Arithmetic Operations

1. Arithmetic Operation using bash scripting

Step -1: Created a Script file using touch command as “arithmetic.sh”.

Step -2: Opened a nano file to write the Script.

Step -3: Changes the file permission.

```
root@8b5c7dd85f01583: ~
root@8b5c7dd85f01583:~# touch arithmetic.sh
root@8b5c7dd85f01583:~# nano arithmetic.sh
root@8b5c7dd85f01583:~# chmod +x arithmetic.sh
```

The Nano Editor:

```
root@8b5c7dd85f01583: ~
  GNU nano 7.2                                  arithmetic.sh
#! /bin/bash
X=9
y=4
echo "X=9, y=4"
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 and y"
echo $(( $x /$y ))
echo "Exponentiation of x and 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 "Remaonder of dividing x by 5, x="
(( x %= 5 ))
echo $x
```

Output:

```
root@8b5c7dd85f01583:~# ./arithmetic.sh
X=9, y=4
Addition of x & y
13
Subtraction of x & y
5
Multiplication of x & y
36
Division of x and y
2
Exponentiation of x and y
6561
Modular Division of x,y
1
Incrementing x by 5,then  x=
14
Decrementing x by 5,then x=
9
Multiply of x by 5,then x=
45
Dividing x by 5,x=
9
Remaonder of dividing x by 5, x=
4
```

2.Arithmetic Expression

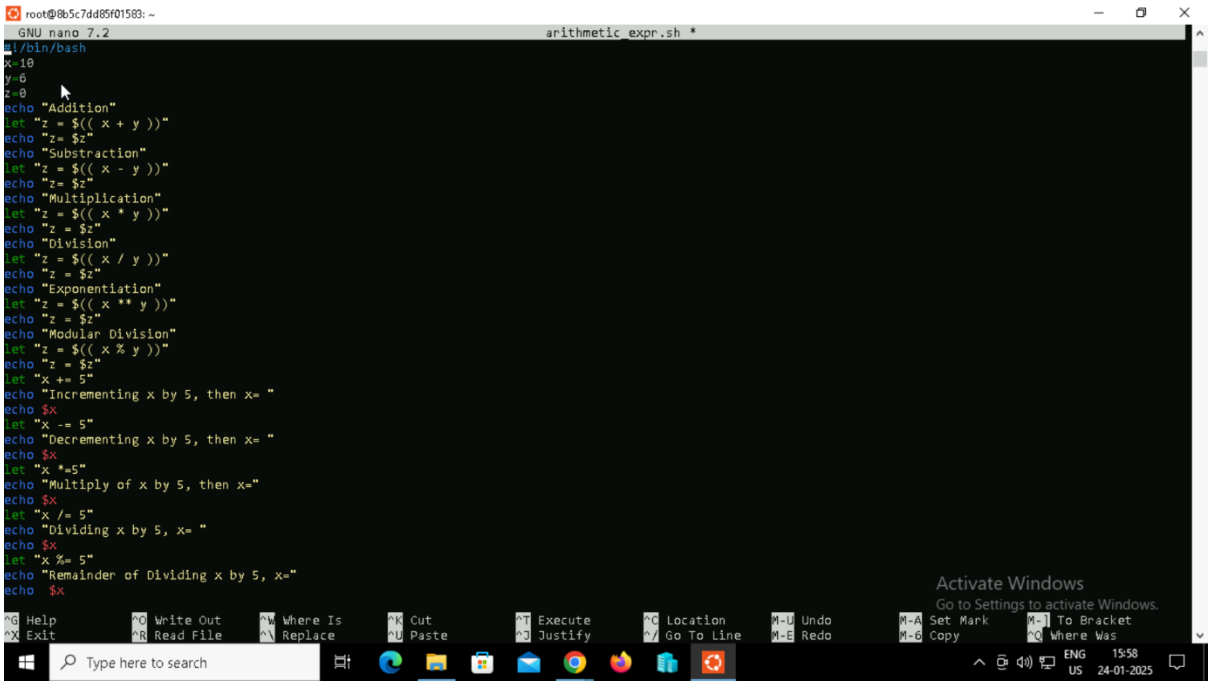
Step -1: Created a Script file using touch command as “arithmetic_expr.sh”.

