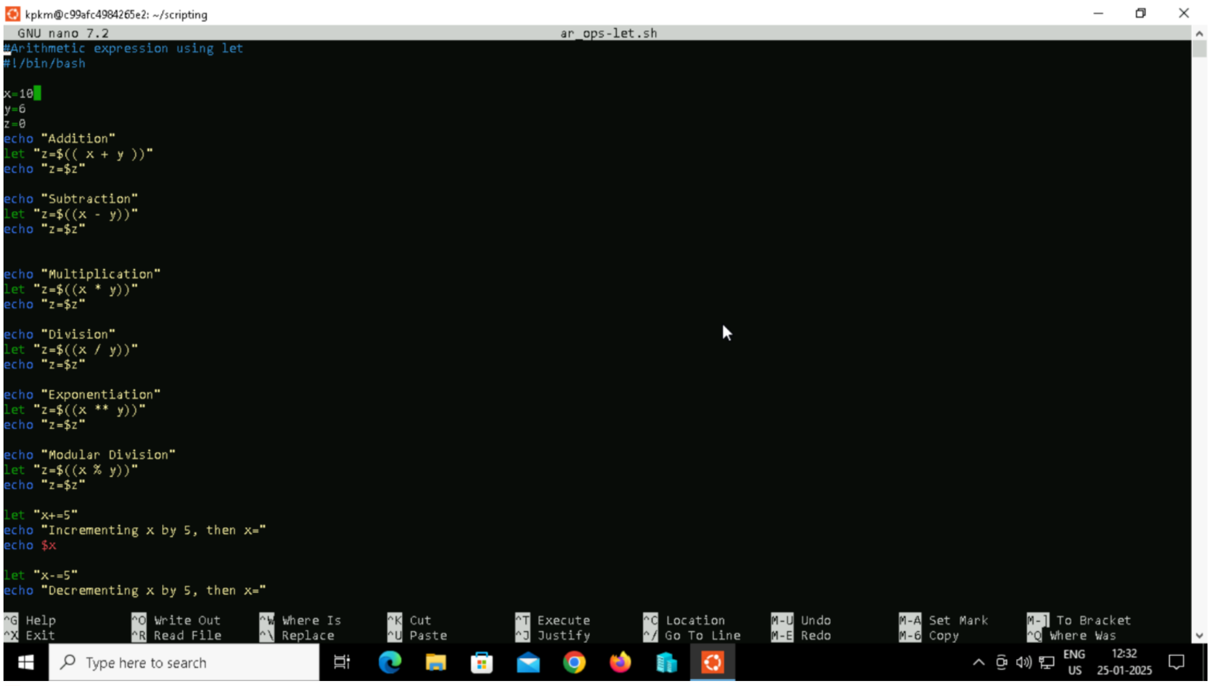
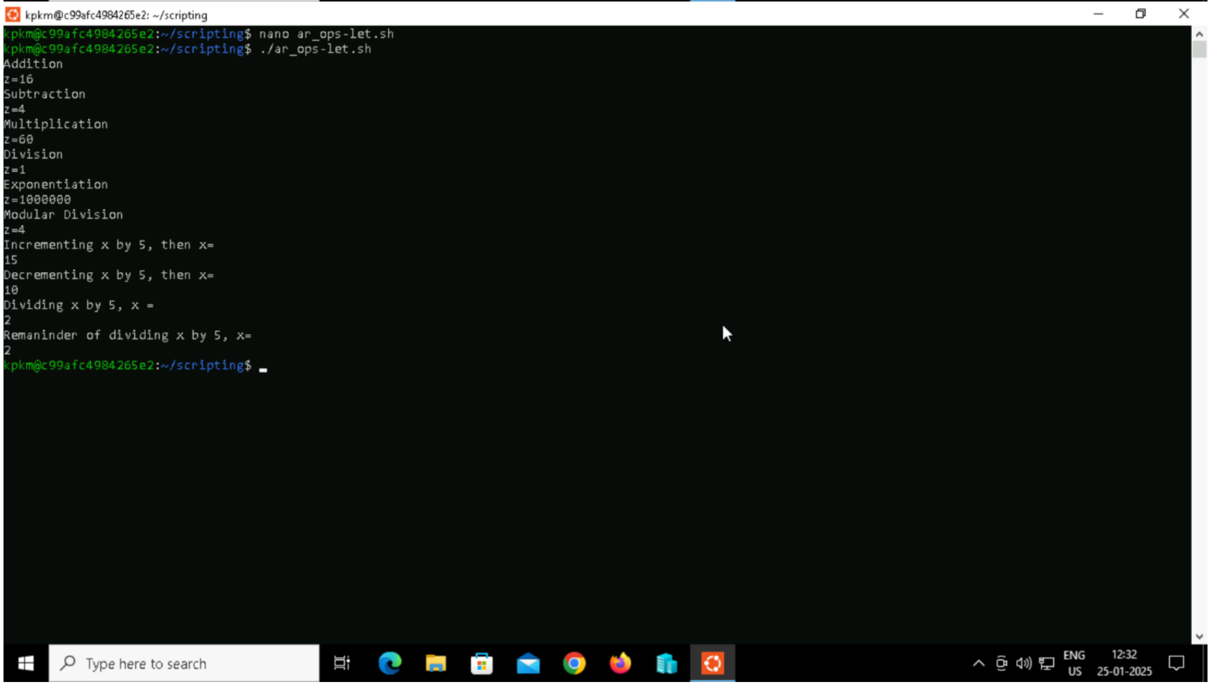
Arithmetic Operations

Q. Arithmetic expressions using let

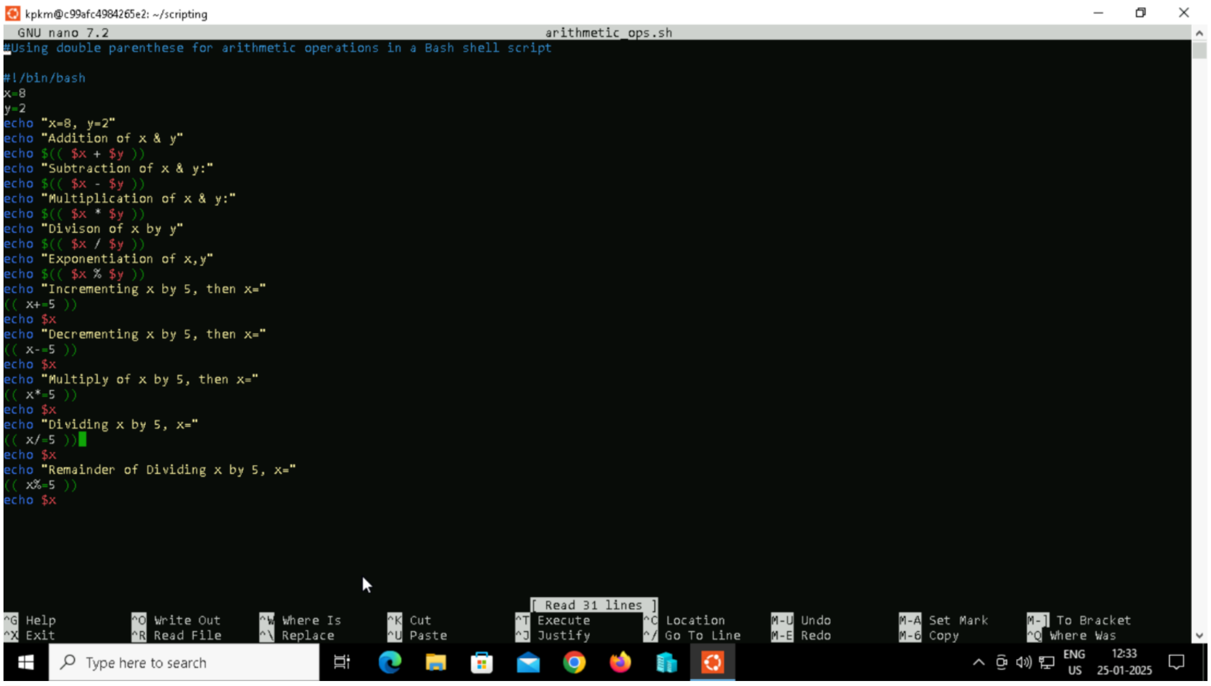


```
kpkm@c99afc4984265e2: ~/scripting
  GNU nano 7.2                                      ar_ops-let.sh
#Arithmetic expression using let
#!/bin/bash

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="
^G Help      ^O Write Out  ^W Where Is   ^K Cut      ^T Execute   ^C Location    M-U Undo   M-A Set Mark   M-] To Bracket
^X Exit      ^R Read File  ^\ Replace    ^U Paste    ^J Justify   ^/ Go To Line  M-E Redo   M-6 Copy       ^Q Where Was
```

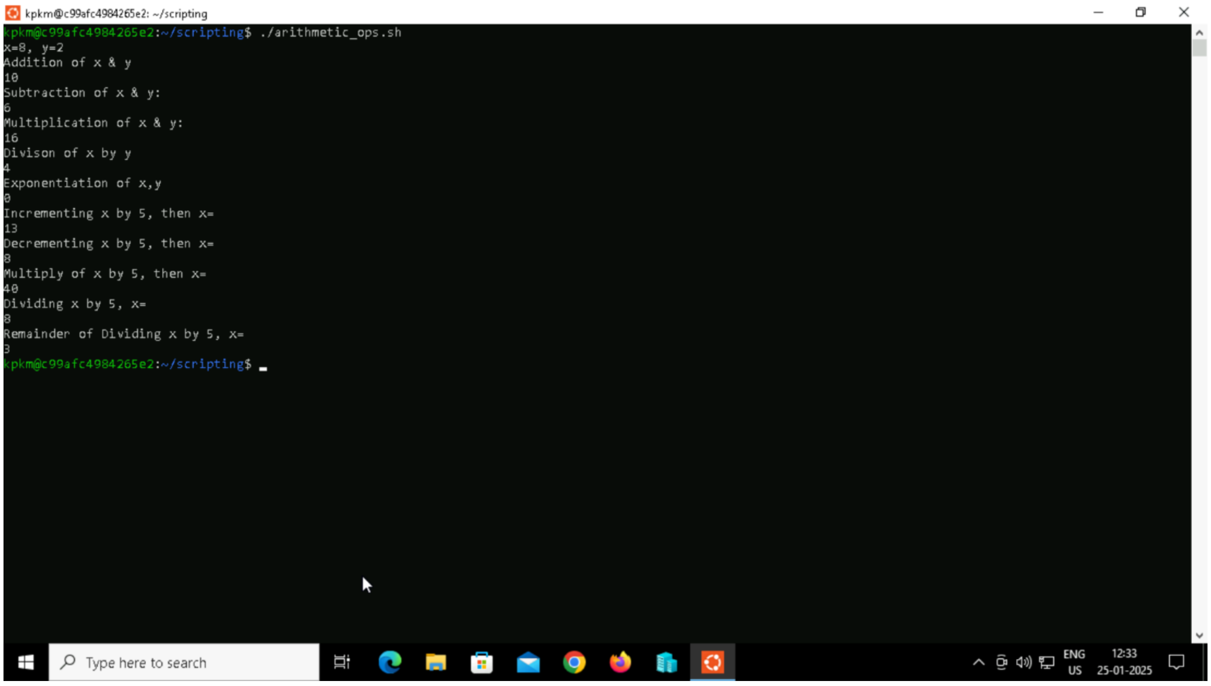


```
kpkm@c99afc4984265e2: ~/scripting
kpkm@c99afc4984265e2:~/scripting$ nano ar_ops-let.sh
kpkm@c99afc4984265e2:~/scripting$ ./ar_ops-let.sh
Addition
z=16
Subtraction
z=4
Multiplication
z=60
Division
z=1
Exponentiation
z=1000000
Modular Division
z=4
Incrementing x by 5, then x=
15
Decrementing x by 5, then x=
10
Dividing x by 5, x =
2
Remainder of dividing x by 5, x=
2
kpkm@c99afc4984265e2:~/scripting$
```

