Arithmetic Operations


```
bhargava@23372bc849b55ff:~$ nano arithmetic_operations
bhargava@23372bc849b55ff:~$ chmod +x arithmetic_operations
bhargava@23372bc849b55ff:~$ cat arithmetic_operations
#!/bin/bash
x=20
y=10
echo "x=20,y=10"
echo "Addition of x and y"
echo $(( $x + $y ))
echo "Subtraction of x and y"
echo $(( $x - $y ))
echo "Multiplication of x and y"
echo $(( $x * $y ))
echo "Division of x and 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 "Multiplying x by 5,then x="
(( x*=5 ))
echo $x
echo "Dividing x by 5,then x="
(( x/=5 ))
echo $x
echo "REminder of Dividing x by 5,then x="
(( x%=5 ))
echo $x
bhargava@23372bc849b55ff:~$ ./arithmetic_operations
x=20,y=10
Addition of x and y
30
Subtraction of x and y
10
Multiplication of x and y
```



```
echo $x
echo "Multiplying x by 5,then x="
(( x*=5 ))
echo $x
echo "Dividing x by 5,then x="
(( x/=5 ))
echo $x
echo "REminder of Dividing x by 5,then x="
(( x%=5 ))
echo $x
bhargava@23372bc849b55ff:~$ ./arithmetic_operations
x=20,y=10
Addition of x and y
30
Subtraction of x and y
10
Multiplication of x and y
200
Division of x and y
2
Exponentiation of x,y
10240000000000
Modular Division of x,y
0
Incrementing x by 5,then x=
25
Decrementing x by 5,then x=
20
Multiplying x by 5,then x=
100
Dividing x by 5,then x=
20
REminder of Dividing x by 5,then x=
0
bhargava@23372bc849b55ff:~$
```

```
bhargava@23372bc849b55ff:~$ nano operations
bhargava@23372bc849b55ff:~$ chmod +x operations
bhargava@23372bc849b55ff:~$ cat operations
#!/bin/bash
x=10
y=5
echo "x=10,y=5"
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=$(( x / y ))"
echo "Exponentiation"
let "z=$(( x ** y ))"
echo "z=$z"
echo "Modular Division"
let "z=$(( x % y ))"
echo "z=$z"
echo "Incrementing x by 5,then x="
let "x+=5"
echo $x
echo "Decrementing x by 5,then x="
let "x-=5"
echo $x
echo "Multiplying x by 5,then x="
let "x*=5"
echo $x
echo "Dividing x by 5,then x="
let "x/=5"
echo $x
echo "REminder of Dividing x by 5,then x="
let "x%=5"
echo $x
bhargava@23372bc849b55ff:~$ ./operations
x=10,y=5
```

```
let "x-=5"
echo $x
echo "Multiplying x by 5,then x="
let "x*=5"
echo $x
echo "Dividing x by 5,then x="
let "x/=5"
echo $x
echo "REminder of Dividing x by 5,then x="
let "x%=5"
echo $x
bhargava@23372bc849b55ff:~$ ./operations
x=10,y=5
Addition
z=15
Subtraction
z=5
Multiplication
z=50
Division
Exponentiation
z=100000
Modular Division
z=0
Incrementing x by 5,then x=
15
Decrementing x by 5,then x=
10
Multiplying x by 5,then x=
50
Dividing x by 5,then x=
10
REminder of Dividing x by 5,then x=
0
bhargava@23372bc849b55ff:~$
```
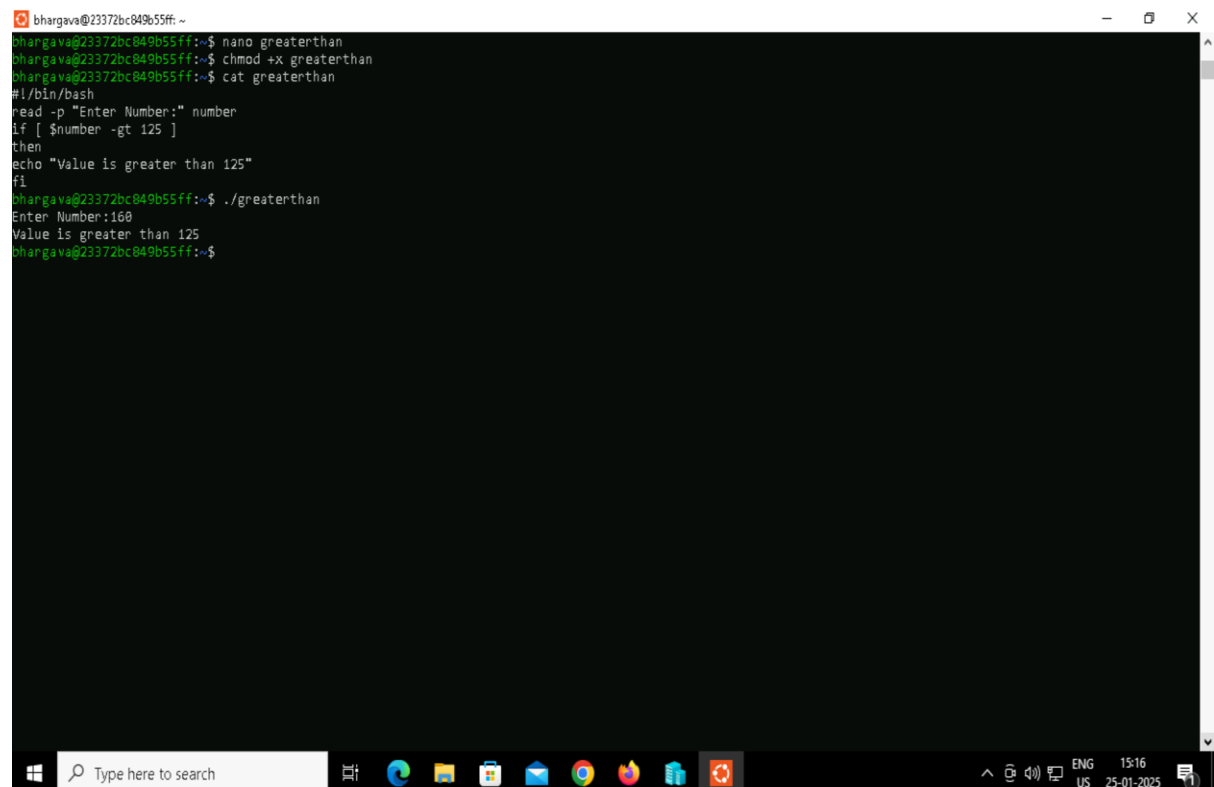
Backticks

```
bhargava@23372bc849b55ff:~$ nano backticks
bhargava@23372bc849b55ff:~$ chmod +x backticks
bhargava@23372bc849b55ff:~$ cat backticks
#!/bin/bash
echo "a=10,b=3"
echo "c is the value of addition c=a+b"
a=10
b=3
echo "c=`expr $a + $b`"

bhargava@23372bc849b55ff:~$ ./backticks
a=10,b=3
c is the value of addition c=a+b
c=13
bhargava@23372bc849b55ff:~$
```

