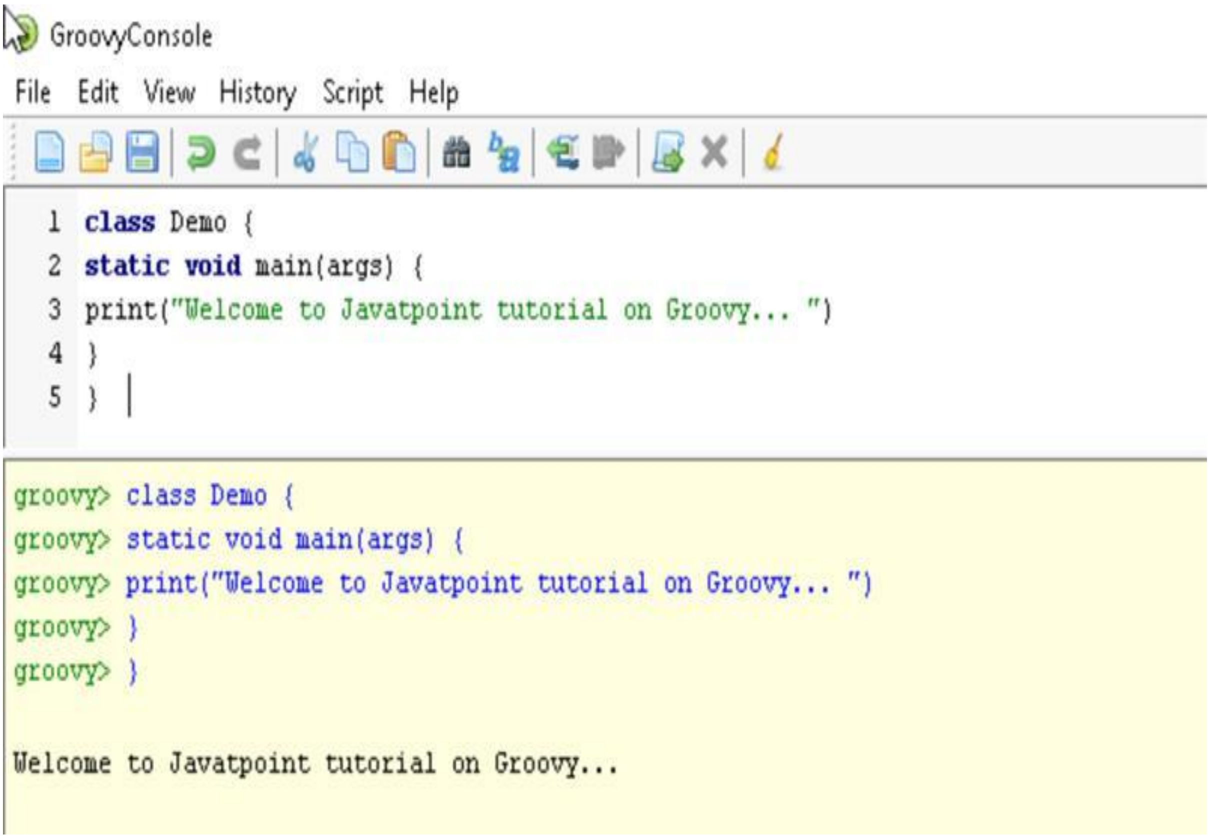
Groovy Basic Syntax



```
class Demo {
static void main(args) {
print("Welcome to Javatpoint tutorial on Groovy... ")
}
}
```