Q. Arithmetic operations using double parenthese

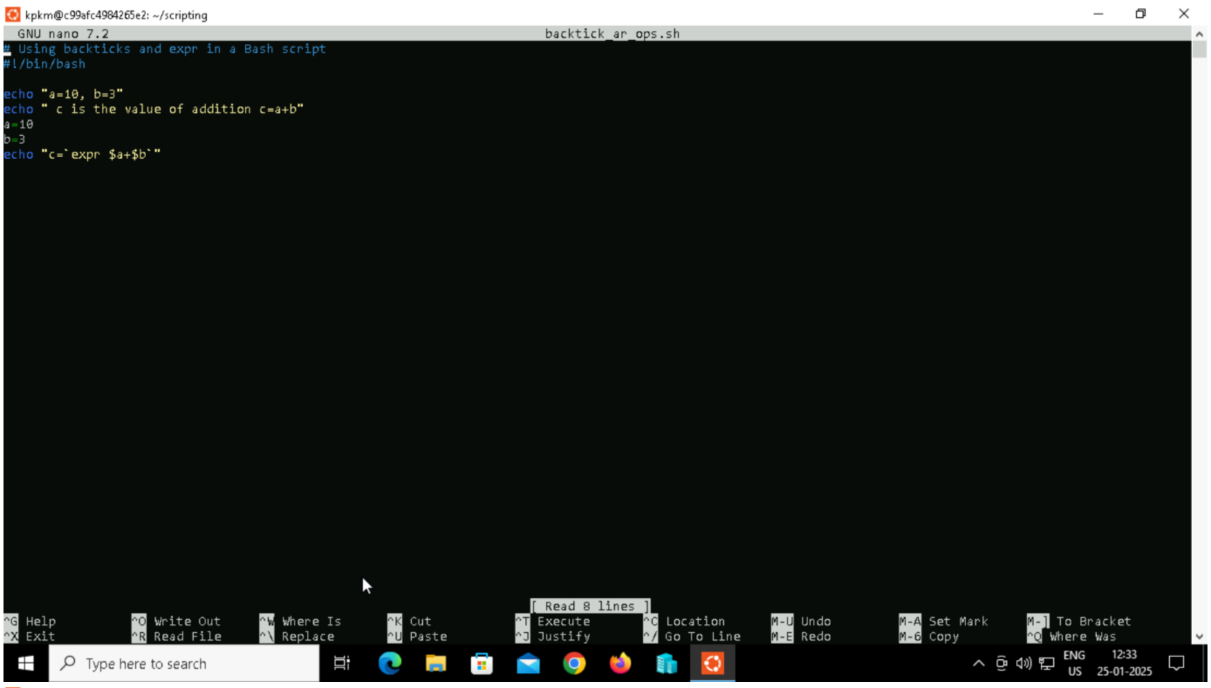


```
kpkm@c99afc4984265e2: ~/scripting
  GNU nano 7.2                                   arithmetic_ops.sh
#Using double parenthese for arithmetic operations in a Bash shell script

#!/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 "Divison of x by y"
echo $(( $x / $y ))
echo "Exponentiation 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

                                             [ Read 31 lines ]
^G Help        ^O Write Out   ^W Where Is    ^K Cut         ^T Execute     ^C Location    M-U Undo       M-A Set Mark   M-] To Bracket
^X Exit        ^R Read File   ^\ Replace     ^U Paste       ^J Justify     ^/ Go To Line  M-E Redo       M-6 Copy       ^Q Where Was
```

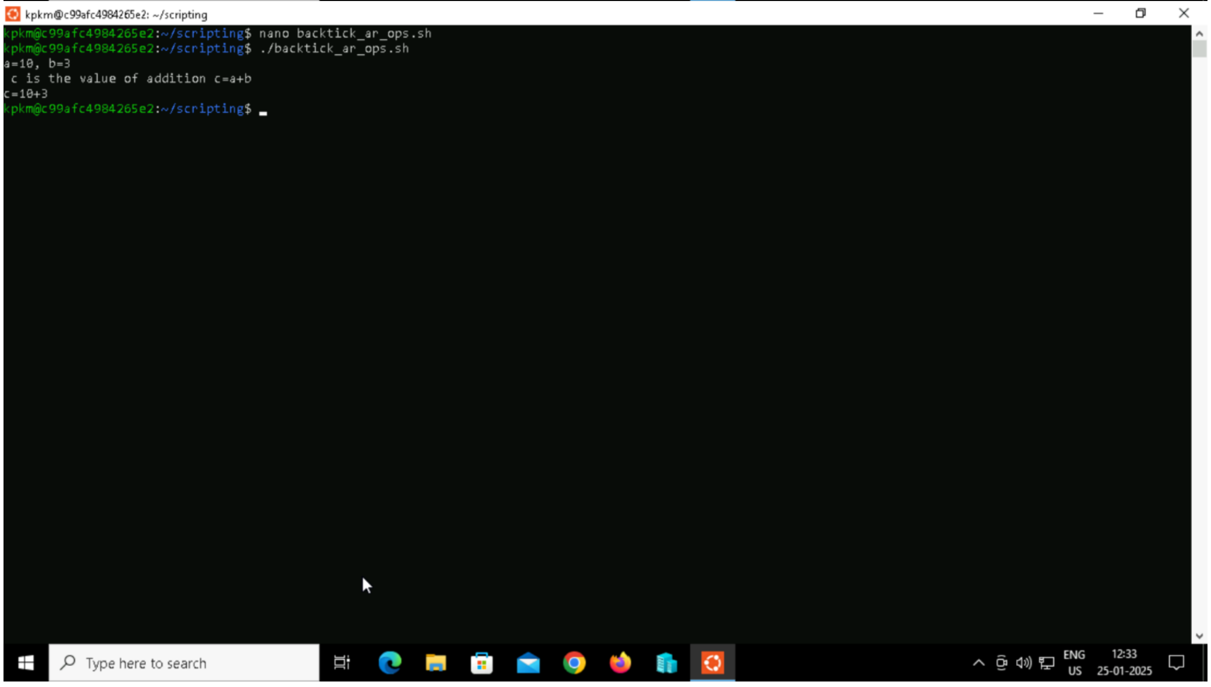


```
kpkm@c99afc4984265e2: ~/scripting
kpkm@c99afc4984265e2:~/scripting$ ./arithmetic_ops.sh
x=8, y=2
Addition of x & y
10
Subtraction of x & y:
6
Multiplication of x & y:
16
Divison of x by y
4
Exponentiation of x,y
0
Incrementing x by 5, then x=
13
Decrementing x by 5, then x=
8
Multiply of x by 5, then x=
40
Dividing x by 5, x=
8
Remainder of Dividing x by 5, x=
3
kpkm@c99afc4984265e2:~/scripting$ _
```

Q. Using backticks and expr in Bash script



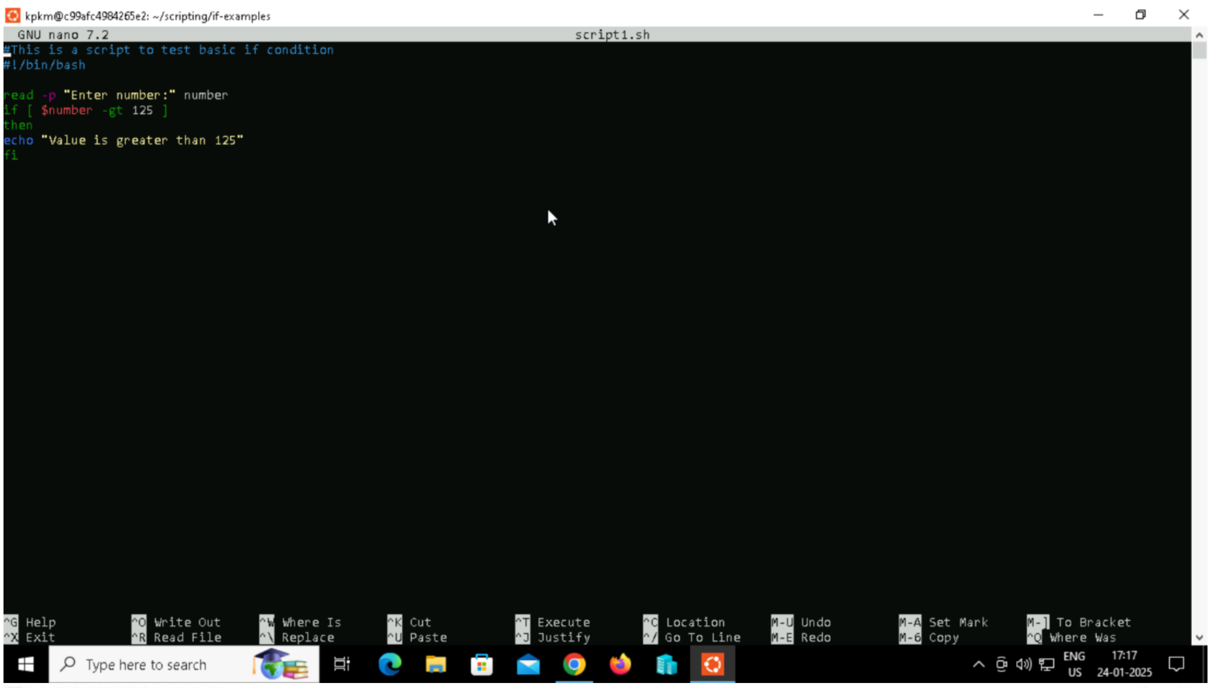
```
# Using backticks and expr in a Bash script
#!/bin/bash

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



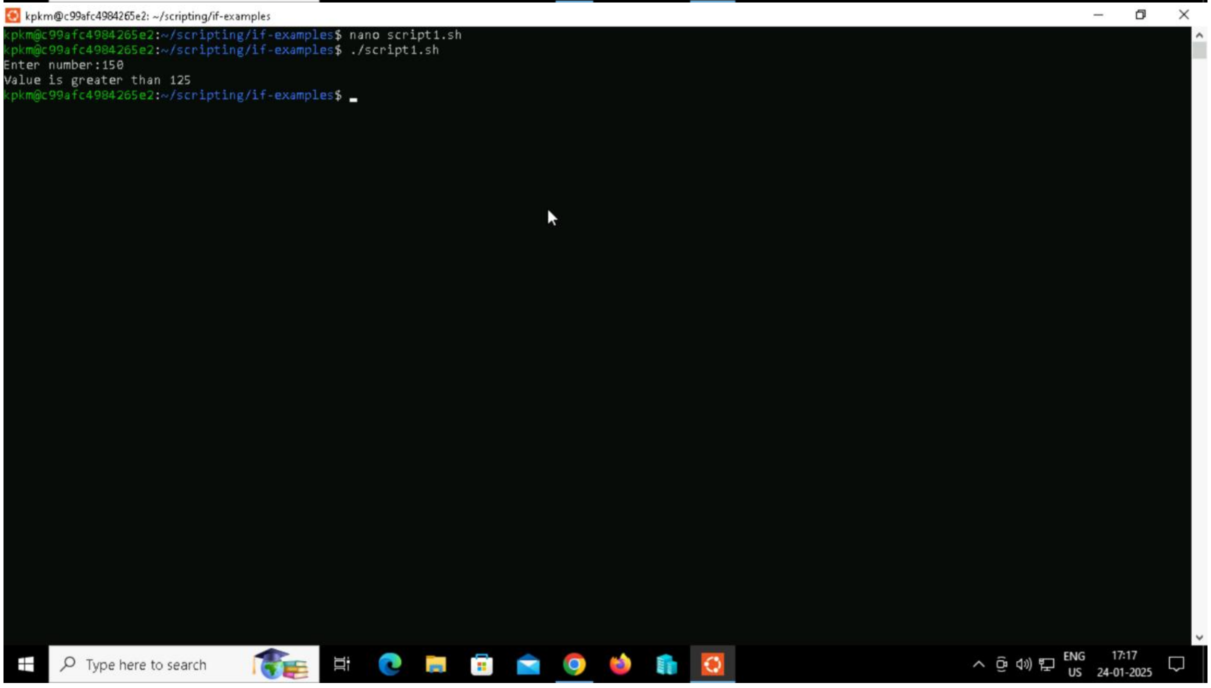
```
kpkm@c99afc4984265e2:~/scripting$ nano backtick_ar_ops.sh
kpkm@c99afc4984265e2:~/scripting$ ./backtick_ar_ops.sh
a=10, b=3
 c is the value of addition c=a+b
c=10+3
kpkm@c99afc4984265e2:~/scripting$ 
```

IF Operations

Q. Basic if operations

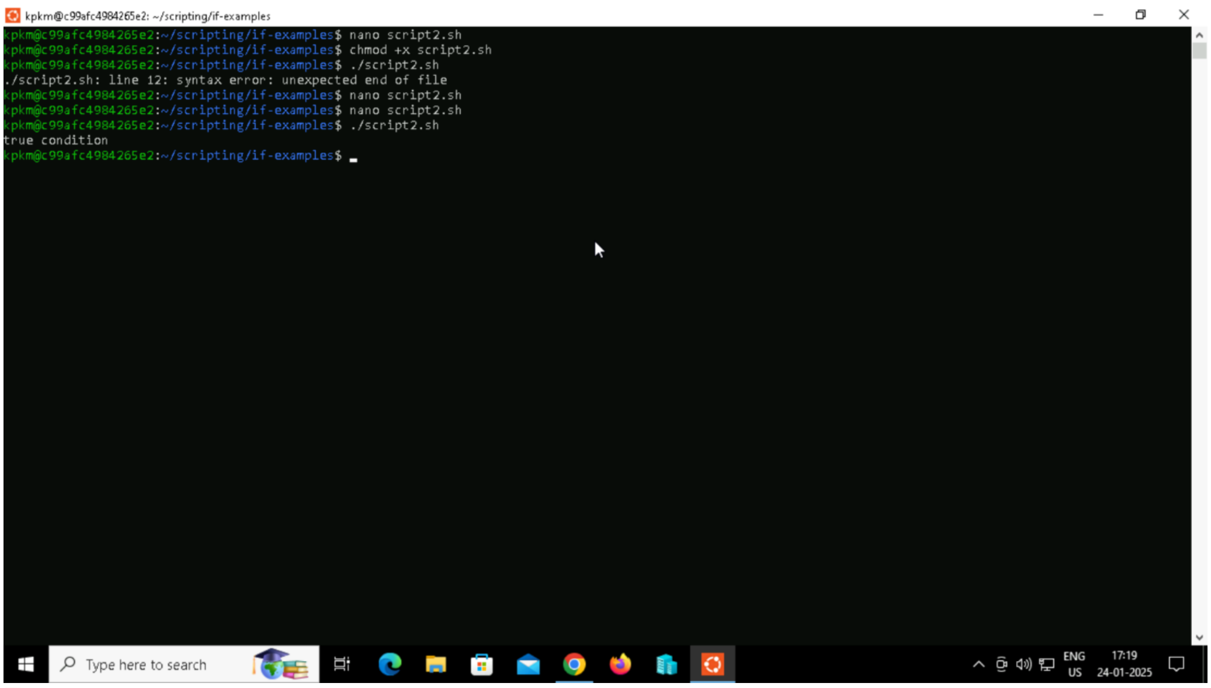