Greater than Example

```
bhargava@23372bc849b55ff:~$ nano greaterthan
bhargava@23372bc849b55ff:~$ chmod +x greaterthan
bhargava@23372bc849b55ff:~$ cat greaterthan
#!/bin/bash
read -p "Enter Number:" number
if [ $number -gt 125 ]
then
echo "Value is greater than 125"
fi
bhargava@23372bc849b55ff:~$ ./greaterthan
Enter Number:160
Value is greater than 125
bhargava@23372bc849b55ff:~$
```

Comparing two strings example


```
bhargava@23372bc849b55ff:~$ nano comparing_two_strings
bhargava@23372bc849b55ff:~$ chmod +x comparing_two_strings
bhargava@23372bc849b55ff:~$ cat comparing_two_strings
#!/bin/bash
if [ "myfile" == "myfile" ];
then
echo "true condition"
fi

if [ "myfile" == "yourfile" ];
then
echo "false condition"
fi
bhargava@23372bc849b55ff:~$ ./comparing_two_strings
true condition
bhargava@23372bc849b55ff:~$
```

Numbers Comparison


```
bhargava@23372bc849b55ff:~$ nano numbers_comparision
bhargava@23372bc849b55ff:~$ chmod +x numbers_comparision
bhargava@23372bc849b55ff:~$ cat numbers_comparision
#!/bin/bash
if [ 10 -gt 3 ];
then
echo "10 is greater than 3"
fi
if [ 3 -gt 10 ];
then
echo "3 is not greater than 10"
fi

if [ 3 -lt 10 ];
then
echo "3 is less than 10"
fi

if [ 10 -lt 3 ];
then
echo "10 is not less than 3"
fi

if [ 10 -eq 10 ];
then
echo "10 is equal to 10"
fi

if [ 10 -eq 9 ];
then
echo "10 is not equal to 9"
fi
bhargava@23372bc849b55ff:~$ ./numbers_comparision
10 is greater than 3
3 is less than 10
10 is equal to 10
bhargava@23372bc849b55ff:~$
```

And Operator


```
bhargava@23372bc849b55ff:~$ nano And_operator
bhargava@23372bc849b55ff:~$ chmod +x And_operator
bhargava@23372bc849b55ff:~$ cat And_operator
#!/bin/bash
if [ 8 -gt 6 ] && [ 15 -eq 15 ];
then
echo "Conditions are True"
fi

if [ "mylife" == "mylife" ] && [ 3 -gt 10 ];
then
echo "Conditions are False"
fi
bhargava@23372bc849b55ff:~$ ./And_operator
Conditions are True
bhargava@23372bc849b55ff:~$
```