Step -2: Opened a nano file to write the Script.

Step -3: Changes the file permission.

```
root@8b5c7dd85f01583:~# touch arithmetic_expr.sh
root@8b5c7dd85f01583:~# nano arithmetic_expr.sh
root@8b5c7dd85f01583:~# chmod +x arithmetic_expr.sh
```

The Nano Editor:



Output:

3. Using expr

Step -1: Created a Script file using touch command as “backticks.sh”.

Step -2: Opened a nano file to write the Script.

Step -3: Changes the file permission.

```
root@8b5c7dd85f01583:~# touch backticks.sh
root@8b5c7dd85f01583:~# nano backticks.sh
```

The Nano Editor:

```
root@8b5c7dd85f01583: ~
  GNU nano 7.2                    backticks.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`"
```

Output:

```
root@8b5c7dd85f01583:~# chmod +x backticks.sh
root@8b5c7dd85f01583:~# ./backticks.sh
a=10, b=3
c is the value of addition c=a+b
c= 13
```

Basic If Statement

4. Take a user input of any number and check if the value is greater than 125.

Step -1: Created a Script file using touch command as “if_cond.sh”.

Step -2: Opened a nano file to write the Script.

Step -3: Changes the file permission.

```
root@8b5c7dd85f01583: ~
root@8b5c7dd85f01583:~# touch if_cond.sh
root@8b5c7dd85f01583:~# chmod +x if_cond.sh
root@8b5c7dd85f01583:~# nano if_cond.sh
```

The Nano Editor:

```
root@8b5c7dd85f01583: ~
  GNU nano 7.2                    if_cond.sh *
#!/bin/bash
read -p " Enter number : " number
if [ $number -gt 125 ]
then
echo "Value is greater than 125"
fi
```

Output:

```
root@8b5c7dd85f01583:~# ./if_cond.sh
 Enter number : 158
Value is greater than 125
root@8b5c7dd85f01583:~#
```

5. Comparing of two string using if statement.

Step -1: Created a Script file using touch command as “if_cond1.sh”.

Step -2: Opened a nano file to write the Script.

Step -3: Changes the file permission.

```
root@8b5c7dd85f01583:~# touch if_cond1.sh
root@8b5c7dd85f01583:~# chmod +x if_cond1.sh
root@8b5c7dd85f01583:~# nano if_cond1.sh
root@8b5c7dd85f01583:~# ./if_cond1.sh
true condition
root@8b5c7dd85f01583:~#
```

The Nano Editor:

```
root@8b5c7dd85f01583: ~
  GNU nano 7.2                                  if_cond1.sh *
#!/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
```

Output:

```
root@8b5c7dd85f01583:~# ./if_cond1.sh
true condition
root@8b5c7dd85f01583:~#
```

6. Compare numbers by using the if statement.

Step -1: Created a Script file using touch command as “if_cond2.sh”.

Step -2: Opened a nano file to write the Script.

Step -3: Changes the file permission.

```
root@8b5c7dd85f01583:~# touch if_cond2.sh
root@8b5c7dd85f01583:~# chmod +x if_cond2.sh
root@8b5c7dd85f01583:~# nano if_cond2.sh
```

The Nano Editor:

```
root@8b5c7dd85f01583: ~
  GNU nano 7.2                                  if_cond2.sh *

#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
```

Output:

```
root@8b5c7dd85f01583:~# ./if_cond2.sh
10 is greater than 3.
9 is less than 10.
10 is equal to 10.
root@8b5c7dd85f01583:~#
```

7. Using AND Operators along with multiple conditions.

Step -1: Created a Script file using touch command as “if_cond3.sh”.

Step -2: Opened a nano file to write the Script.