```
#This is a script to test basic if condition
#!/bin/bash

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



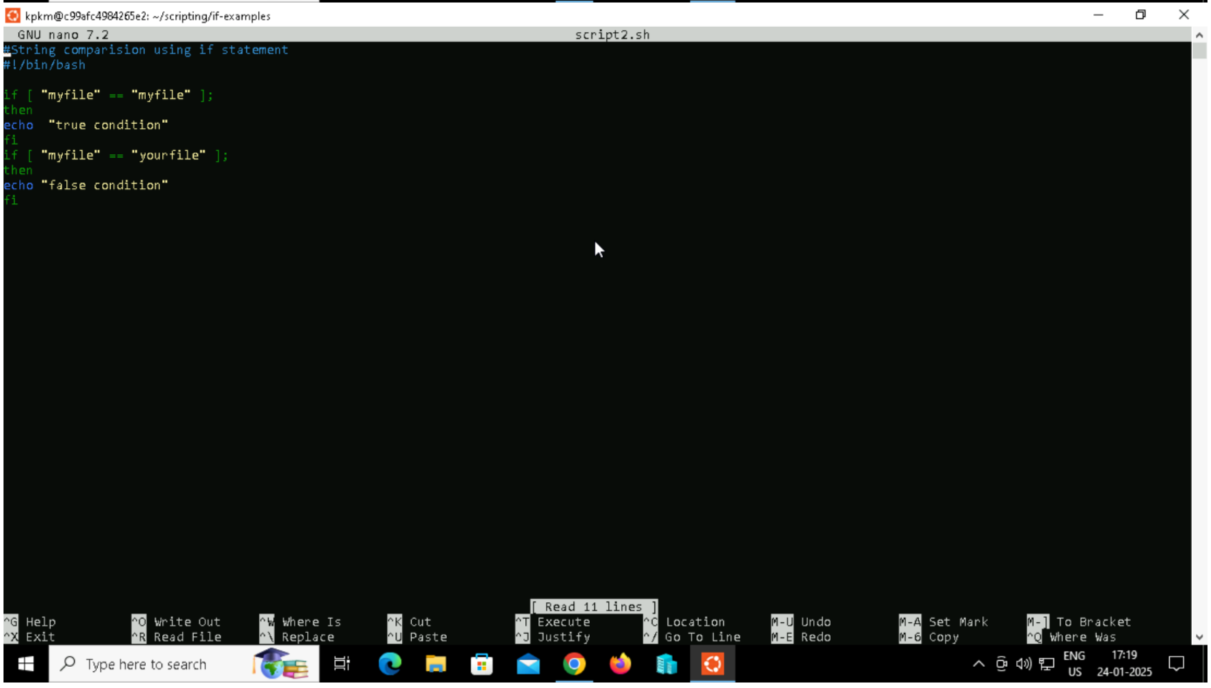
```
kpkm@c99afc4984265e2:~/scripting/if-examples$ nano script1.sh
kpkm@c99afc4984265e2:~/scripting/if-examples$ ./script1.sh
Enter number:150
Value is greater than 125
kpkm@c99afc4984265e2:~/scripting/if-examples$ 
```

Q. String comparison using if conditions



kpkm@c99afc4984265e2: ~/scripting/if-examples

```
kpkm@c99afc4984265e2:~/scripting/if-examples$ nano script2.sh
kpkm@c99afc4984265e2:~/scripting/if-examples$ chmod +x script2.sh
kpkm@c99afc4984265e2:~/scripting/if-examples$ ./script2.sh
./script2.sh: line 12: syntax error: unexpected end of file
kpkm@c99afc4984265e2:~/scripting/if-examples$ nano script2.sh
kpkm@c99afc4984265e2:~/scripting/if-examples$ nano script2.sh
kpkm@c99afc4984265e2:~/scripting/if-examples$ ./script2.sh
true condition
kpkm@c99afc4984265e2:~/scripting/if-examples$ 
```

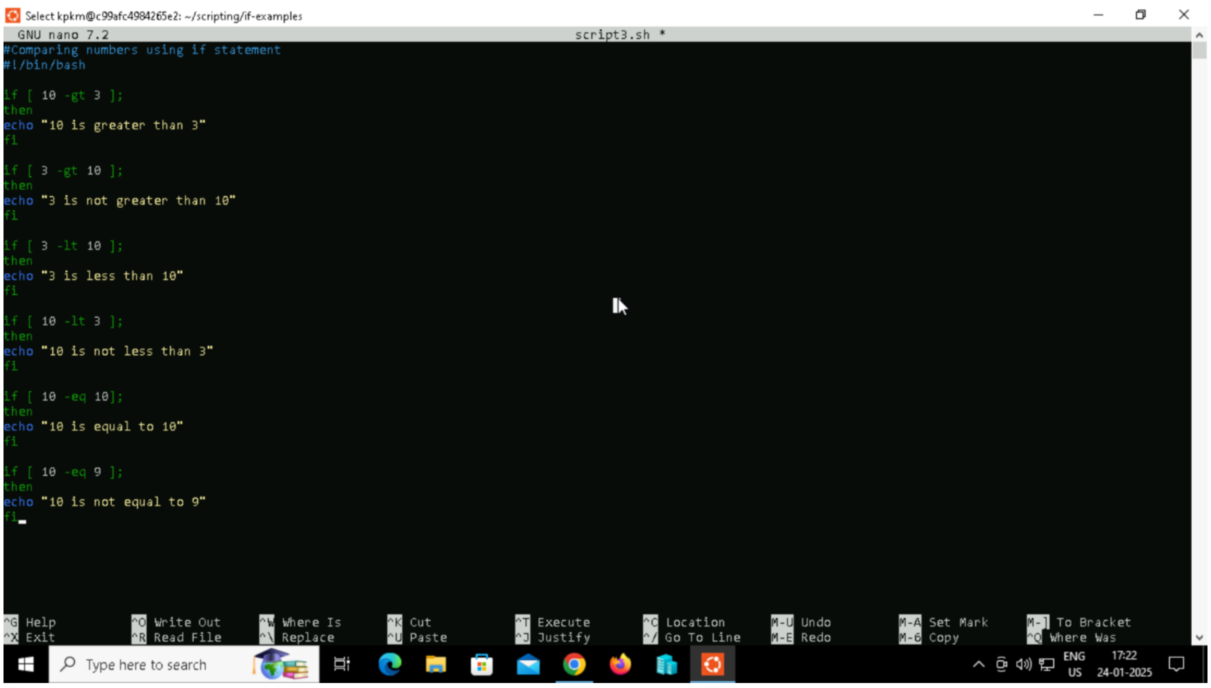


kpkm@c99afc4984265e2: ~/scripting/if-examples

```
  GNU nano 7.2                                   script2.sh
#String comparision using if statement
#!/bin/bash

if [ "myfile" == "myfile" ];
then
echo  "true condition"
fi
if [ "myfile" == "yourfile" ];
then
echo "false condition"
fi
```

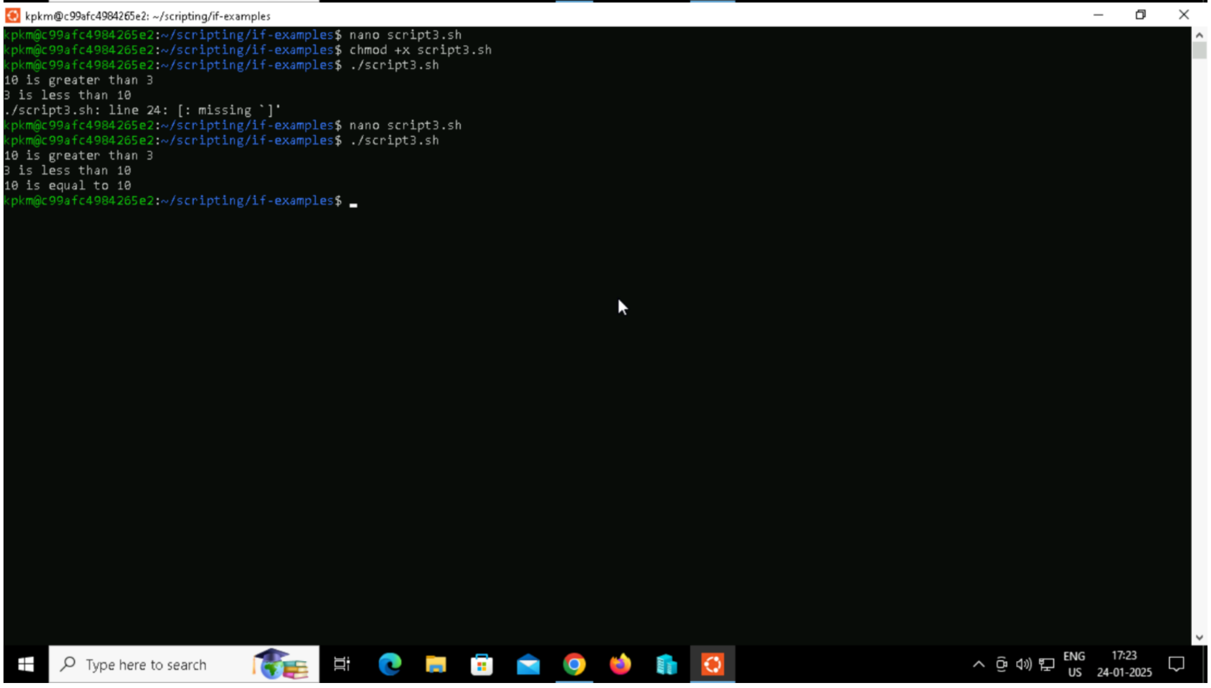
```
                                              [ Read 11 lines ]
^G Help        ^O Write Out   ^W Where Is    ^K Cut         ^T Execute     ^C Location    M-U Undo       M-A Set Mark   M-] To Bracket
^X Exit        ^R Read File   ^\ Replace     ^U Paste       ^J Justify     ^/ Go To Line  M-E Redo       M-6 Copy       ^Q Where Was
```

