arithmetic using expr

echo "a=10, b=3"
echo "c is the value of addition c=a+b"
a=10
b=3
echo "c= `expr $a + $b`"

USTR+289226@J4DR353 MINGW64 ~/Desktop
$ nano bash3.sh

USTR+289226@J4DR353 MINGW64 ~/Desktop
$ chmod 777 bash3.sh

USTR+289226@J4DR353 MINGW64 ~/Desktop
$ . bash3.sh
a=10, b=3
c is the value of addition c=a+b
c= 13

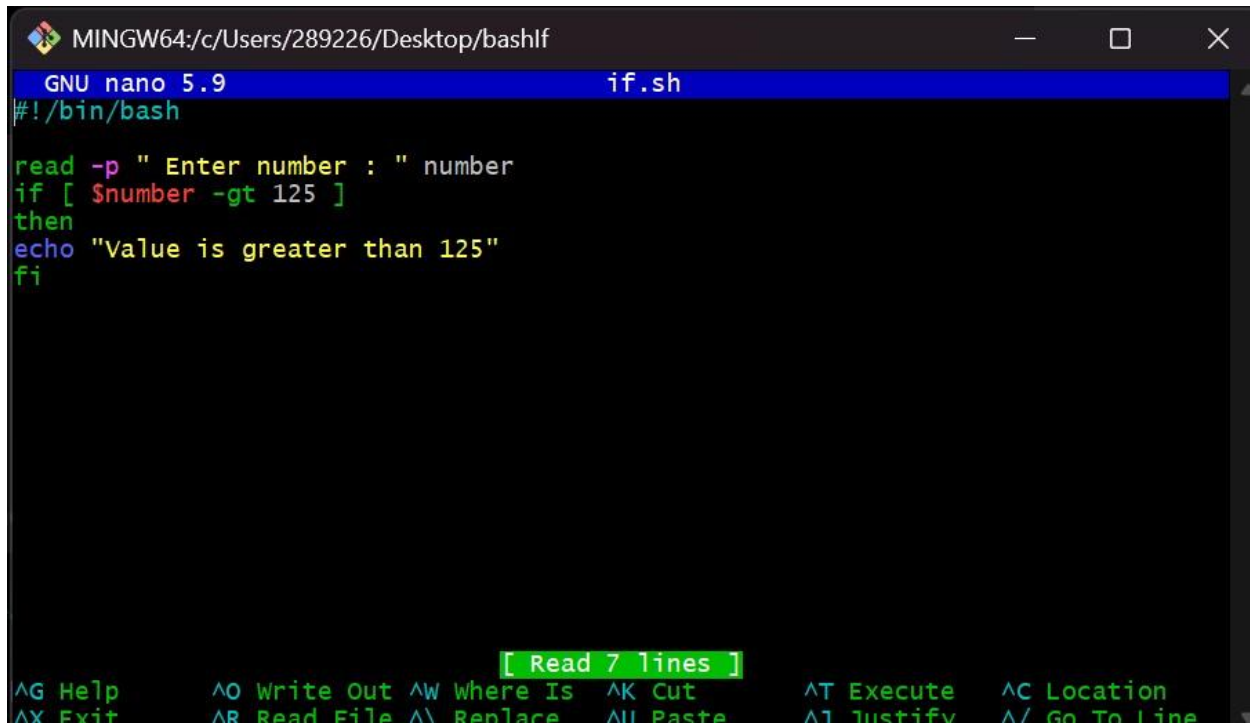
USTR+289226@J4DR353 MINGW64 ~/Desktop
$ |
```

PRATHEEK U B

289226

## Bash IF Statement

### Script 1: If Statement



A screenshot of a MINGW64 terminal window with the title bar "MINGW64:/c/Users/289226/Desktop/bashIf". The window contains the GNU nano 5.9 editor editing a file named "if.sh". The script content is as follows:

```
#!/bin/bash  
  
read -p " Enter number : " number  
if [ $number -gt 125 ]  
then  
echo "Value is greater than 125"  
fi
```

At the bottom of the editor, a status bar shows "[ Read 7 lines ]" and a list of keyboard shortcuts: ^G Help, ^O Write Out, ^W Where Is, ^K Cut, ^T Execute, ^C Location, ^X Exit, ^R Read File, ^\ Replace, ^U Paste, ^J Justify, ^\_ Go To Line.

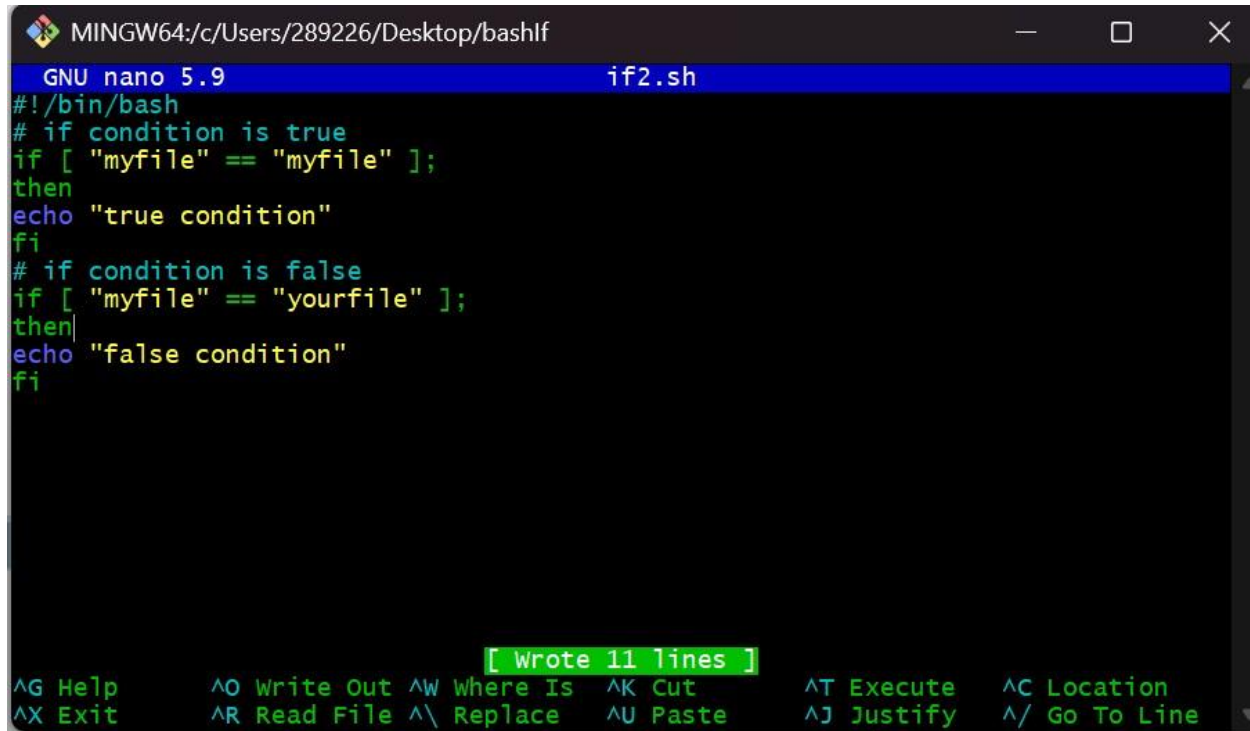


A screenshot of a terminal window showing the execution of the script. The prompt is "USTR+289226@J4DR353 MINGW64 ~/Desktop/bashIf". The user enters ". if.sh". The prompt changes to "Enter number :". The user enters "299". The output is "Value is greater than 125".

