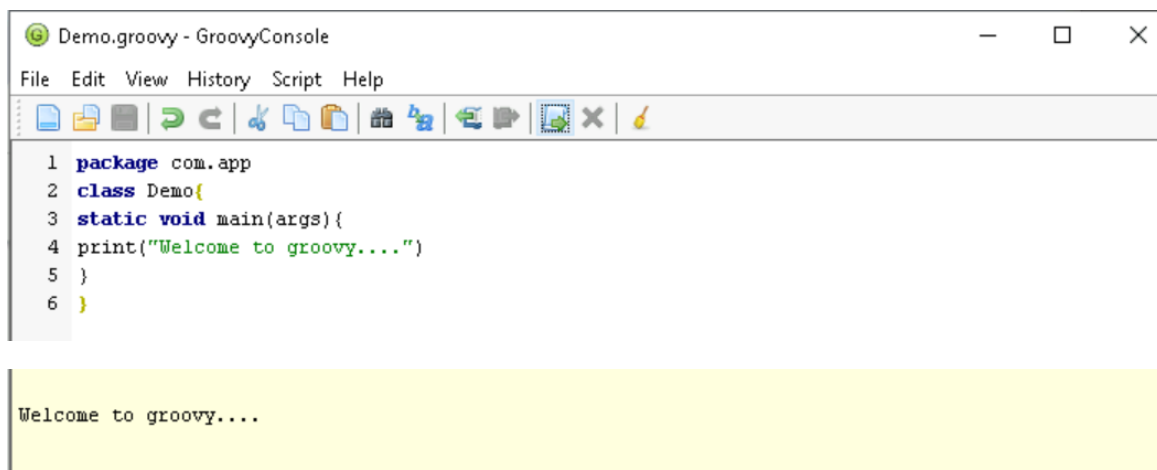


## First Groovy program:

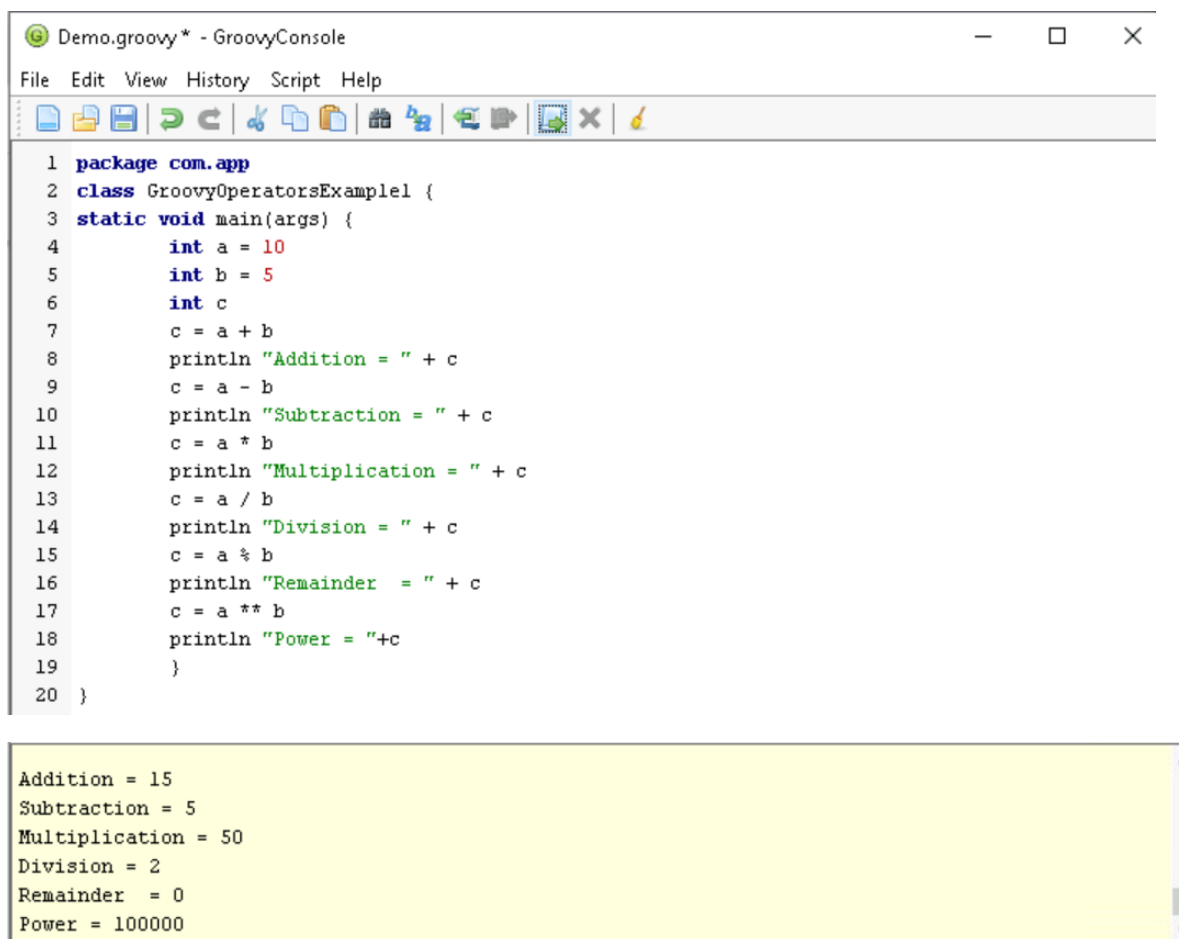


```
1 package com.app
2 class Demo{
3 static void main(args){
4 print("Welcome to groovy....")
5 }
6 }
```