Step -3: Changes the file permission.

```
root@8b5c7dd85f01583:~# touch if_cond3.sh
root@8b5c7dd85f01583:~# chmod +x if_cond3.sh
root@8b5c7dd85f01583:~# nano if_cond3.sh
```

The Nano Editor:

```
root@8b5c7dd85f01583: ~
  GNU nano 7.2                                  if_cond3.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
```

Output:

```
root@8b5c7dd85f01583:~# ./if_cond3.sh
Conditions are true
```

8.Using OR operator with multiple conditions.

Step -1: Created a Script file using touch command as “if_cond4.sh”.

Step -2: Opened a nano file to write the Script.

Step -3: Changes the file permission.

```
root@8b5c7dd85f01583:~# touch if_cond4.sh
root@8b5c7dd85f01583:~# chmod +x if_cond4.sh
root@8b5c7dd85f01583:~# nano if_cond4.sh
```

The Nano Editor:

```
root@8b5c7dd85f01583: ~
  GNU nano 7.2                                  if_cond4.sh *
#!/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
```

Output:

```
root@8b5c7dd85f01583:~# ./if_cond4.sh
 Condition is true.
root@8b5c7dd85f01583:~#
```

9. Use both AND and OR operators to include multiple conditions in the if expression.

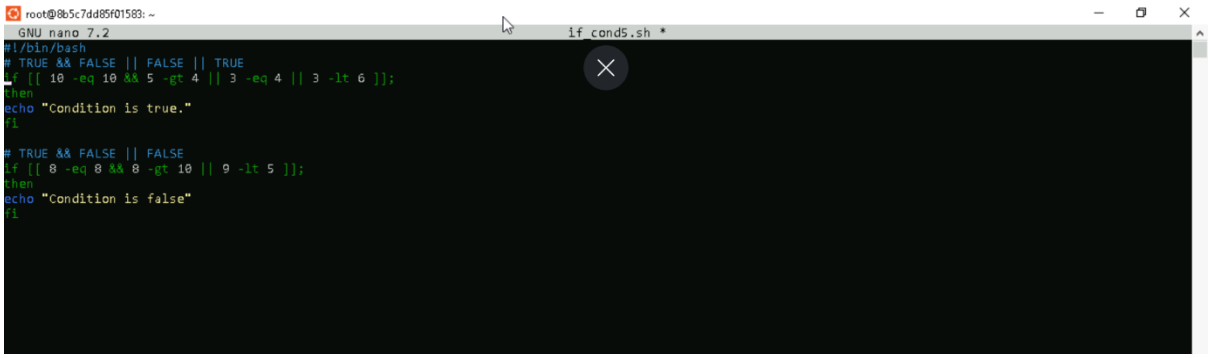
Step -1: Created a Script file using touch command as “if_cond5.sh”.

Step -2: Opened a nano file to write the Script.

Step -3: Changes the file permission.

```
root@8b5c7dd85f01583:~# touch if_cond5.sh
root@8b5c7dd85f01583:~# chmod +x if_cond5.sh
root@8b5c7dd85f01583:~# nano if_cond5.sh
```

The Nano Editor:



Output:

```
root@8b5c7dd85f01583:~# ./if_cond5.sh
Condition is true.
```

Nested if

10.Find the given number is greater than 50 and even number by using nested if expression.