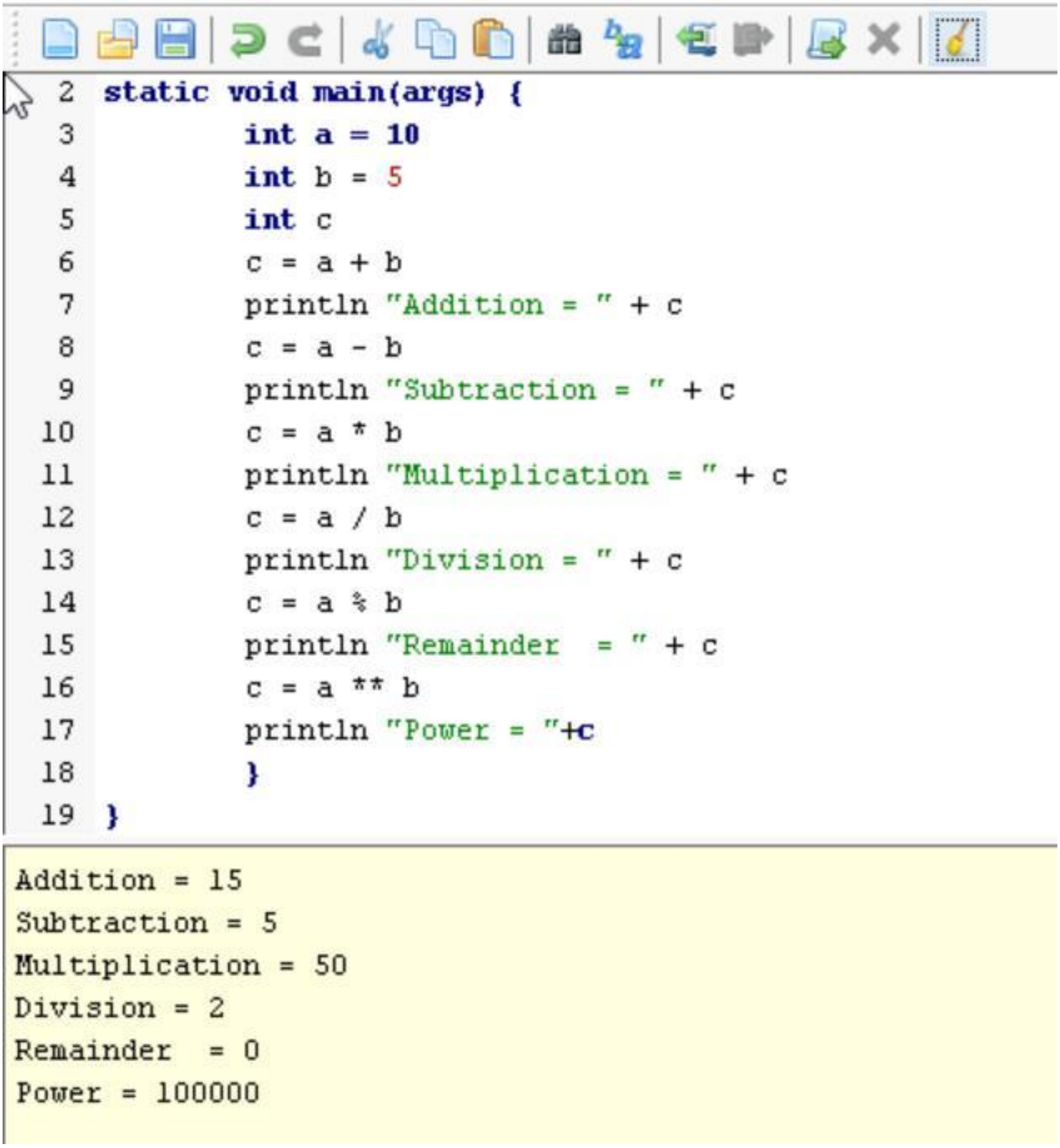
```
groovy> class Demo {
groovy> static void main(args) {
groovy> print("Welcome to Javatpoint tutorial on Groovy... ")
groovy> }
groovy> }

Welcome to Javatpoint tutorial on Groovy...
```

GROOVY OPERATORS

In groovy, operators are symbols which are used to tell the compiler to perform specified operations. Types of operators are:

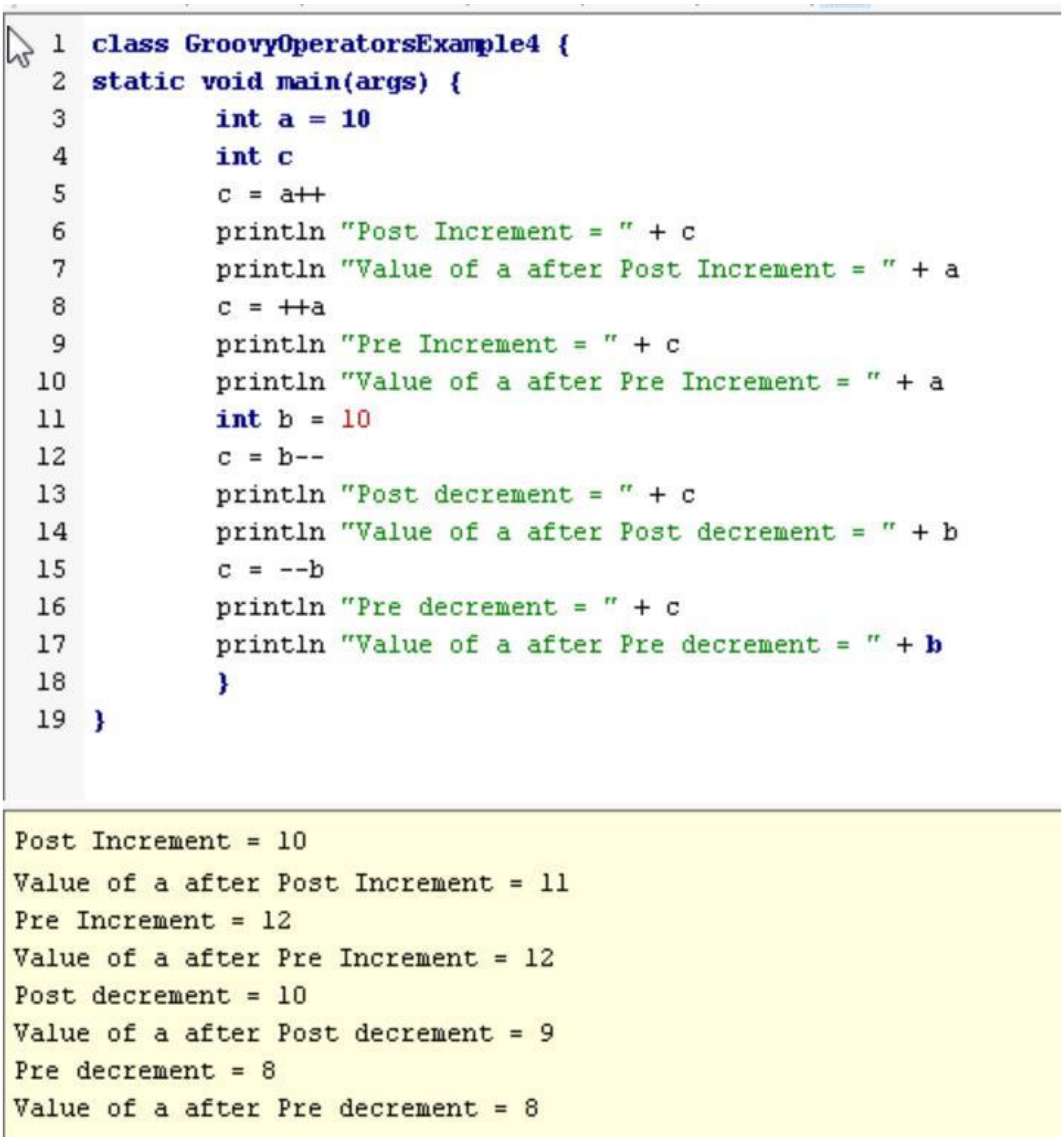
1. **Arithmetic operators**



```
static void main(args) {
        int a = 10
        int b = 5
        int c
        c = a + b
        println "Addition = " + c
        c = a - b
        println "Subtraction = " + c
        c = a * b
        println "Multiplication = " + c
        c = a / b
        println "Division = " + c
        c = a % b
        println "Remainder  = " + c
        c = a ** b
        println "Power = "+c
        }
}
```