```
USTR+289226@J4DR353 MINGW64 ~/Desktop/bashIf  
$ . if.sh  
Enter number : 299  
Value is greater than 125
```

PRATHEEK U B  
289226

## Script 2: If Statement



The screenshot shows a terminal window titled "MINGW64:/c/Users/289226/Desktop/bashIf". Inside, the GNU nano 5.9 editor is open, editing a file named "if2.sh". The script content is as follows:

```
#!/bin/bash
# if condition is true
if [ "myfile" == "myfile" ];
then
echo "true condition"
fi
# if condition is false
if [ "myfile" == "yourfile" ];
then
echo "false condition"
fi
```

At the bottom of the editor, a status bar indicates "[ Wrote 11 lines ]". Below the editor, a help menu is visible with the following options:

|         |              |             |          |            |               |
|---------|--------------|-------------|----------|------------|---------------|
| ^G Help | ^O Write Out | ^W Where Is | ^K Cut   | ^T Execute | ^C Location   |
| ^X Exit | ^R Read File | ^\ Replace  | ^U Paste | ^J Justify | ^_ Go To Line |

```
USTR+289226@J4DR353 MINGW64 ~/Desktop/bashIf
$ . if2.sh
true condition

USTR+289226@J4DR353 MINGW64 ~/Desktop/bashIf
$
```

## Script 3:

```
GNU nano 5.9                                if3.sh                                Modified
#!/bin/bash

#if condition (greater than) is true
if [ 10 -gt 3 ];
then
echo "10 is greater than 3."
fi

#if condition (greater than) is false
if [ 3 -gt 10 ];
then
echo "3 is not greater than 10."
fi

#if condition (lesser than) is true
if [ 3 -lt 10 ];
then
echo "3 is less than 10."
fi

#if condition (lesser than) is false
if [ 10 -lt 3 ];
then
echo "10 is not less than 3."
fi

#if condition (equal to) is true
if [ 10 -eq 10 ];
then
echo "10 is equal to 10."
fi

#if condition (equal to) is false
if [ 10 -eq 9 ];
then
echo "10 is not equal to 9"
fi
```

```
USTR+289226@J4DR353 MINGW64 ~/Desktop/bashIf
$ . if3.sh
10 is greater than 3.
3 is less than 10.
10 is equal to 10.

USTR+289226@J4DR353 MINGW64 ~/Desktop/bashIf
$ |
```



PRATHEEK U B

289226

## Script 4: AND

```
GNU nano 5.9                                if4.sh
#!/bin/bash

# TRUE && TRUE
if [ 8 -gt 6 ] && [ 10 -eq 10 ];
then
echo "Conditions are true"
fi

# TRUE && FALSE
if [ "mylife" == "mylife" ] && [ 3 -gt 10 ];
then
echo "Conditions are false"
fi
```

```
USTR+289226@J4DR353 MINGW64 ~/Desktop/bashIf
$ . and.sh
Conditions are true
```

## Script 5: OR

```
GNU nano 5.9                                if5.sh                                Modified
#!/bin/bash

# TRUE || FALSE
if [ 8 -gt 7 ] || [ 10 -eq 3 ];
then
echo " Condition is true. "
fi

# FALSE || FALSE
if [ "mylife" == "yourlife" ] || [ 3 -gt 10 ];
then
echo " Condition is false. "
fi
```

```
USTR+289226@J4DR353 MINGW64 ~/Desktop/bashIf
$ . or.sh
Condition is true.
```



## Script 6: AND and OR

```
GNU nano 5.9 AndOr.sh
#!/bin/bash

# TRUE && FALSE || FALSE || TRUE
if [[ 10 -eq 10 && 5 -gt 4 || 3 -eq 4 || 3 -lt 6 ]];
then
echo "Condition is true."
fi

# TRUE && FALSE || FALSE
if [[ 8 -eq 8 && 8 -gt 10 || 9 -lt 5 ]];
then
echo "Condition is false"
fi
```

```
USTR+289226@J4DR353 MINGW64 ~/Desktop/bashIf
$ . AndOr.sh
Condition is true.
```

## Script 7: Nested If

```
GNU nano 5.9 nestedIf.sh Modified
#!/bin/bash
#Nested if statement

if [ $1 -gt 50 ]
then
echo "Number is greater than 50."

if (( $1 % 2 == 0 ))
then
echo "and it is an even number."
fi
fi
```

```
USTR+289226@J4DR353 MINGW64 ~/Desktop/bashIf
$ . nestedIf.sh
Number is greater than 50.
and it is an even number.
```

## Bash IF ELSE

### Script 1: If Else