Welcome to groovy....

## Groovy Operators:

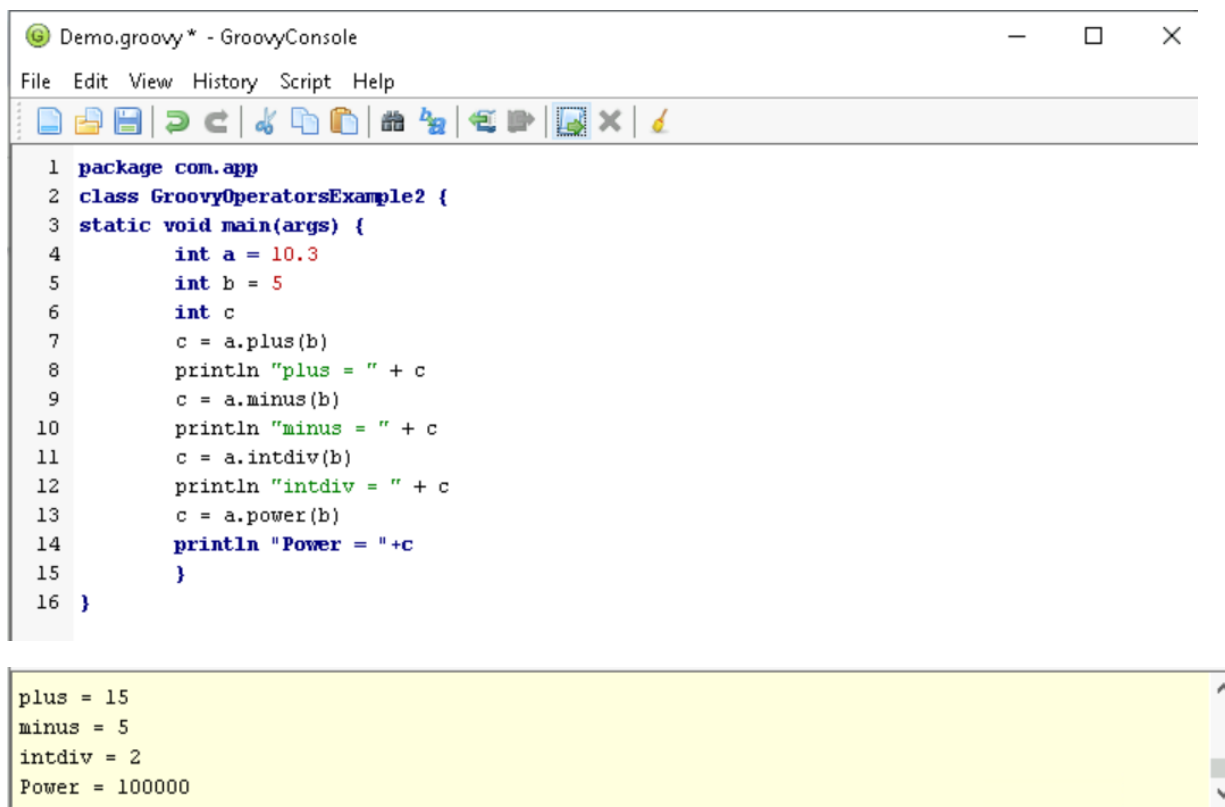
Example 1:



```
1 package com.app
2 class GroovyOperatorsExample1 {
3 static void main(args) {
4     int a = 10
5     int b = 5
6     int c
7     c = a + b
8     println "Addition = " + c
9     c = a - b
10    println "Subtraction = " + c
11    c = a * b
12    println "Multiplication = " + c
13    c = a / b
14    println "Division = " + c
15    c = a % b
16    println "Remainder = " + c
17    c = a ** b
18    println "Power = "+c
19 }
20 }
```