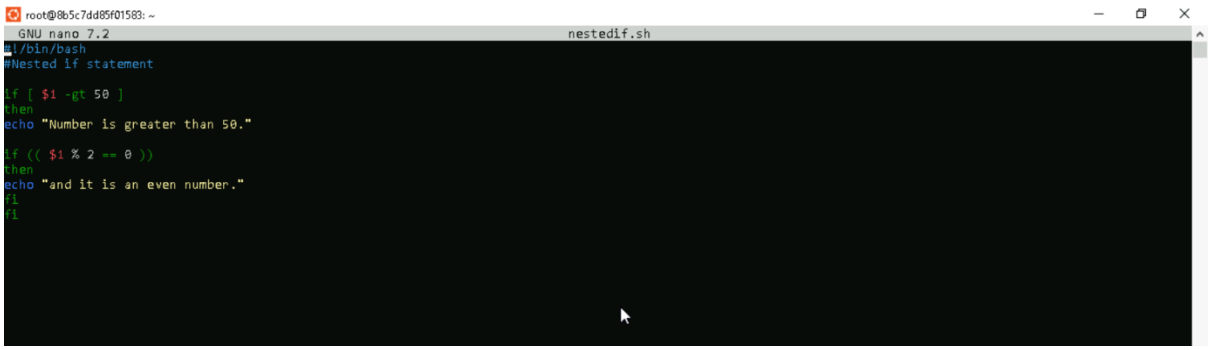
Step -1: Created a Script file using touch command as “nestedif.sh”.

Step -2: Opened a nano file to write the Script.

Step -3: Changes the file permission.

```
root@8b5c7dd85f01583:~# touch nestedif.sh
root@8b5c7dd85f01583:~# chmod +x nestedif.sh
root@8b5c7dd85f01583:~# nano nestedif.sh
```

The Nano Editor:



Output:

```
root@8b5c7dd85f01583:~# ./nestedif.sh 57
Number is greater than 50.
root@8b5c7dd85f01583:~#
```

Bash If Else Statement

11. Find the user input is greater than 3 for true condition and greater than 10 for false condition.

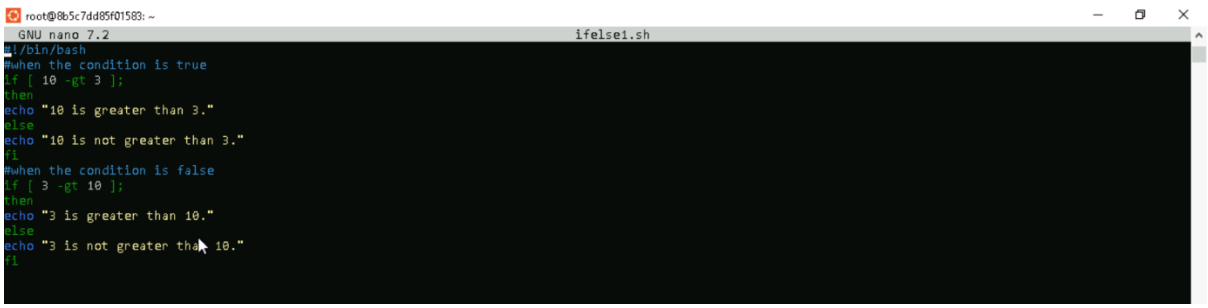
Step -1: Created a Script file using touch command as “ifelse1.sh”.

Step -2: Opened a nano file to write the Script.

Step -3: Changes the file permission.

```
root@8b5c7dd85f01583:~# touch ifelse1.sh
root@8b5c7dd85f01583:~# chmod +x ifelse1.sh
root@8b5c7dd85f01583:~# nano ifelse1.sh
```

The Nano Editor:



Output:

```
root@8b5c7dd85f01583:~# ./ifelse1.sh
10 is greater than 3.
3 is not greater than 10.
root@8b5c7dd85f01583:~#
```

12. Check whether the given condition is true or false.

Step -1: Created a Script file using touch command as “ifelse2.sh”.

Step -2: Opened a nano file to write the Script.