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

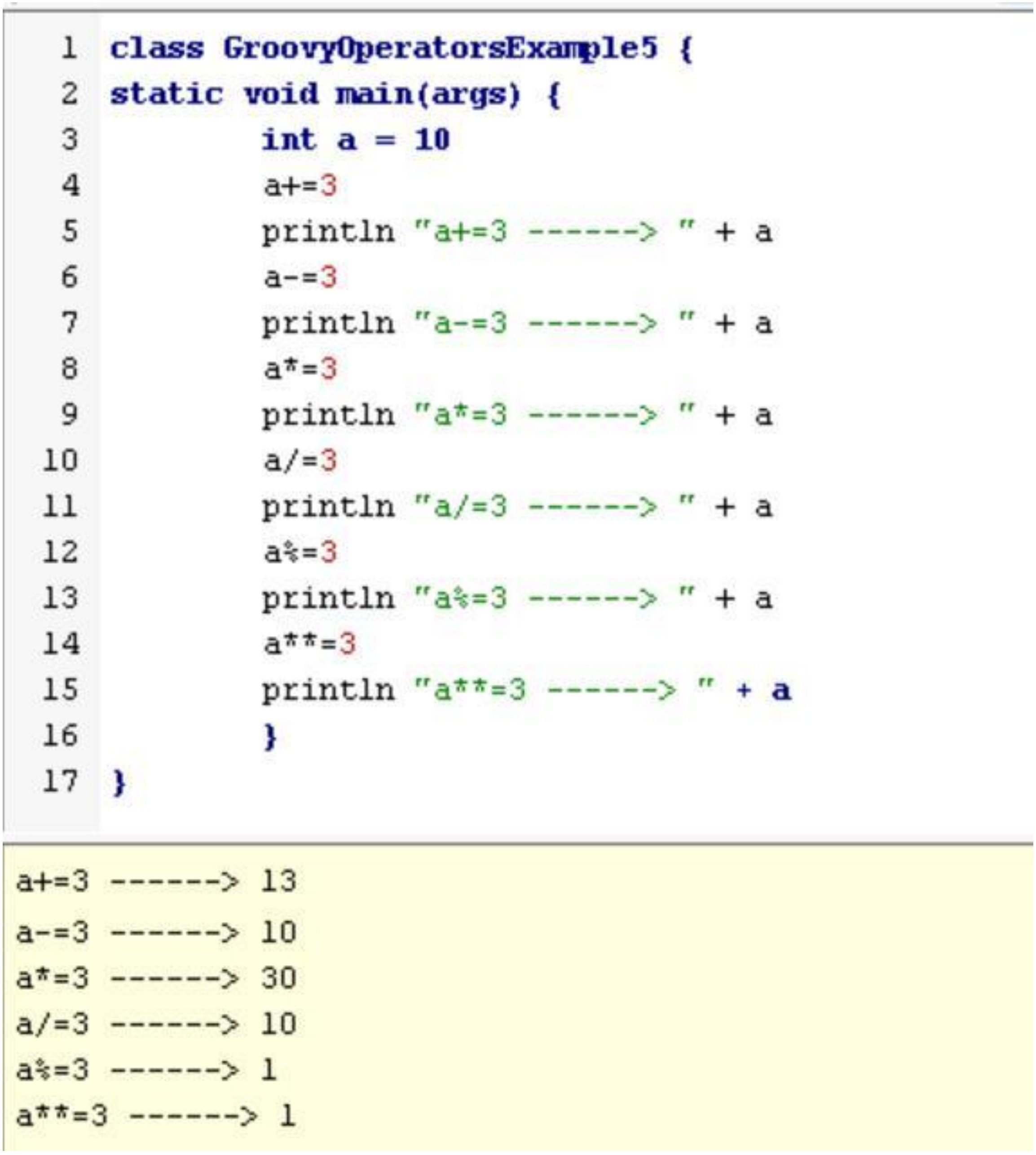
2. **Unary operators**



```
class GroovyOperatorsExample4 {
static void main(args) {
        int a = 10
        int c
        c = a++
        println "Post Increment = " + c
        println "Value of a after Post Increment = " + a
        c = ++a
        println "Pre Increment = " + c
        println "Value of a after Pre Increment = " + a
        int b = 10
        c = b--
        println "Post decrement = " + c
        println "Value of a after Post decrement = " + b
        c = --b
        println "Pre decrement = " + c
        println "Value of a after Pre decrement = " + b
        }
}
```

```
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
```