Q. Comparing numbers using if conditions



Select kpkm@c99afc4984265e2: ~/scripting/if-examples

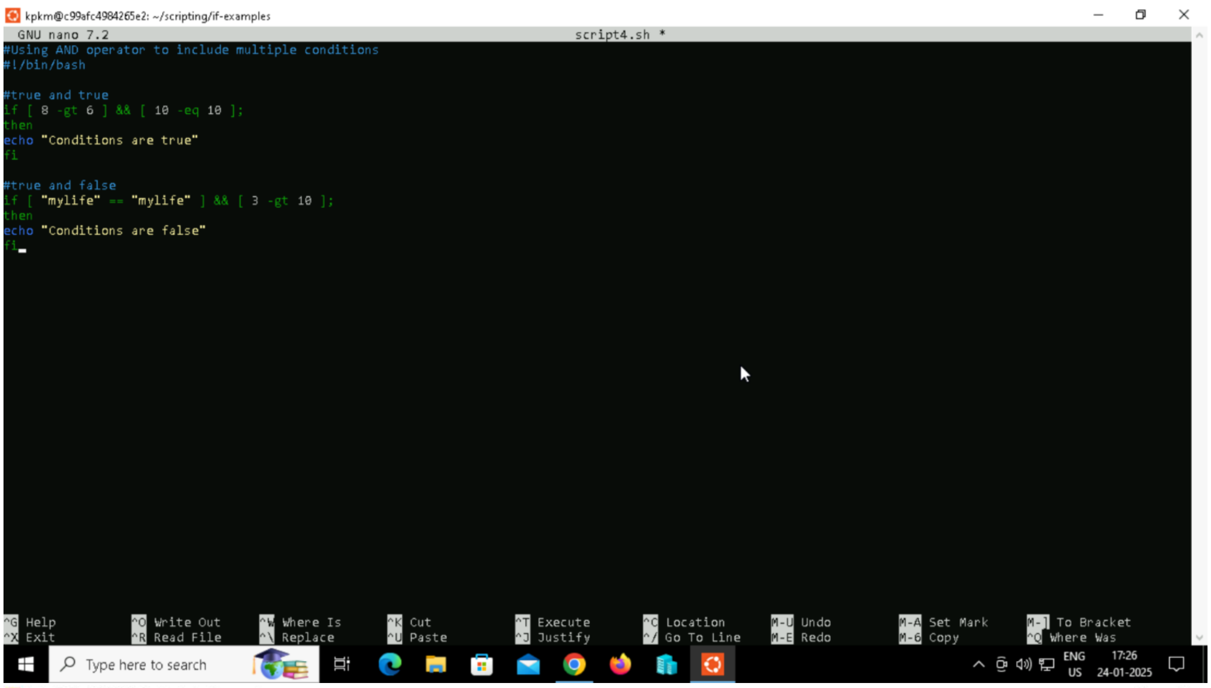
```
  GNU nano 7.2                                   script3.sh *
#Comparing numbers using if statement
#!/bin/bash

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
```



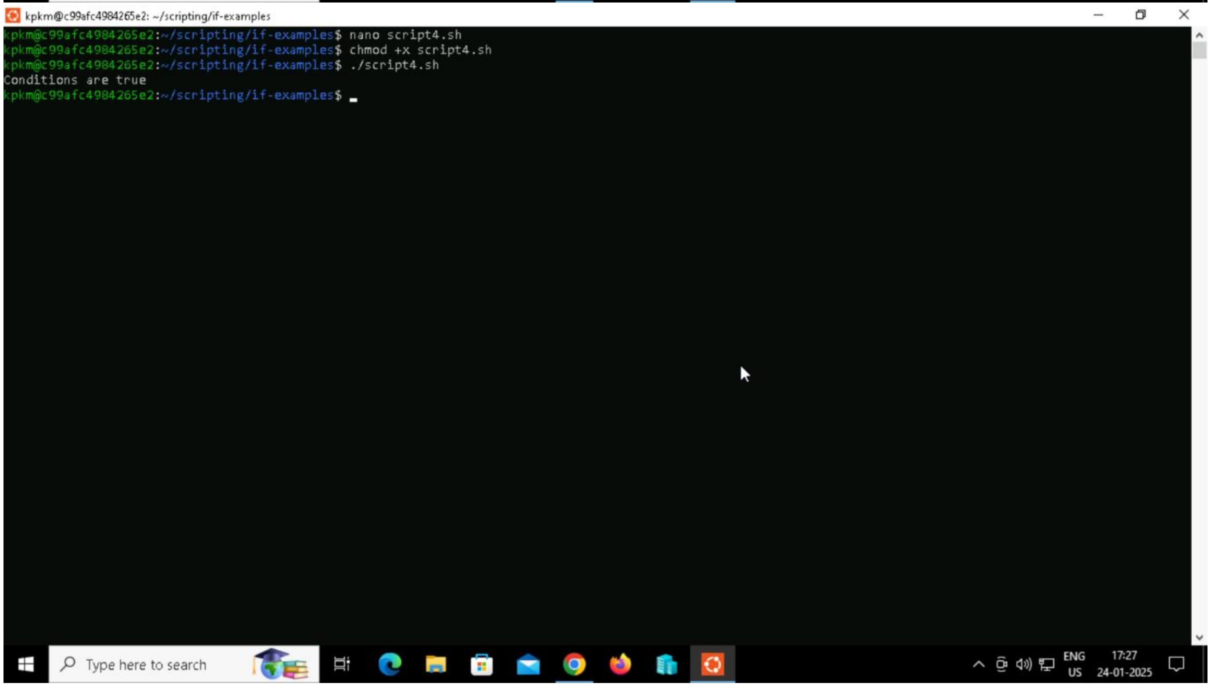
kpkm@c99afc4984265e2: ~/scripting/if-examples

```
kpkm@c99afc4984265e2:~/scripting/if-examples$ nano script3.sh
kpkm@c99afc4984265e2:~/scripting/if-examples$ chmod +x script3.sh
kpkm@c99afc4984265e2:~/scripting/if-examples$ ./script3.sh
10 is greater than 3
3 is less than 10
./script3.sh: line 24: [: missing `]'
kpkm@c99afc4984265e2:~/scripting/if-examples$ nano script3.sh
kpkm@c99afc4984265e2:~/scripting/if-examples$ ./script3.sh
10 is greater than 3
3 is less than 10
10 is equal to 10
kpkm@c99afc4984265e2:~/scripting/if-examples$ _
```

Q. Using AND operator

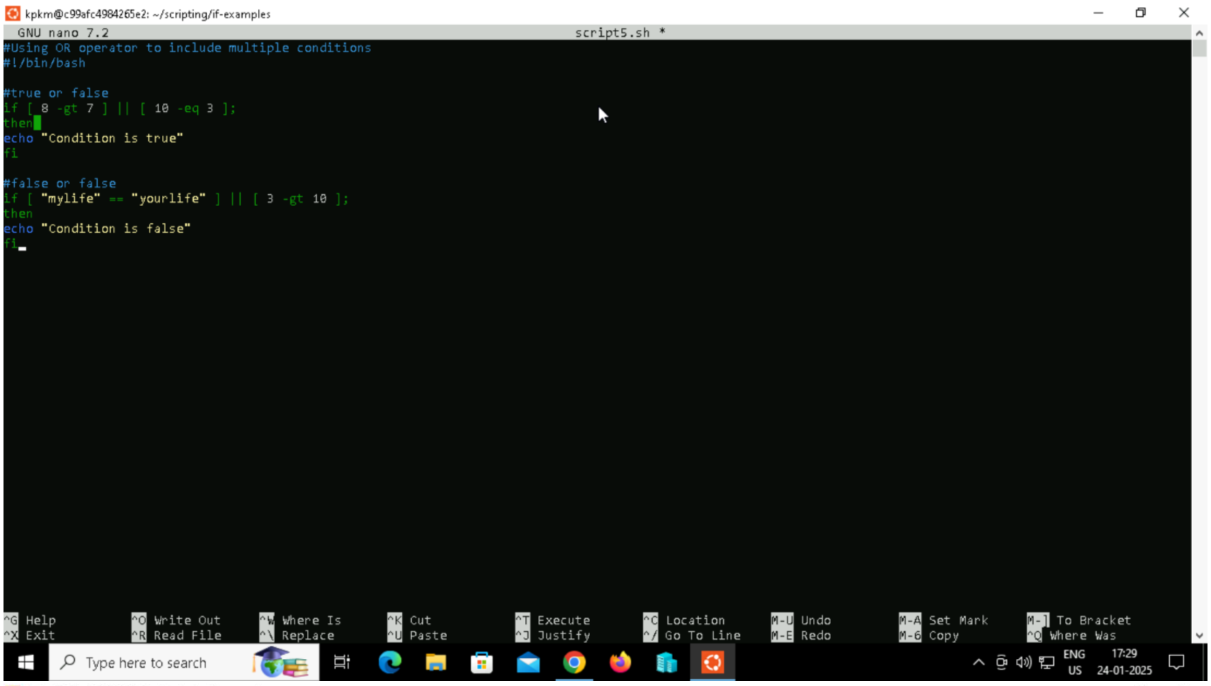


```
kpkm@c99afc4984265e2: ~/scripting/if-examples
  GNU nano 7.2                                   script4.sh *
#Using AND operator to include multiple conditions
#!/bin/bash

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

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

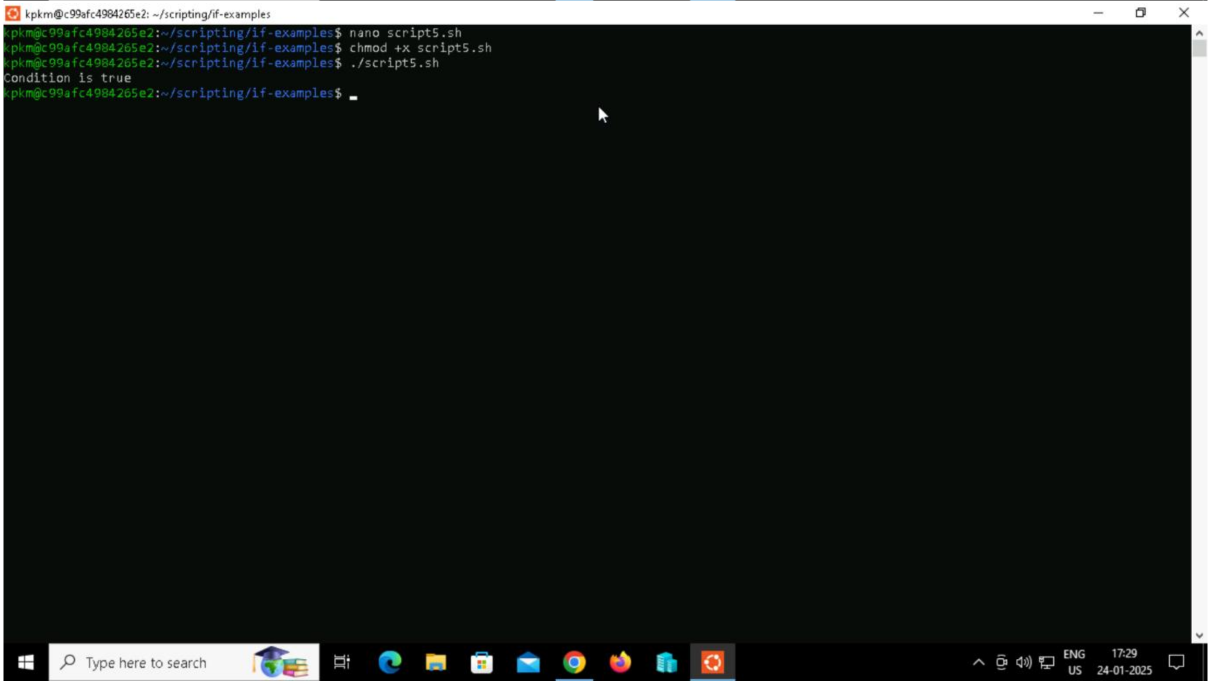