OR Operator


```
bhargava@23372bc849b55ff:~$ nano or_operator
bhargava@23372bc849b55ff:~$ chmod +x or_operator
bhargava@23372bc849b55ff:~$ cat or_operator
#!/bin/bash
if [ 8 -gt 6 ] || [ 15 -eq 15 ];
then
echo "Conditions are True"
fi

if [ "mylife" == "mylife" ] || [ 3 -gt 10 ];
then
echo "Conditions are False"
fi
bhargava@23372bc849b55ff:~$ ./or_operator
Conditions are True
Conditions are False
bhargava@23372bc849b55ff:~$
```

And_OR Operators

```
bhargava@23372bc849b55ff:~$ nano and_or
bhargava@23372bc849b55ff:~$ chmod +x and_or
bhargava@23372bc849b55ff:~$ cat and_or
#!/bin/bash
if [[ 8 -gt 6 && 10 -eq 10 || 3 -eq 4 || 3 -lt 6 ]];
then
echo "Conditions are True"
fi

if [[ 8 -eq 8 && 3 -gt 10 || 9 -lt 5 ]];
then
echo "Conditions are False"
fi
bhargava@23372bc849b55ff:~$ ./and_or
Conditions are True
bhargava@23372bc849b55ff:~$
```

Nested_if

```
bhargava@23372bc849b55ff:~$ nano nested_if
bhargava@23372bc849b55ff:~$ chmod +x nested_if
bhargava@23372bc849b55ff:~$ cat nested_if
#!/bin/bash
echo "enter a number"
read a
if [ $a -gt 50 ]
then
echo "Number is greater than 50"

if (( $a % 2 == 0 ))
then
echo "and it an even number"
fi
fi
bhargava@23372bc849b55ff:~$ ./nested_if
enter a number
64
Number is greater than 50
and it an even number
bhargava@23372bc849b55ff:~$
```

Bash if else example1


```
bhargava@23372bc849b55ff:~$ nano one.bash
bhargava@23372bc849b55ff:~$ chmod +x one.bash
bhargava@23372bc849b55ff:~$ cat one.bash
#!/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
bhargava@23372bc849b55ff:~$ ./one.bash
10 is greater than 3.
3 is not greater than 10.
bhargava@23372bc849b55ff:~$
```

Bash if else example2


```
bhargava@23372bc849b55ff:~$ nano two.bash
bhargava@23372bc849b55ff:~$ chmod +x two.bash
bhargava@23372bc849b55ff:~$ cat two.bash
#!/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
bhargava@23372bc849b55ff:~$ ./two.bash
Given condition is true.
Given condition is not true.
bhargava@23372bc849b55ff:~$
```

Bash if else example3


```
bhargava@23372bc849b55ff:~$ nano three.bash
bhargava@23372bc849b55ff:~$ chmod +x three.bash
bhargava@23372bc849b55ff:~$ cat three.bash
#!/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

bhargava@23372bc849b55ff:~$ ./three.bash
Enter a value:18
The value you typed is greater than 9.
bhargava@23372bc849b55ff:~$
```

Bash if else example4


```
bhargava@23372bc849b55ff:~$ nano four.bash
bhargava@23372bc849b55ff:~$ chmod +x four.bash
bhargava@23372bc849b55ff:~$ cat four.bash
#!/bin/bash
read -p "Enter a value: " value
if [ $value -gt 9 ]; then
    echo "The value you typed is greater than 9"
    if [ $value -lt 11 ]; then
        echo "$value > 9 and $value < 11"
    else
        echo "$value is greater than or equal to 11"
    fi
else
    echo "The value you typed is not greater than 9."
fi

bhargava@23372bc849b55ff:~$ ./four.bash
Enter a value: 10
The value you typed is greater than 9
10 > 9 and 10 < 11
bhargava@23372bc849b55ff:~$
```

Bash else if example1

```
bhargava@23372bc849b55ff:~$ nano five.bash
bhargava@23372bc849b55ff:~$ chmod +x five.bash
bhargava@23372bc849b55ff:~$ cat five.bash
#!/bin/bash
read -p "Enter a number of quantity:" num
if [ $num -gt 100 ];
then
echo "Eligible for 10% discount"
elif [ $num -lt 100 ];
then
echo "Eligible for 5% discount"
else
echo "Lucky Draw Winner"
echo "Eligible to get the item for free"
fi
bhargava@23372bc849b55ff:~$ ./five.bash
Enter a number of quantity:100
Lucky Draw Winner
Eligible to get the item for free
bhargava@23372bc849b55ff:~$
```

Bash else if example2

```
bhargava@23372bc849b55ff:~$ nano six.bash
bhargava@23372bc849b55ff:~$ chmod +x six.bash
bhargava@23372bc849b55ff:~$ cat six.bash
#!/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 [ $num -lt 100 ];
then
echo "No discount"
fi
bhargava@23372bc849b55ff:~$ ./six.bash
Enter a number of quantity:210
Eligible for 20% discount
bhargava@23372bc849b55ff:~$
```