Step -3: Changes the file permission**.**

```
root@8b5c7dd85f01583:~# touch ifelse2.sh
root@8b5c7dd85f01583:~# chmod +x ifelse2.sh
root@8b5c7dd85f01583:~# nano ifelse2.sh
```

The Nano Editor:

Output:

```
root@8b5c7dd85f01583:~# ./ifelse2.sh
Given condition is true.
Given condition is not true.
root@8b5c7dd85f01583:~#
```

13. Demonstrating how to use if-else statement in a single line.

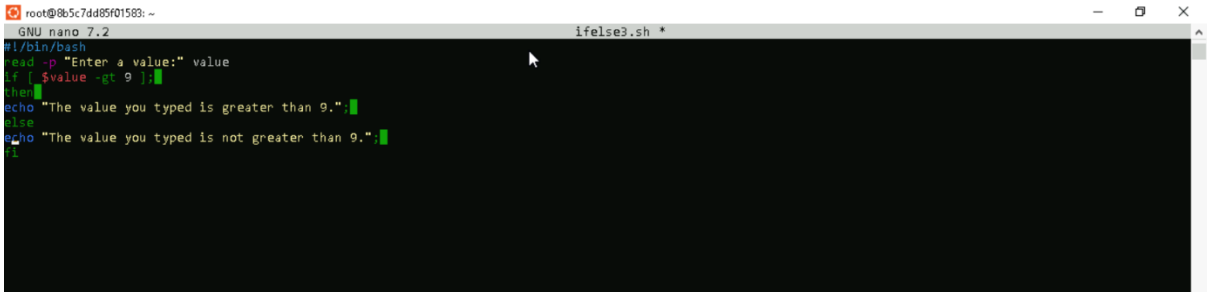
Step -1: Created a Script file using touch command as “ifelse3.sh”.

Step -2: Opened a nano file to write the Script.

Step -3: Changes the file permission.

```
root@8b5c7dd85f01583:~# touch ifelse3.sh
root@8b5c7dd85f01583:~# chmod +x ifelse3.sh
root@8b5c7dd85f01583:~# nano ifelse3.sh
```

The Nano Editor:



Output:

```
root@8b5c7dd85f01583:~# ./ifelse3.sh
Enter a value:26
The value you typed is greater than 9.
root@8b5c7dd85f01583:~#
```

14. How to make use of the nested if-else statement in bash.

Step -1: Created a Script file using touch command as “ifelse4.sh”.

Step -2: Opened a nano file to write the Script.

Step -3: Changes the file permission.

```
root@8b5c7dd85f01583: ~
The value you typed is greater than 9.
root@8b5c7dd85f01583:~# touch ifelse4.sh
root@8b5c7dd85f01583:~# chmod +x ifelse4.sh
root@8b5c7dd85f01583:~# nano ifelse4.sh
```

The Nano Editor:

Output:

```
root@8b5c7dd85f01583:~# ./ifelse4.sh
Enter a value:10
10>9, 10<11
```

Bash Else-if

15.Write a program to find the eligibility for discount using else-if condition.

Step -1: Created a Script file using touch command as “elseif1.sh”.

Step -2: Opened a nano file to write the Script.

Step -3: Changes the file permission.

```
root@8b5c7dd85f01583:~# touch elseif1.sh
root@8b5c7dd85f01583:~# chmod +x elseif1.sh
root@8b5c7dd85f01583:~# nano elseif1.sh
```

The Nano Editor:

```
root@8b5c7dd85f01583: ~
  GNU nano 7.2                                elseif1.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
```

Output:

```
root@8b5c7dd85f01583:~# ./elseif1.sh
Enter a number of quantity:110
Eligible for 10% discount
```

16. Write a program to find the eligibility for discount using else-if condition along with AND and OR Operators.

Step -1: Created a Script file using touch command as “elseif2.sh”.

Step -2: Opened a nano file to write the Script.

Step -3: Changes the file permission.

```
root@8b5c7dd85f01583:~# touch elseif2.sh
root@8b5c7dd85f01583:~# nano elseif2.sh
root@8b5c7dd85f01583:~# chmod +x elseif2.sh
```

The Nano Editor:

```
root@8b5c7dd85f01583: ~
  GNU nano 7.2                                elseif2.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
```

Output:

```
root@8b5c7dd85f01583:~# ./elseif2.sh
Enter a number of quantity:100
Lucky Draw Winner
Eligible to get the item for free
root@8b5c7dd85f01583:~#
```