```
kpkm@c99afc4984265e2:~/scripting/if-examples$ nano script4.sh
kpkm@c99afc4984265e2:~/scripting/if-examples$ chmod +x script4.sh
kpkm@c99afc4984265e2:~/scripting/if-examples$ ./script4.sh
Conditions are true
kpkm@c99afc4984265e2:~/scripting/if-examples$
```

Q. Using OR operator

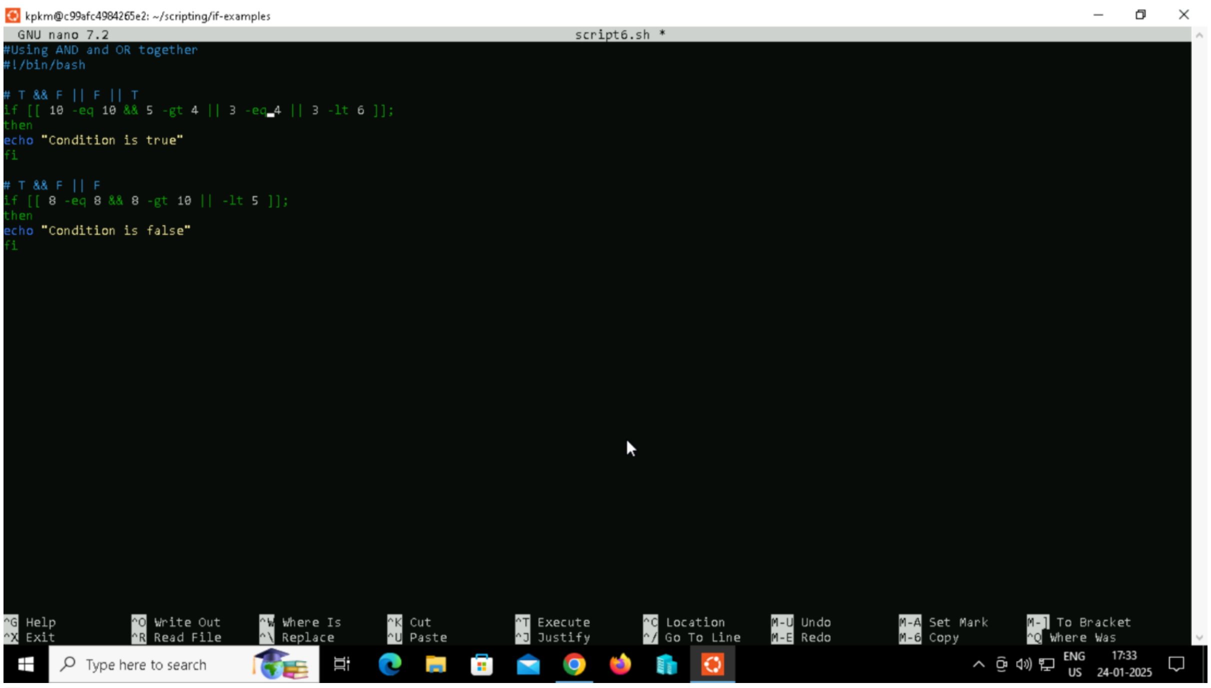


```
kpkm@c99afc4984265e2: ~/scripting/if-examples
  GNU nano 7.2                              script5.sh *
#Using OR operator to include multiple conditions
#!/bin/bash

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

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

^G Help        ^O Write Out   ^W Where Is    ^K Cut         ^T Execute     ^C Location    M-U Undo       M-A Set Mark   M-] To Bracket
^X Exit        ^R Read File   ^\ Replace     ^U Paste       ^J Justify     ^/ Go To Line  M-E Redo       M-6 Copy       ^Q Where Was
```

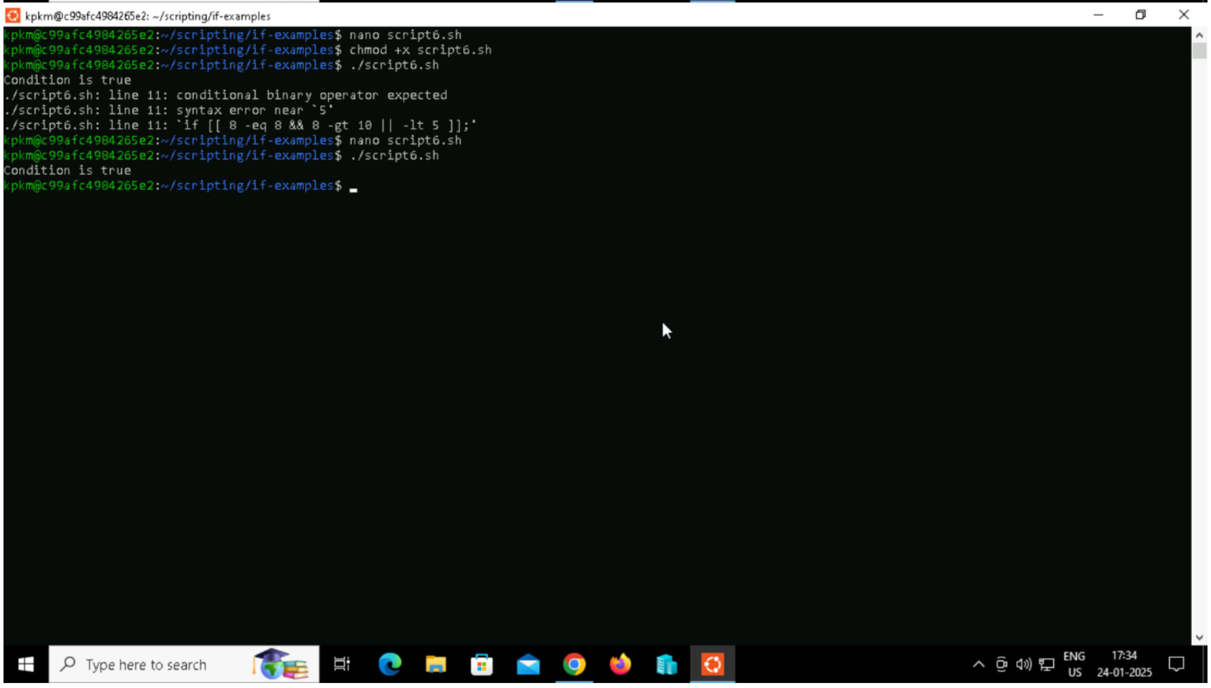


```
kpkm@c99afc4984265e2: ~/scripting/if-examples
kpkm@c99afc4984265e2:~/scripting/if-examples$ nano script5.sh
kpkm@c99afc4984265e2:~/scripting/if-examples$ chmod +x script5.sh
kpkm@c99afc4984265e2:~/scripting/if-examples$ ./script5.sh
Condition is true
kpkm@c99afc4984265e2:~/scripting/if-examples$
```

Q. Using AND and OR operator

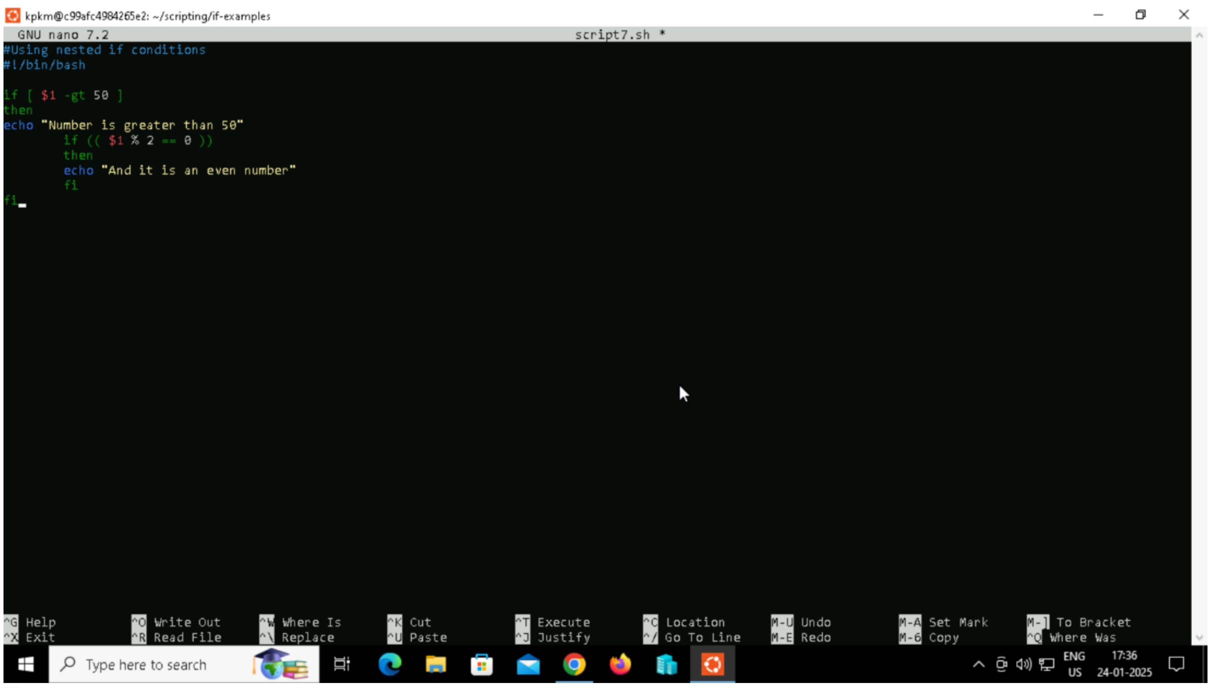


```
#Using AND and OR together
#!/bin/bash

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

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



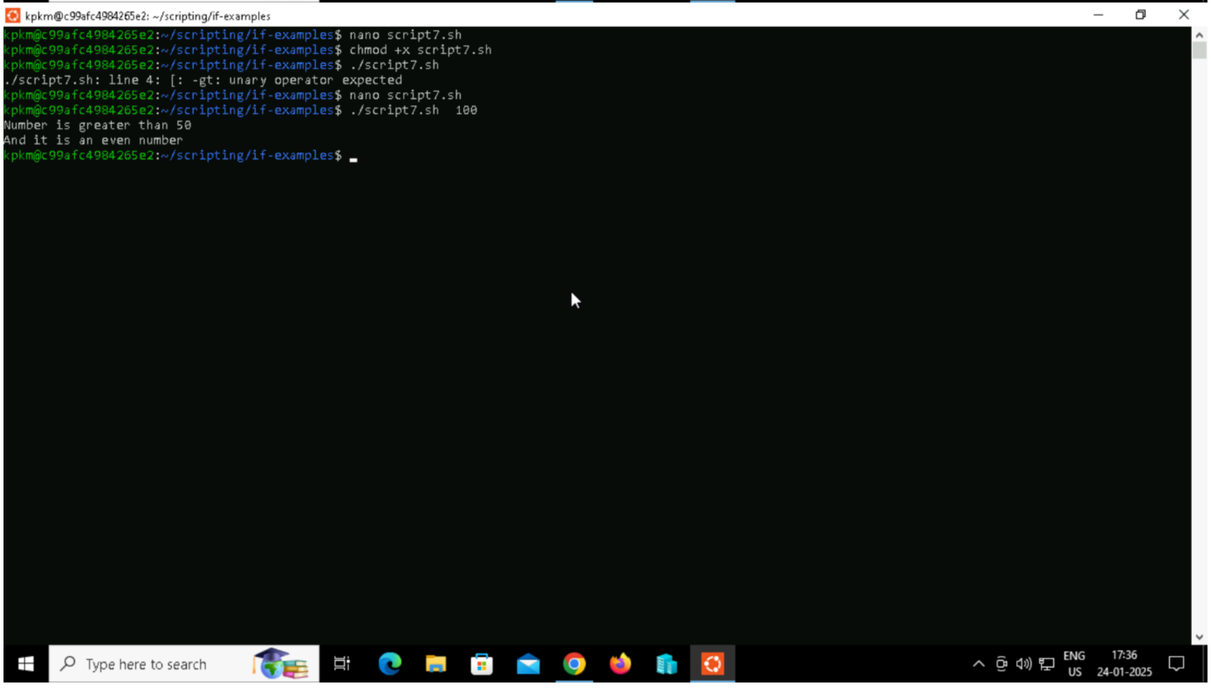
```
kpkm@c99afc4984265e2:~/scripting/if-examples$ nano script6.sh
kpkm@c99afc4984265e2:~/scripting/if-examples$ chmod +x script6.sh
kpkm@c99afc4984265e2:~/scripting/if-examples$ ./script6.sh
Condition is true
./script6.sh: line 11: conditional binary operator expected
./script6.sh: line 11: syntax error near `5'
./script6.sh: line 11: `if [[ 8 -eq 8 && 8 -gt 10 || -lt 5 ]];'
kpkm@c99afc4984265e2:~/scripting/if-examples$ nano script6.sh
kpkm@c99afc4984265e2:~/scripting/if-examples$ ./script6.sh
Condition is true
kpkm@c99afc4984265e2:~/scripting/if-examples$ 
```

Q. Using nested if conditions



kpkm@c99afc4984265e2: ~/scripting/if-examples

```
  GNU nano 7.2                                   script7.sh *
#Using nested if conditions
#!/bin/bash

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
```