```
GNU nano 5.9 ifElse.sh Modified
#!/bin/bash
#when the condition is true
if [ 10 -gt 3 ];
then
echo "10 is greater than 3."
else
echo "10 is not greater than 3."
fi
#when the condition is false
if [ 3 -gt 10 ];
then
echo "3 is greater than 10."
else
echo "3 is not greater than 10."
fi
```

```
USTR+289226@J4DR353 MINGW64 ~/Desktop/bashIfElse
$ . ifElse.sh
10 is greater than 3.
3 is not greater than 10.

USTR+289226@J4DR353 MINGW64 ~/Desktop/bashIfElse
$
```

## Script 2: If else

```
GNU nano 5.9 ifElse2.sh Modified
#!/bin/bash
# When condition is true
# TRUE && FALSE || FALSE || TRUE
if [[ 10 -gt 9 && 10 == 9 || 2 -lt 1 || 25 -gt 20 ]];
then
echo "Given condition is true."
else
echo "Given condition is false."
fi
# When condition is false
#TRUE && FALSE || FALSE || TRUE
if [[ 10 -gt 9 && 10 == 8 || 3 -gt 4 || 8 -gt 8 ]];
then
echo "Given condition is true."
else
echo "Given condition is not true."
fi

USTR+289226@J4DR353 MINGW64 ~/Desktop/bashIfElse
$ . ifElse2.sh
Given condition is true.
Given condition is not true.

USTR+289226@J4DR353 MINGW64 ~/Desktop/bashIfElse
$ |
```

## Script 3: If else single line

```
GNU nano 5.9 ifElse3.sh
#!/bin/bash
read -p "Enter a value:" value
if [ $value -gt 9 ]; then echo "The value you typed is greater than 9."; else echo "The value you typed is not greater than 9."; fi

USTR+289226@J4DR353 MINGW64 ~/Desktop/bashIfElse
$ . ifElse3.sh
Enter a value:25
The value you typed is greater than 9.
```

## Script 4: Nested If Else

```
GNU nano 5.9 nestedIfElse.sh
#!/bin/bash
read -p "Enter a value:" value
if [ $value -gt 9 ];
then
if [ $value -lt 11 ];
then
echo "$value>9, $value<11"
else
echo "The value you typed is greater than 9."
fi
else echo "The value you typed is not greater than 9."
fi
```

```
USTR+289226@J4DR353 MINGW64 ~/Desktop/bashIfElse
$ . nestedIfElse.sh
Enter a value:10
10>9, 10<11

USTR+289226@J4DR353 MINGW64 ~/Desktop/bashIfElse
$ |
```

## Script 5: ELIF

```
GNU nano 5.9 elif.sh
#!/bin/bash

read -p "Enter a number of quantity:" num

if [ $num -gt 100 ];
then
echo "Eligible for 10% discount"
elif [ $num -lt 100 ];
then
echo "Eligible for 5% discount"
else
echo "Lucky Draw Winner"
echo "Eligible to get the item for free"
fi
```

```
USTR+289226@J4DR353 MINGW64 ~/Desktop/bashIfElse
$ . elif.sh
Enter a number of quantity:110
Eligible for 10% discount
```

## PRATHEEK U B

### 289226

Script 6:

```
GNU nano 5.9                                     elif2.sh
#!/bin/bash

read -p "Enter a number of quantity:" num

if [ $num -gt 200 ];
then
echo "Eligible for 20% discount"

elif [[ $num == 200 || $num == 100 ]];
then
echo "Lucky Draw Winner"
echo "Eligible to get the item for free"

elif [[ $num -gt 100 && $num -lt 200 ]];
then
echo "Eligible for 10% discount"

elif [ $num -lt 100 ];
then
echo "No discount"
fi
```

```
USTR+289226@J4DR353 MINGW64 ~/Desktop/bashIfElse
$ . elif2.sh
Enter a number of quantity:100
Lucky Draw Winner
Eligible to get the item for free
```