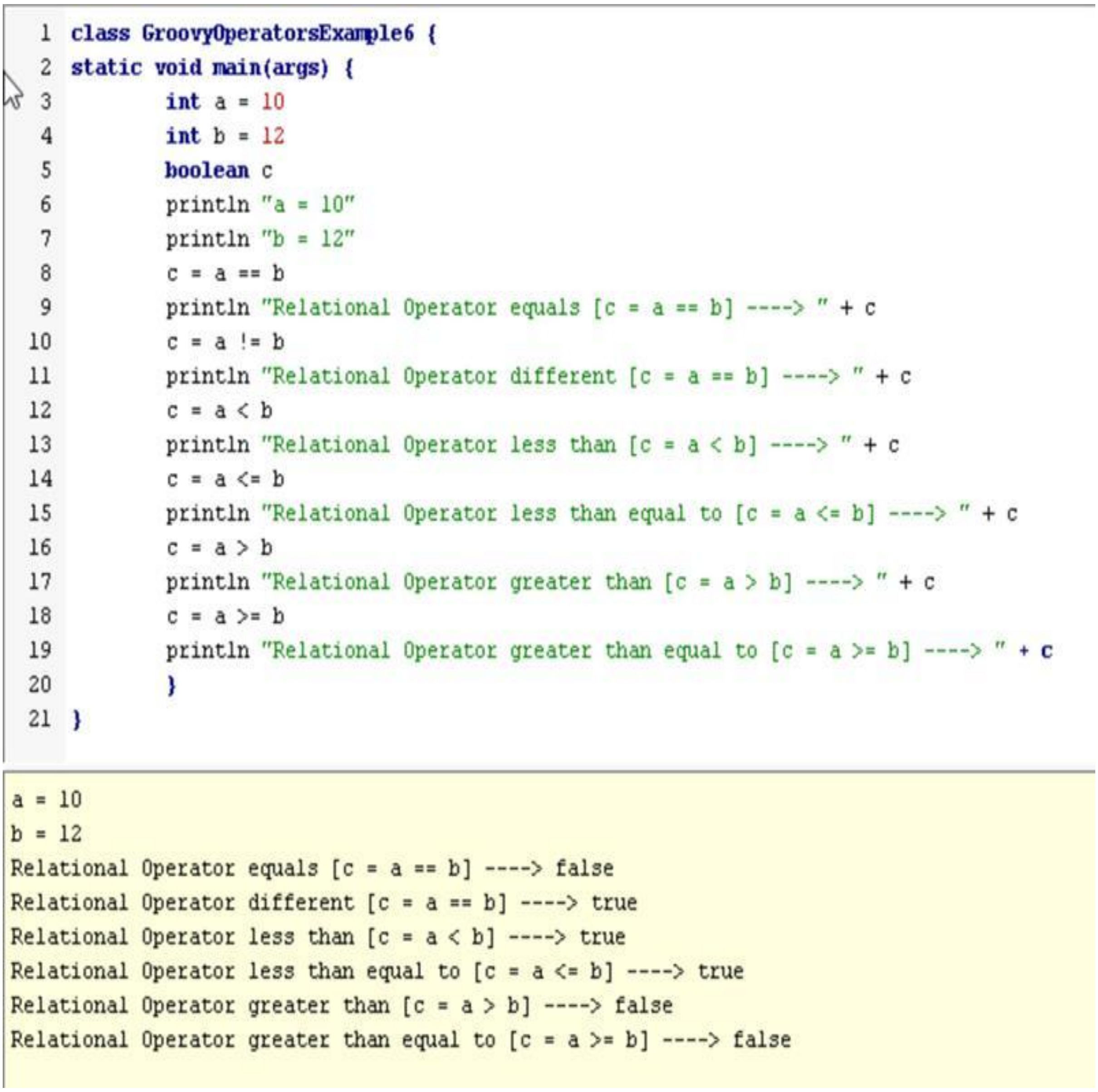
3. **Assignment arithmetic operators**



```
class GroovyOperatorsExample5 {
static void main(args) {
        int a = 10
        a+=3
        println "a+=3 ------> " + a
        a-=3
        println "a-=3 ------> " + a
        a*=3
        println "a*=3 ------> " + a
        a/=3
        println "a/=3 ------> " + a
        a%=3
        println "a%=3 ------> " + a
        a**=3
        println "a**=3 ------> " + a
        }
}
```