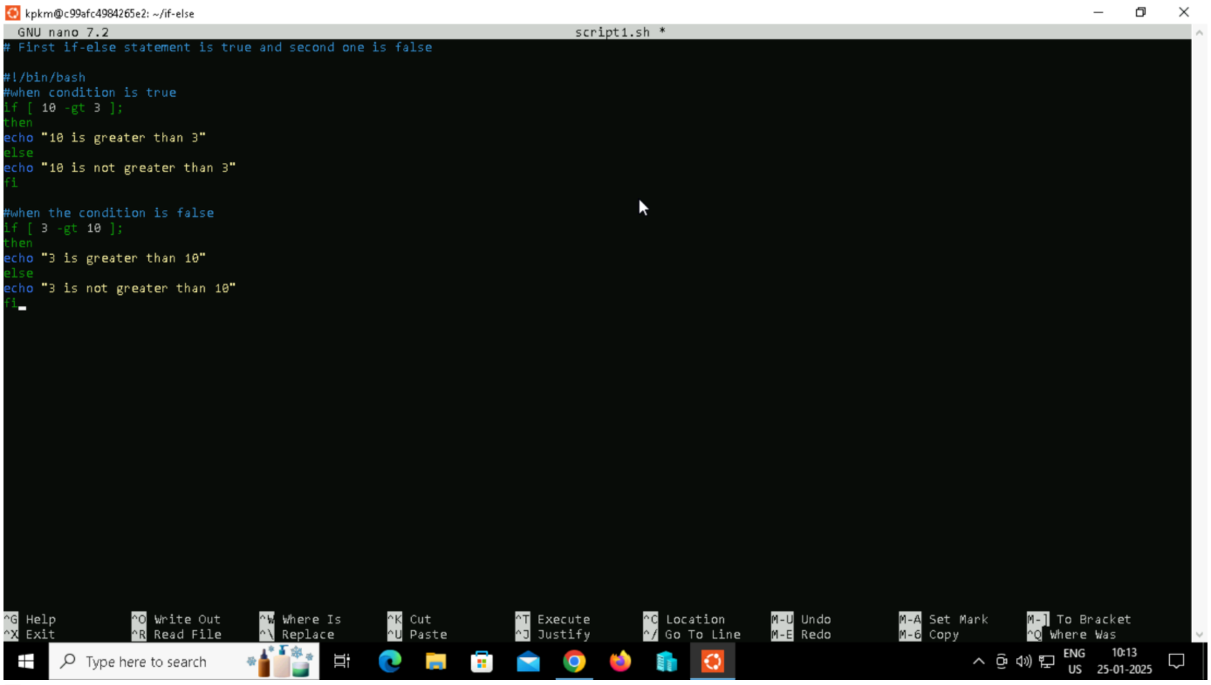


kpkm@c99afc4984265e2: ~/scripting/if-examples

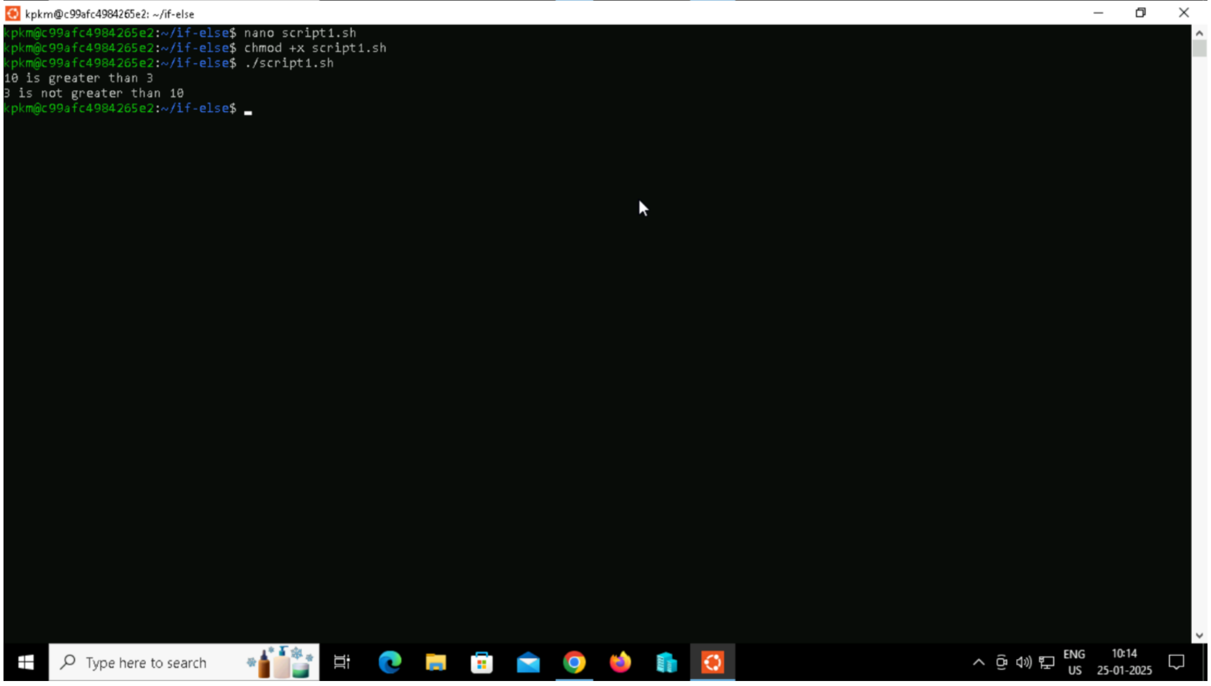
```
kpkm@c99afc4984265e2:~/scripting/if-examples$ nano script7.sh
kpkm@c99afc4984265e2:~/scripting/if-examples$ chmod +x script7.sh
kpkm@c99afc4984265e2:~/scripting/if-examples$ ./script7.sh
./script7.sh: line 4: [: -gt: unary operator expected
kpkm@c99afc4984265e2:~/scripting/if-examples$ nano script7.sh
kpkm@c99afc4984265e2:~/scripting/if-examples$ ./script7.sh  100
Number is greater than 50
And it is an even number
kpkm@c99afc4984265e2:~/scripting/if-examples$ 
```

If-Else and Elif operations

Q. If else operations

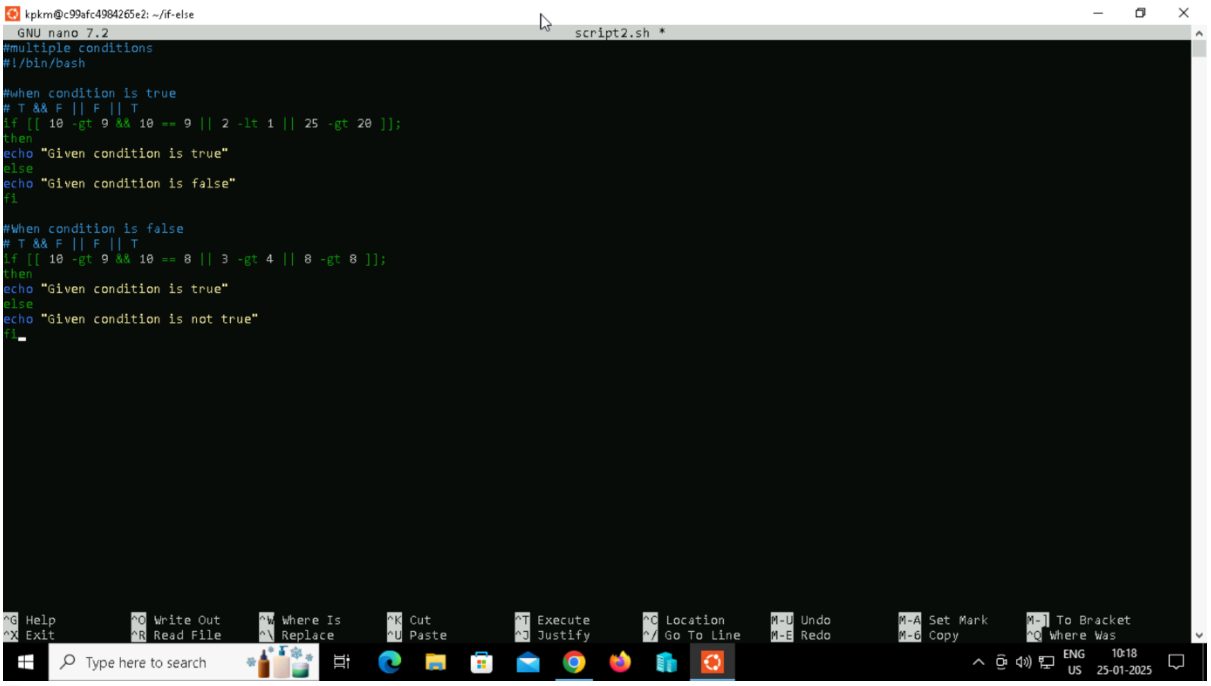


```
kpkm@c99afc4984265e2: ~/if-else
  GNU nano 7.2                                       script1.sh *
# First if-else statement is true and second one is false

#!/bin/bash
#when 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
```



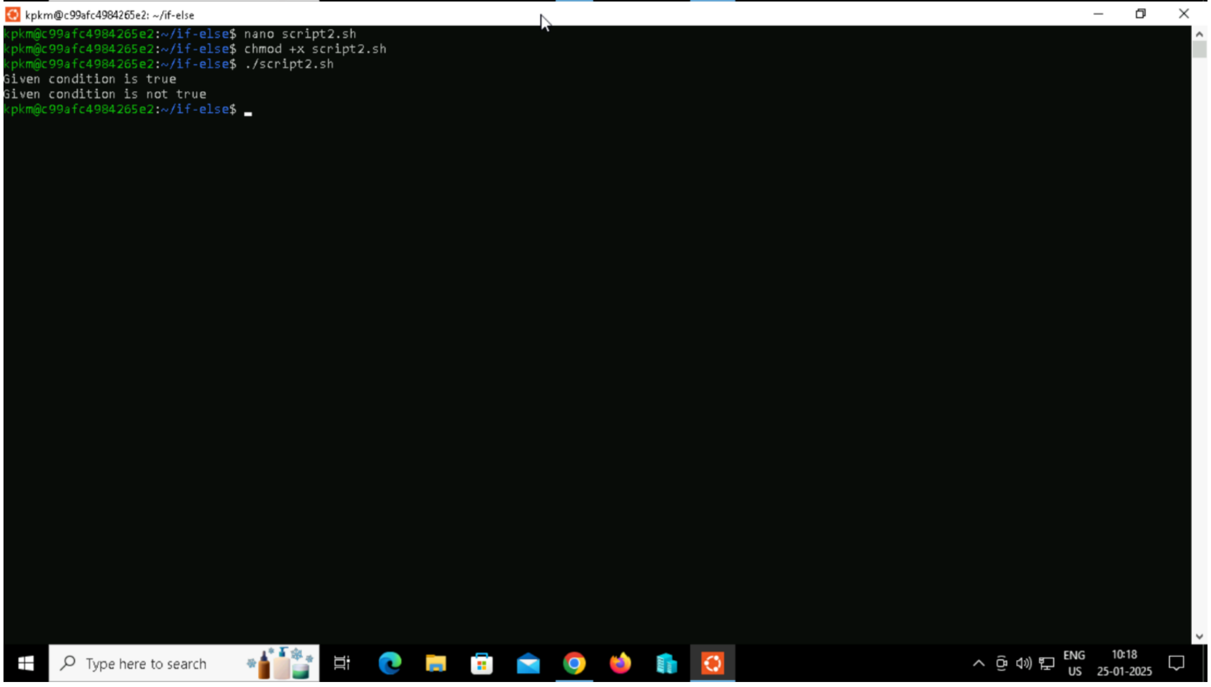
```
kpkm@c99afc4984265e2: ~/if-else
kpkm@c99afc4984265e2:~/if-else$ nano script1.sh
kpkm@c99afc4984265e2:~/if-else$ chmod +x script1.sh
kpkm@c99afc4984265e2:~/if-else$ ./script1.sh
10 is greater than 3
3 is not greater than 10
kpkm@c99afc4984265e2:~/if-else$
```