Addition = 15  
Subtraction = 5  
Multiplication = 50  
Division = 2  
Remainder = 0  
Power = 100000

## Example 2:



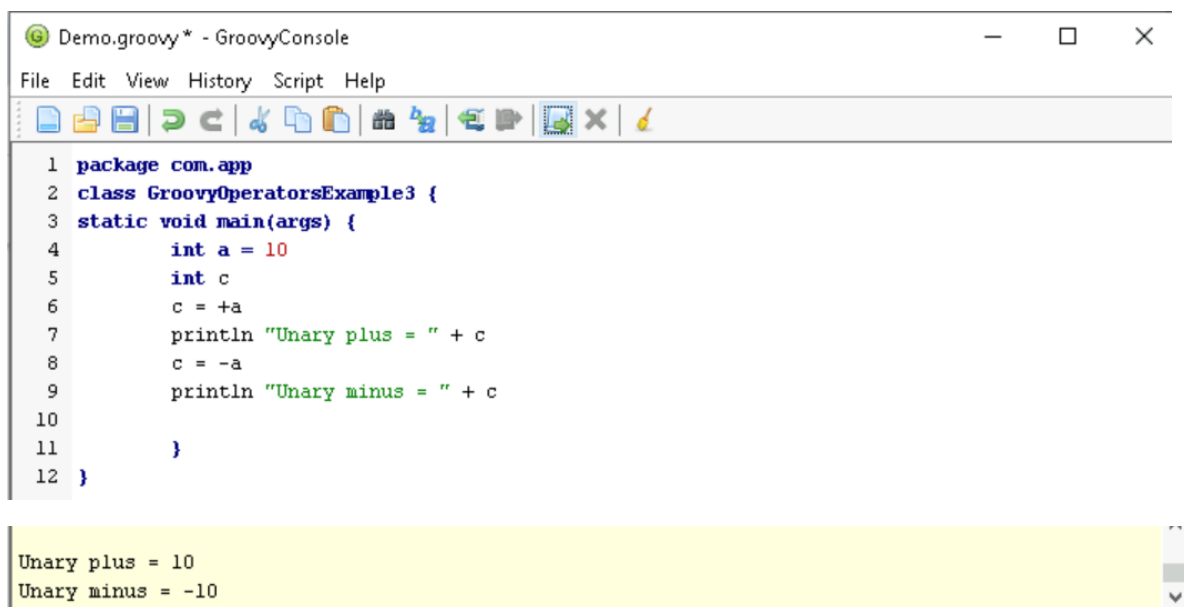
The screenshot shows a GroovyConsole window titled "Demo.groovy\* - GroovyConsole". The menu bar includes File, Edit, View, History, Script, and Help. The toolbar contains icons for file operations and execution. The code editor displays the following Groovy script:

```
1 package com.app
2 class GroovyOperatorsExample2 {
3     static void main(args) {
4         int a = 10.3
5         int b = 5
6         int c
7         c = a.plus(b)
8         println "plus = " + c
9         c = a.minus(b)
10        println "minus = " + c
11        c = a.intdiv(b)
12        println "intdiv = " + c
13        c = a.power(b)
14        println "Power = "+c
15    }
16 }
```

The output console at the bottom shows the results of the script execution:

```
plus = 15
minus = 5
intdiv = 2
Power = 100000
```