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

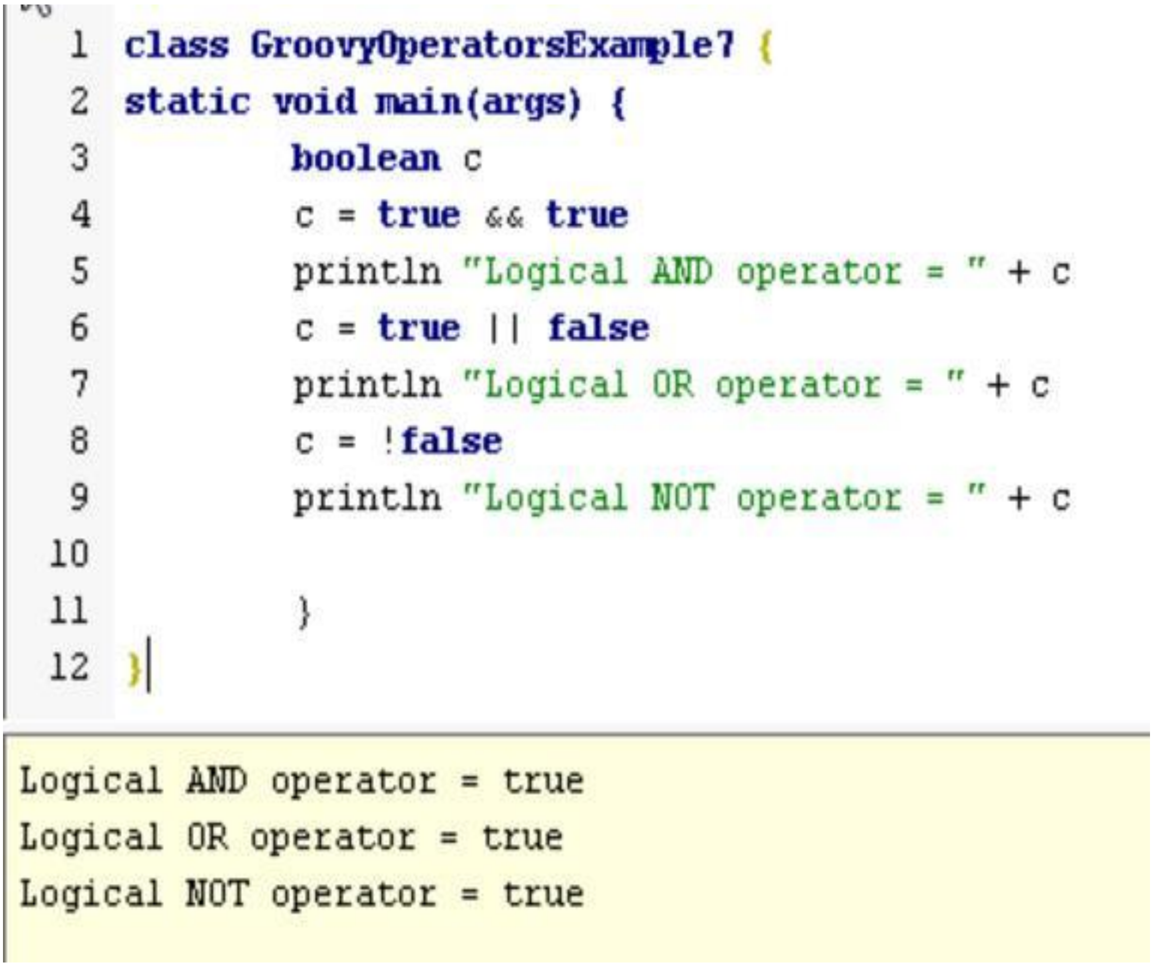
4. **Relational operators**



```
class GroovyOperatorsExample6 {
static void main(args) {
        int a = 10
        int b = 12
        boolean c
        println "a = 10"
        println "b = 12"
        c = a == b
        println "Relational Operator equals [c = a == b] ----> " + c
        c = a != b
        println "Relational Operator different [c = a == b] ----> " + c
        c = a < b
        println "Relational Operator less than [c = a < b] ----> " + c
        c = a <= b
        println "Relational Operator less than equal to [c = a <= b] ----> " + c
        c = a > b
        println "Relational Operator greater than [c = a > b] ----> " + c
        c = a >= b
        println "Relational Operator greater than equal to [c = a >= b] ----> " + c
        }
}
```

```
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
```