Q. Multiple if-else conditions



kpkm@c99afc4984265e2: ~/if-else

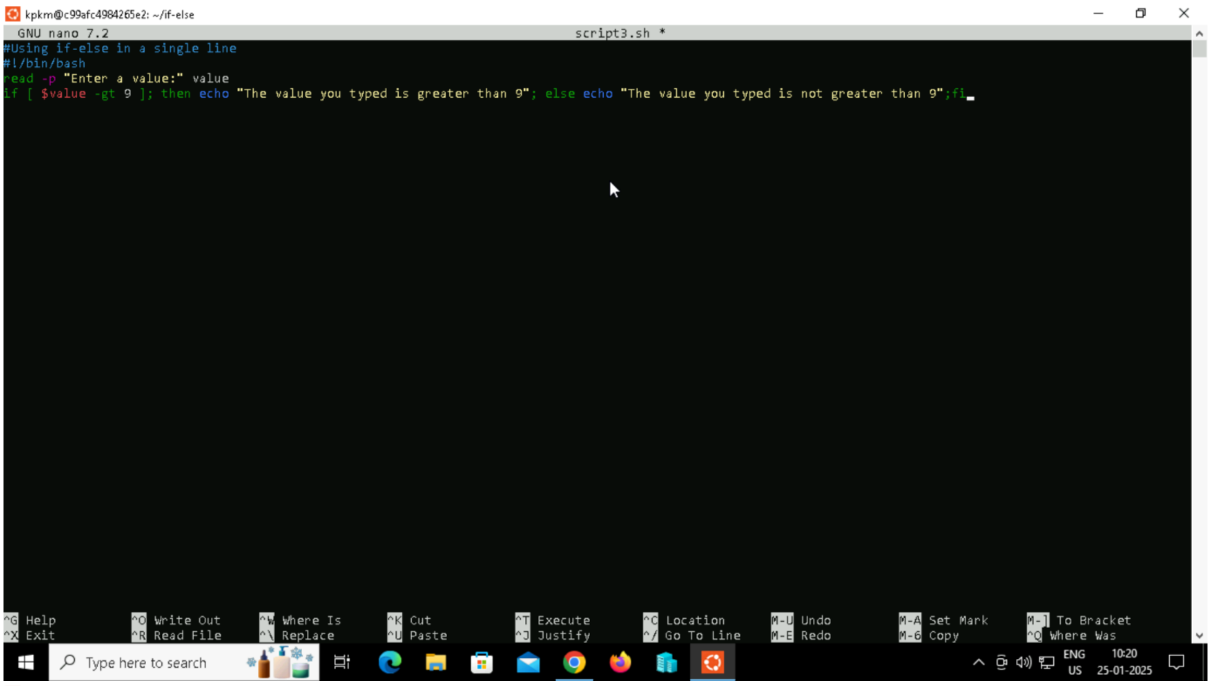
```
  GNU nano 7.2                                          script2.sh *
#multiple conditions
#!/bin/bash

#when condition is true
# T && F || F || T
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
# T && F || F || T
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
```



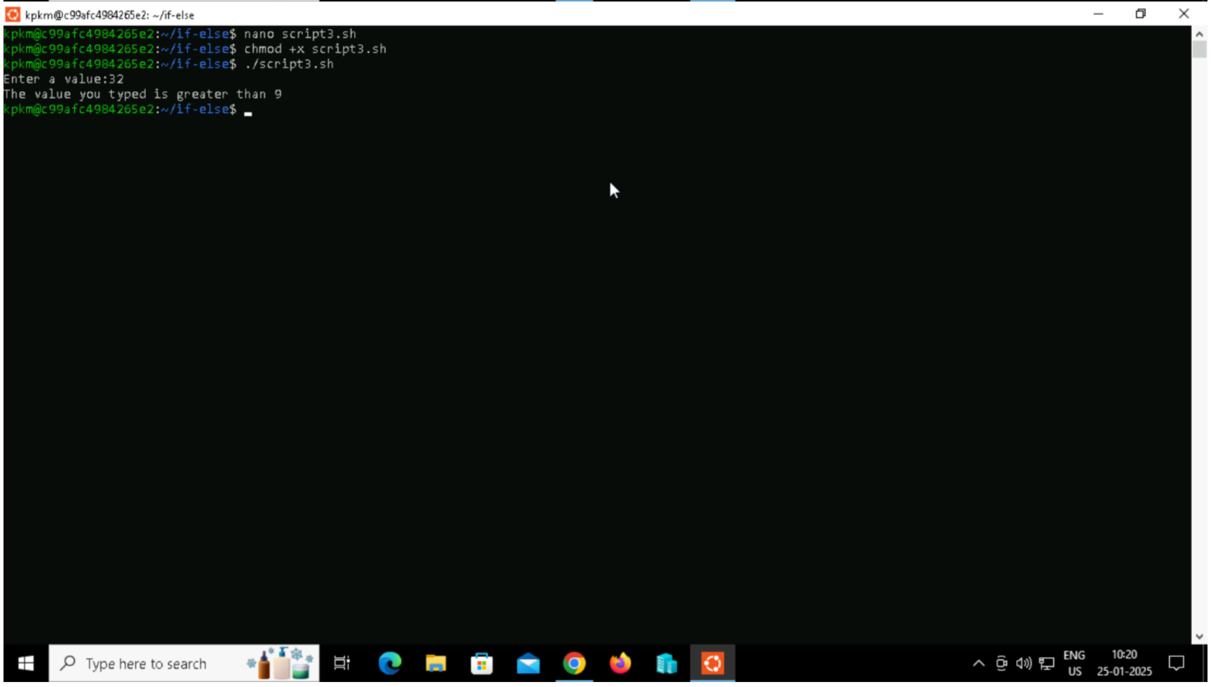
kpkm@c99afc4984265e2: ~/if-else

```
kpkm@c99afc4984265e2:~/if-else$ nano script2.sh
kpkm@c99afc4984265e2:~/if-else$ chmod +x script2.sh
kpkm@c99afc4984265e2:~/if-else$ ./script2.sh
Given condition is true
Given condition is not true
kpkm@c99afc4984265e2:~/if-else$
```

Q. Using if-else in a single line

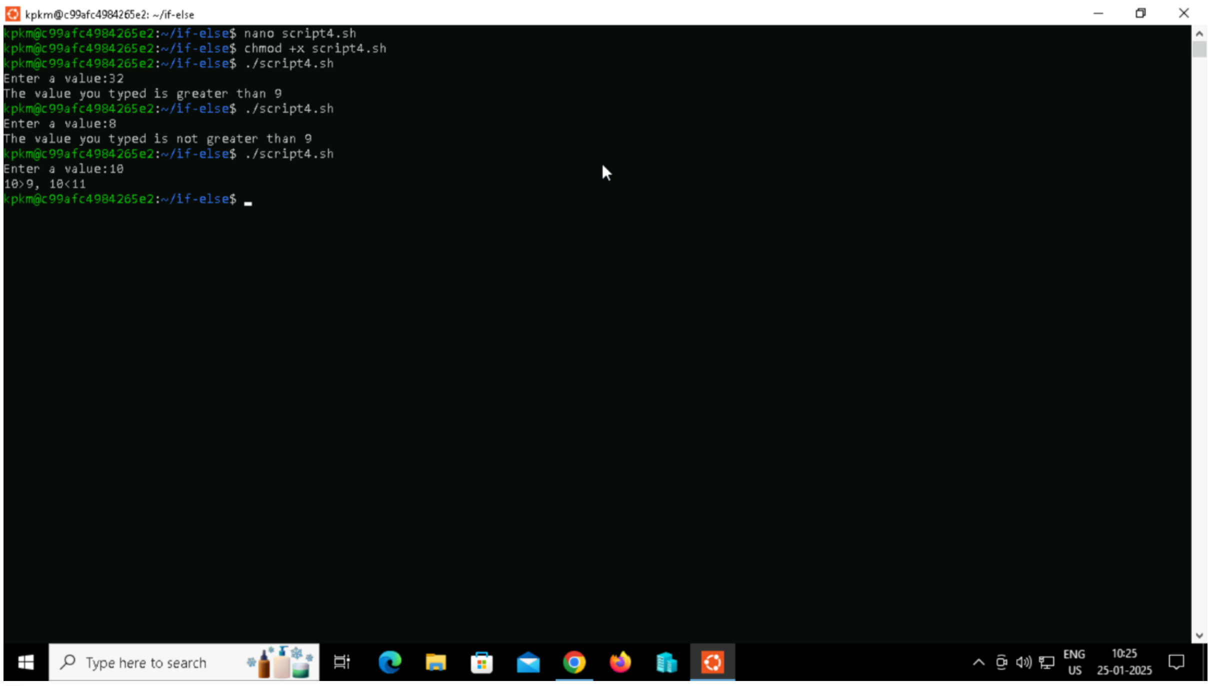


```
#Using if-else in a single line
#!/bin/bash
read -p "Enter a value:" value
if [ $value -gt 9 ]; then echo "The value you typed is greater than 9"; else echo "The value you typed is not greater than 9";fi
```



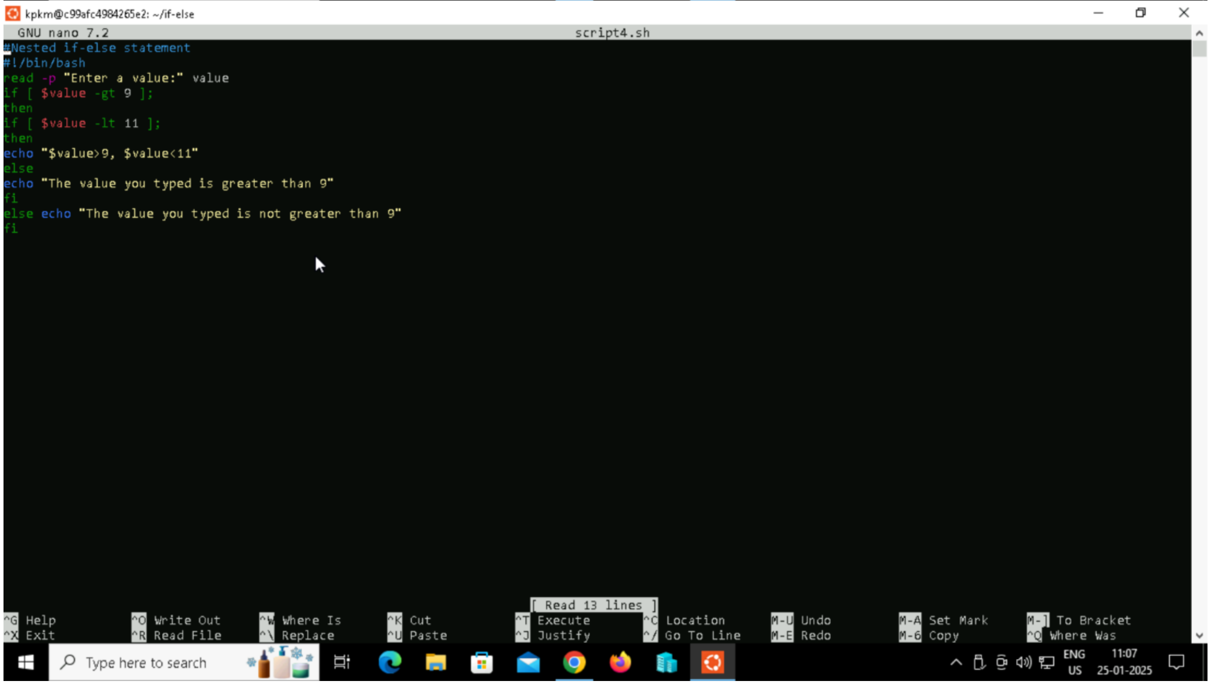
```
kpkm@c99afc4984265e2:~/if-else$ nano script3.sh
kpkm@c99afc4984265e2:~/if-else$ chmod +x script3.sh
kpkm@c99afc4984265e2:~/if-else$ ./script3.sh
Enter a value:32
The value you typed is greater than 9
kpkm@c99afc4984265e2:~/if-else$
```

Q. Nested if-else statement



kpkm@c99afc4984265e2: ~/if-else

```
kpkm@c99afc4984265e2:~/if-else$ nano script4.sh
kpkm@c99afc4984265e2:~/if-else$ chmod +x script4.sh
kpkm@c99afc4984265e2:~/if-else$ ./script4.sh
Enter a value:32
The value you typed is greater than 9
kpkm@c99afc4984265e2:~/if-else$ ./script4.sh
Enter a value:8
The value you typed is not greater than 9
kpkm@c99afc4984265e2:~/if-else$ ./script4.sh
Enter a value:10
10>9, 10<11
kpkm@c99afc4984265e2:~/if-else$ 
```