## Example 3:



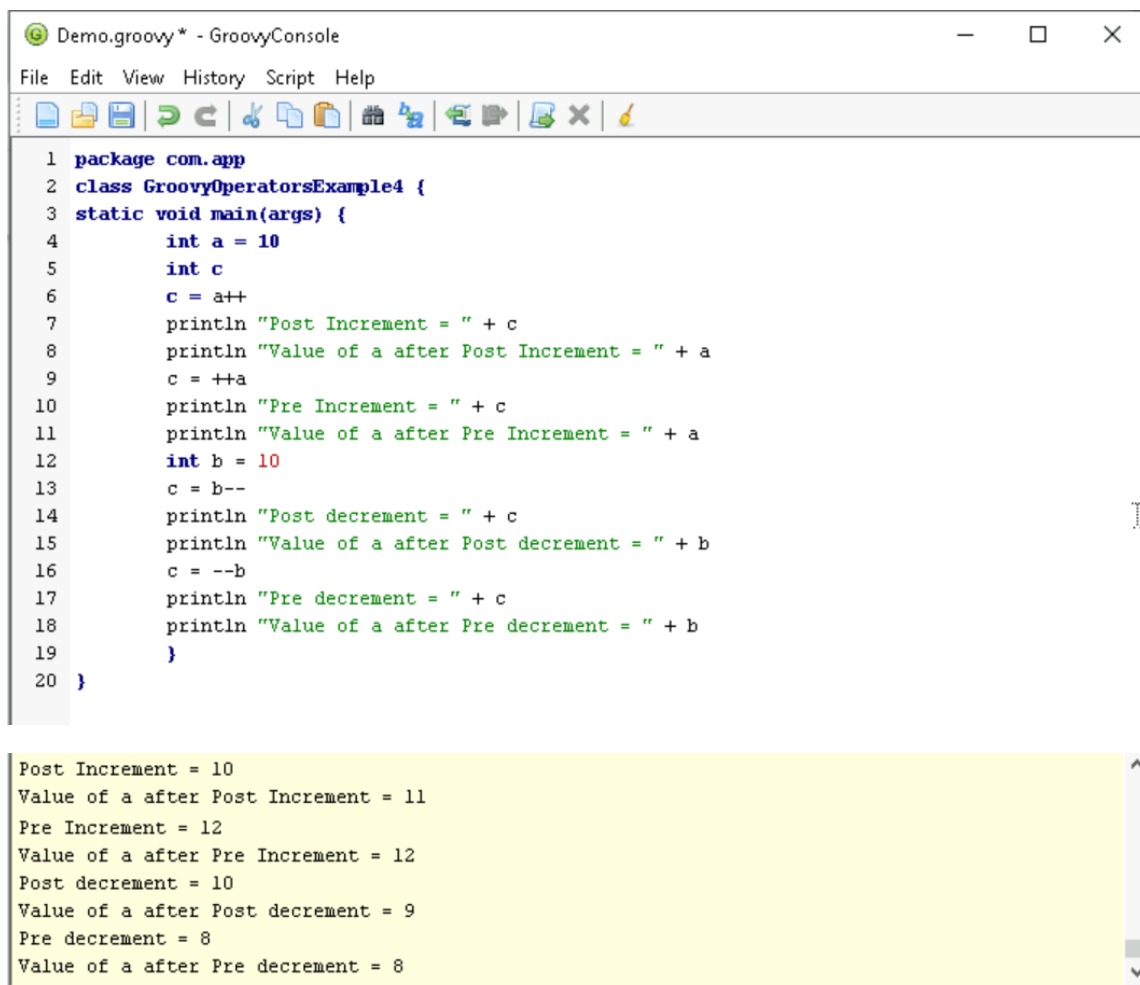
The screenshot shows a GroovyConsole window titled "Demo.groovy\* - GroovyConsole". The menu bar includes File, Edit, View, History, Script, and Help. The toolbar contains icons for file operations and execution. The code editor displays the following Groovy script:

```
1 package com.app
2 class GroovyOperatorsExample3 {
3     static void main(args) {
4         int a = 10
5         int c
6         c = +a
7         println "Unary plus = " + c
8         c = -a
9         println "Unary minus = " + c
10
11    }
12 }
```

The output console at the bottom shows the results of the script execution:

```
Unary plus = 10
Unary minus = -10
```

#### Example 4:

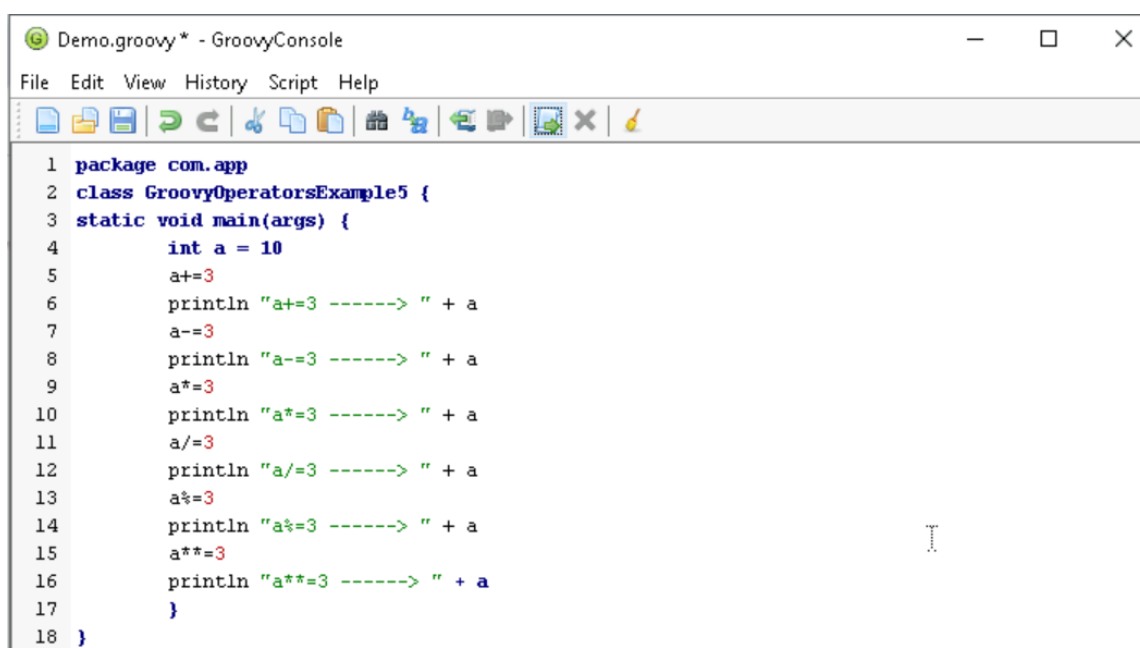


The screenshot shows a Groovy IDE window titled "Demo.groovy\* - GroovyConsole". The code defines a class `GroovyOperatorsExample4` with a `main` method. It demonstrates post-increment, pre-increment, post-decrement, and pre-decrement operations on variables `a` and `b`. The output console shows the results of these operations.

```
1 package com.app
2 class GroovyOperatorsExample4 {
3     static void main(args) {
4         int a = 10
5         int c
6         c = a++
7         println "Post Increment = " + c
8         println "Value of a after Post Increment = " + a
9         c = ++a
10        println "Pre Increment = " + c
11        println "Value of a after Pre Increment = " + a
12        int b = 10
13        c = b--
14        println "Post decrement = " + c
15        println "Value of a after Post decrement = " + b
16        c = --b
17        println "Pre decrement = " + c
18        println "Value of a after Pre decrement = " + b
19    }
20 }
```

Post Increment = 10  
Value of a after Post Increment = 11  
Pre Increment = 12  
Value of a after Pre Increment = 12  
Post decrement = 10  
Value of a after Post decrement = 9  
Pre decrement = 8  
Value of a after Pre decrement = 8

#### Example 5:



The screenshot shows a Groovy IDE window titled "Demo.groovy\* - GroovyConsole". The code defines a class `GroovyOperatorsExample5` with a `main` method. It demonstrates arithmetic operations on variable `a`, including addition, subtraction, multiplication, division, and modulus.

```
1 package com.app
2 class GroovyOperatorsExample5 {
3     static void main(args) {
4         int a = 10
5         a+=3
6         println "a+=3 -----> " + a
7         a-=3
8         println "a-=3 -----> " + a
9         a*=3
10        println "a*=3 -----> " + a
11        a/=3
12        println "a/=3 -----> " + a
13        a%=3
14        println "a%=3 -----> " + a
15        a**=3
16        println "a**=3 -----> " + a
17    }
18 }
```

```
a+=3 -----> 13
a-=3 -----> 10
a*=3 -----> 30
a/=3 -----> 10
a%=3 -----> 1
a**=3 -----> 1
```