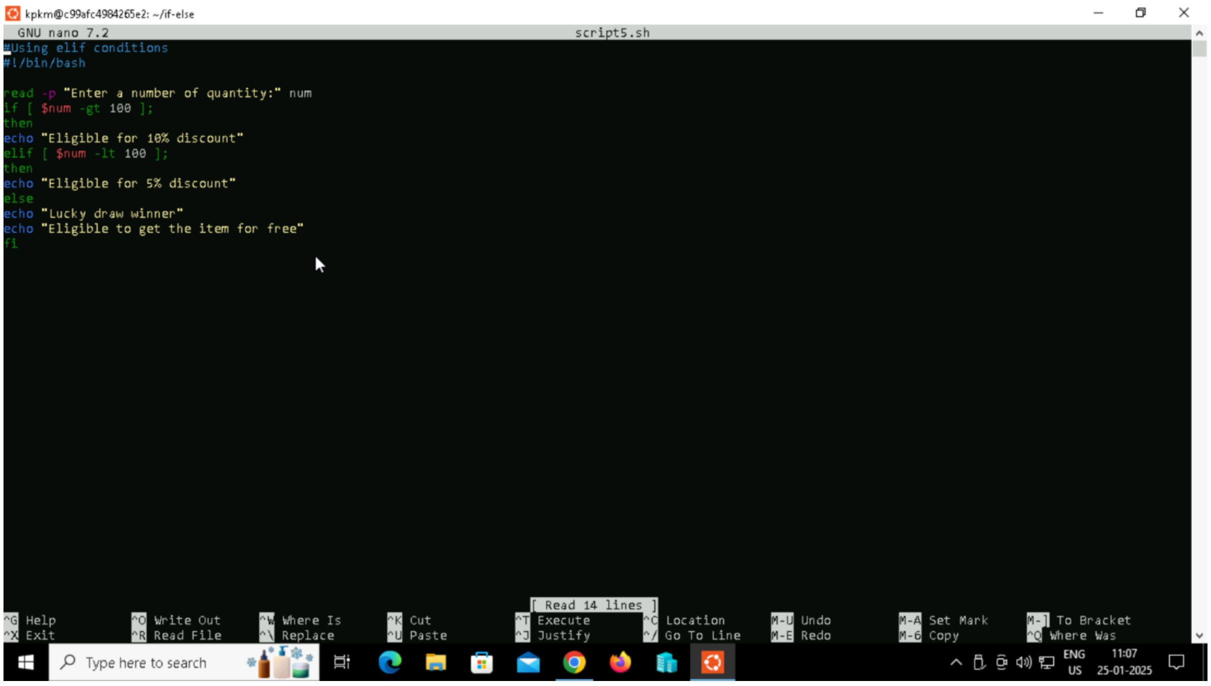


kpkm@c99afc4984265e2: ~/if-else

```
  GNU nano 7.2                                   script4.sh
#Nested if-else statement
#!/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
```

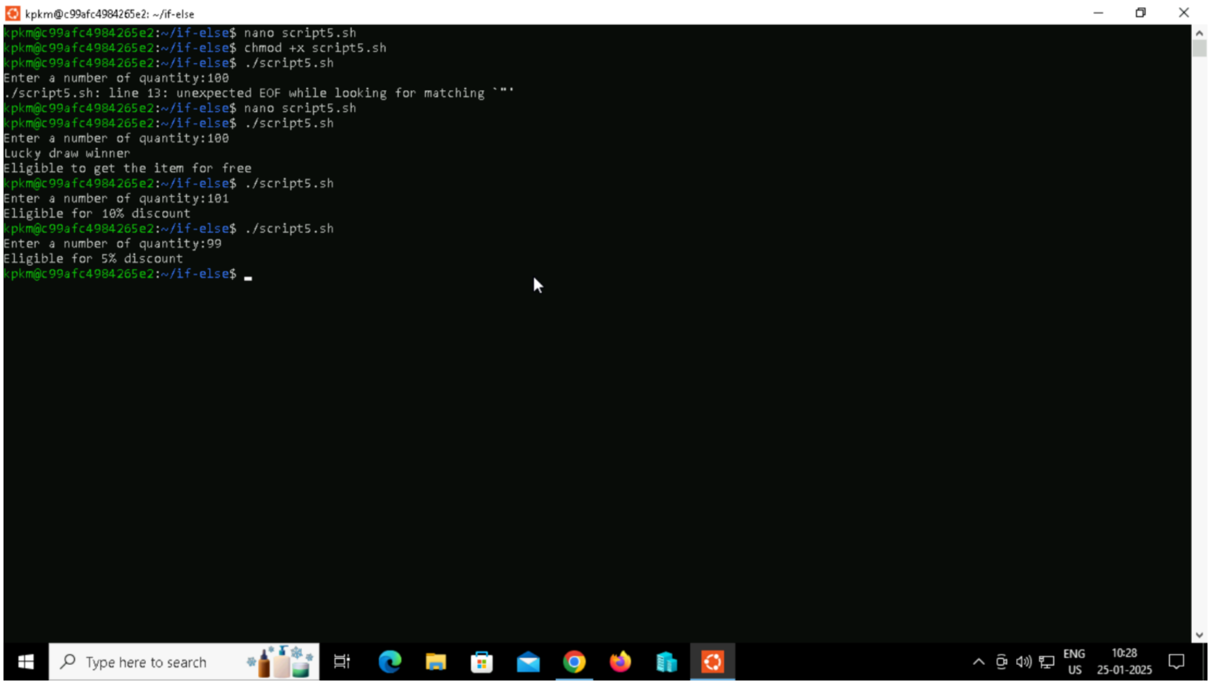
```
                                        [ Read 13 lines ]
^G Help      ^O Write Out  ^W Where Is   ^K Cut      ^T Execute   ^C Location   M-U Undo   M-A Set Mark   M-] To Bracket
^X Exit      ^R Read File  ^\ Replace    ^U Paste    ^J Justify   ^/ Go To Line M-E Redo   M-6 Copy       ^Q Where Was
```

Q. El-if expressions



kpkm@c99afc4984265e2: ~/if-else

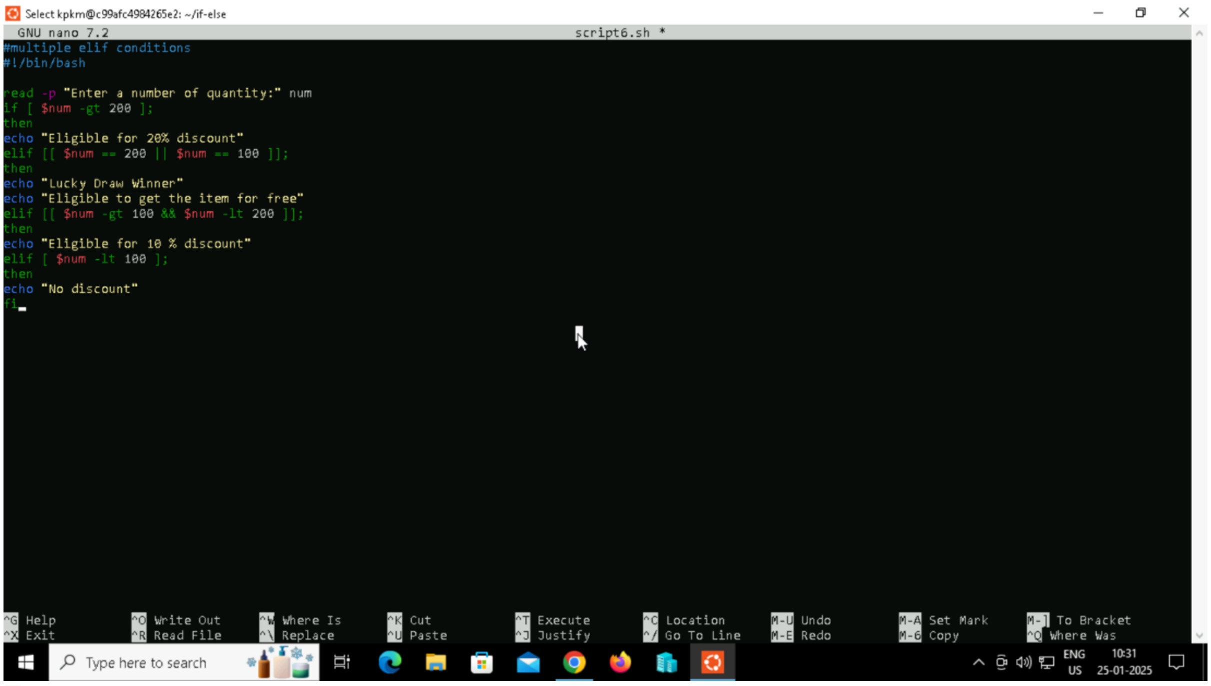
```
  GNU nano 7.2                                   script5.sh
#Using elif conditions
#!/bin/bash

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

                                   [ Read 14 lines ]
^G Help       ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location   M-U Undo      M-A Set Mark  M-] To Bracket
^X Exit       ^R Read File  ^\ Replace    ^U Paste      ^J Justify    ^/ Go To Line M-E Redo      M-6 Copy      ^Q Where Was
```



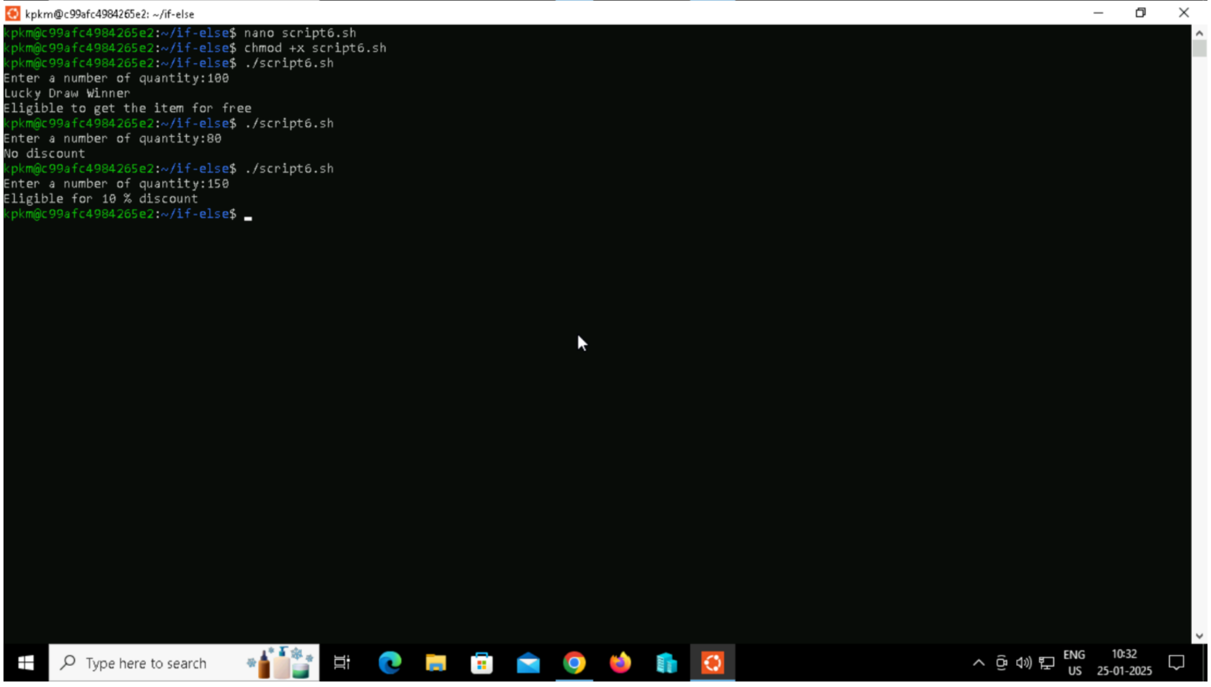
kpkm@c99afc4984265e2: ~/if-else

```
kpkm@c99afc4984265e2:~/if-else$ nano script5.sh
kpkm@c99afc4984265e2:~/if-else$ chmod +x script5.sh
kpkm@c99afc4984265e2:~/if-else$ ./script5.sh
Enter a number of quantity:100
./script5.sh: line 13: unexpected EOF while looking for matching `"'
kpkm@c99afc4984265e2:~/if-else$ nano script5.sh
kpkm@c99afc4984265e2:~/if-else$ ./script5.sh
Enter a number of quantity:100
Lucky draw winner
Eligible to get the item for free
kpkm@c99afc4984265e2:~/if-else$ ./script5.sh
Enter a number of quantity:101
Eligible for 10% discount
kpkm@c99afc4984265e2:~/if-else$ ./script5.sh
Enter a number of quantity:99
Eligible for 5% discount
kpkm@c99afc4984265e2:~/if-else$ _
```

Q. Multiple el-if condition



```
#multiple elif conditions
#!/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
```



```
kpkm@c99afc4984265e2:~/if-else$ nano script6.sh
kpkm@c99afc4984265e2:~/if-else$ chmod +x script6.sh
kpkm@c99afc4984265e2:~/if-else$ ./script6.sh
Enter a number of quantity:100
Lucky Draw Winner
Eligible to get the item for free
kpkm@c99afc4984265e2:~/if-else$ ./script6.sh
Enter a number of quantity:80
No discount
kpkm@c99afc4984265e2:~/if-else$ ./script6.sh
Enter a number of quantity:150
Eligible for 10 % discount
kpkm@c99afc4984265e2:~/if-else$
```