Example 6:

```
1 package com.app
2 class GroovyOperatorsExample6 {
3     static void main(args) {
4         int a = 10
5         int b = 12
6         boolean c
7         println "a = 10"
8         println "b = 12"
9         c = a == b
10        println "Relational Operator equals [c = a == b] ----> " + c
11        c = a != b
12        println "Relational Operator different [c = a != b] ----> " + c
13        c = a < b
14        println "Relational Operator less than [c = a < b] ----> " + c
15        c = a <= b
16        println "Relational Operator less than equal to [c = a <= b] ----> " + c
17        c = a > b
18        println "Relational Operator greater than [c = a > b] ----> " + c
19        c = a >= b
20        println "Relational Operator greater than equal to [c = a >= b] ----> " + c
21
22    }
23 }
```

```
a = 10
b = 12
Relational Operator equals [c = a == b] ----> false
Relational Operator different [c = a != b] ----> true
Relational Operator less than [c = a < b] ----> true
Relational Operator less than equal to [c = a <= b] ----> true
Relational Operator greater than [c = a > b] ----> false
Relational Operator greater than equal to [c = a >= b] ----> false
```

Example 7:

```
1 package com.app
2 class GroovyOperatorsExample7 {
3     static void main(args) {
4         boolean c
5         c = true && true
6         println "Logical AND operator = " + c
7         c = true || false
8         println "Logical OR operator = " + c
9         c = !false
10        println "Logical NOT operator = " + c
11
12    }
13 }
```

```
Logical AND operator = true
Logical OR operator = true
Logical NOT operator = true
```

Example 8:

```
1 package com.app
2 class GroovyOperatorsExample8 {
3     static void main(args) {
4         boolean c
5         c = (!false && false)
6         println c
7     }
8 }
```

```
false
```

Example 9:

```
1 package com.app
2 class GroovyOperatorsExample1 {
3     static void main(args) {
4         boolean c
5         c = true || true && false
6         println c
7     }
8 }
```

```
true
```

Example 10:

```
1 package com.app
2 class GroovyOperatorsExample10 {
3
4     static void main(args) {
5         int a = 0b00101111
6         println "a = 0b00101111 ----> "+a
7         int b = 0b000010101
8         println "b = 0b000010101 ----> "+b
9         println "(a & a) ----> "+(a & a)
10        println "(a & b) ----> "+(a & b)
11        println "(a | a) ----> "+(a | a)
12        println "(a | a) ----> "+(a | b)
13
14        int c = 0b11111111
15        println "c = 0b11111111"
16        println "((a ^ a) & c) ----> "+((a ^ a) & c)
17        println "((a ^ b) & c) ----> "+((a ^ b) & c)
18        println "((~a) & c) ----> "+((~a) & c)
19    }
20 }
```

```

a = 0b00101111 ----> 47
b = 0b000010101 ----> 21
(a & a) ----> 47
(a & b) ----> 5
(a | a) ----> 47
(a | a) ----> 63
c = 0b11111111
((a ^ a) & c) ----> 0
((a ^ b) & c) ----> 58
((~a) & c) ----> 208

```

Example 11:

```

1 package com.app
2 class GroovyOperatorsExample11 {
3 static void main(args) {
4     int a = 23
5     int b = 43
6     println "Converting Integer to Binary a = 23 ----> " + Integer.toBinaryString(a)
7     println "Converting Integer to Binary b = 43 ----> " + Integer.toBinaryString(b)
8     println "Converting binary to integer 10111 ----> a = " + Integer.parseInt("10111", 2)
9     println "Converting binary to integer 101011 ----> b = " + Integer.parseInt("101011", 2)
10 }
11 }

```

```

Converting Integer to Binary a = 23 ----> 10111
Converting Integer to Binary b = 43 ----> 101011
Converting binary to integer 10111 ----> a = 23
Converting binary to integer 101011 ----> b = 23

```

Example 12:

```

1 package com.app
2 class GroovyOperatorsExample12 {
3 static void main(args) {
4     println "(!true) ----> "+(!true)
5     println "(!'javatpoint') ----> "+(!'javatpoint')
6     println "(!Null) ----> "+(!'')
7 }
8 }

```

```

(!true) ----> false
(!'javatpoint') ----> false
!Null ----> true

```

Example 13:

```

1 package com.app
2 class GroovyOperatorsExample13 {
3 static void main(args)
4 {
5     String Answer
6     String s = 'javatpoint'
7     Answer = (s!=null && s.length()>0) ? 'Found' : 'Not found'
8     print(Answer)
9 }
10 }

```

Found

Example 14:

```
1 package com.app
2 class GroovyOperatorsExample1 {
3     static void main(args) {
4         String Answer
5         String s = 'javatpoint'
6         Answer = s ? 'Found' : 'Not Found'
7         println(Answer)
8
9         Answer = s ?: 'Found'
10        println(Answer)
11    }
12 }
```

```
Found
javatpoint
```