5. **Logical operators**

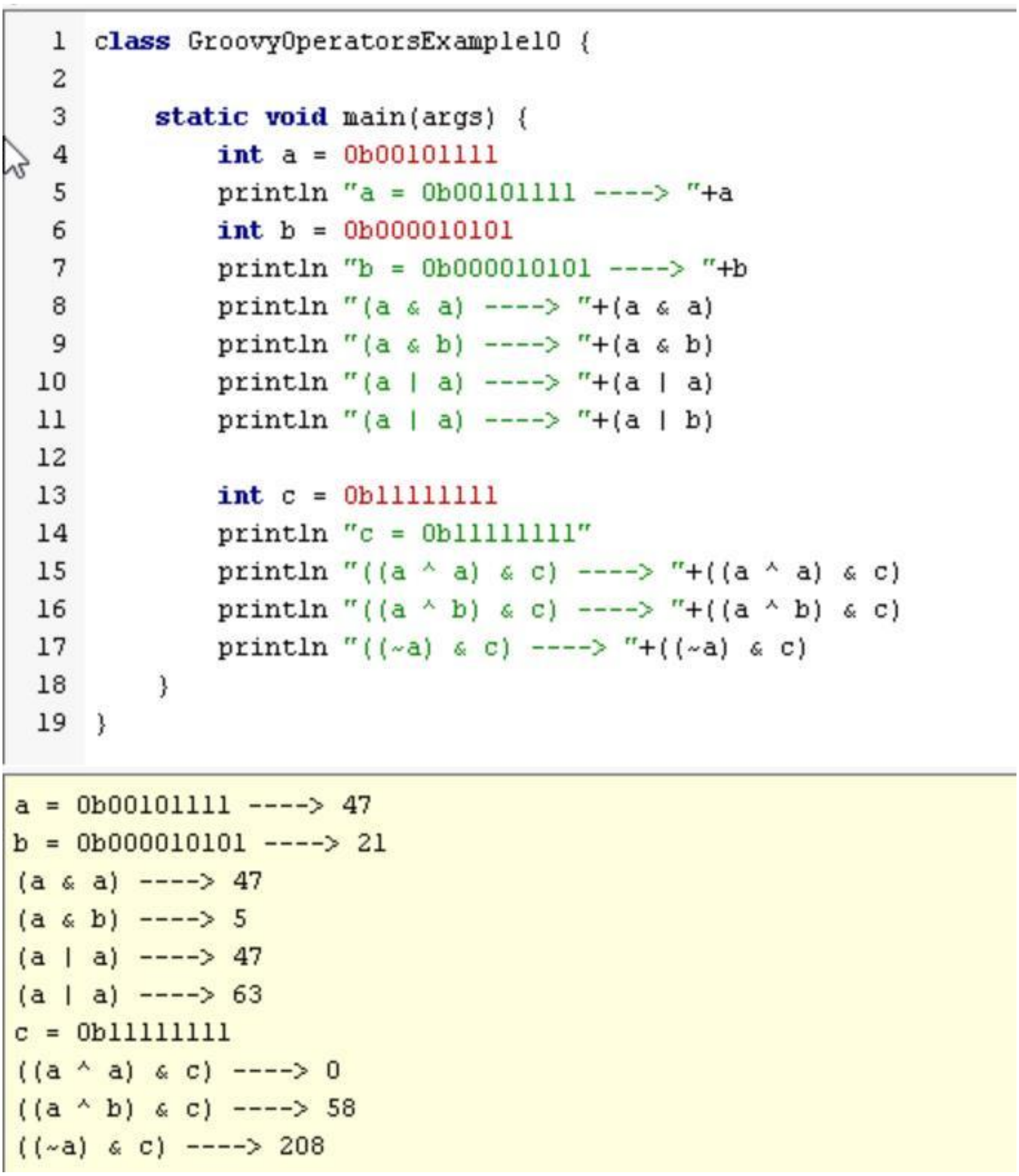


```
class GroovyOperatorsExample7 {
static void main(args) {
        boolean c
        c = true && true
        println "Logical AND operator = " + c
        c = true || false
        println "Logical OR operator = " + c
        c = !false
        println "Logical NOT operator = " + c

        }
}
```

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

6. **Bitwise operators**



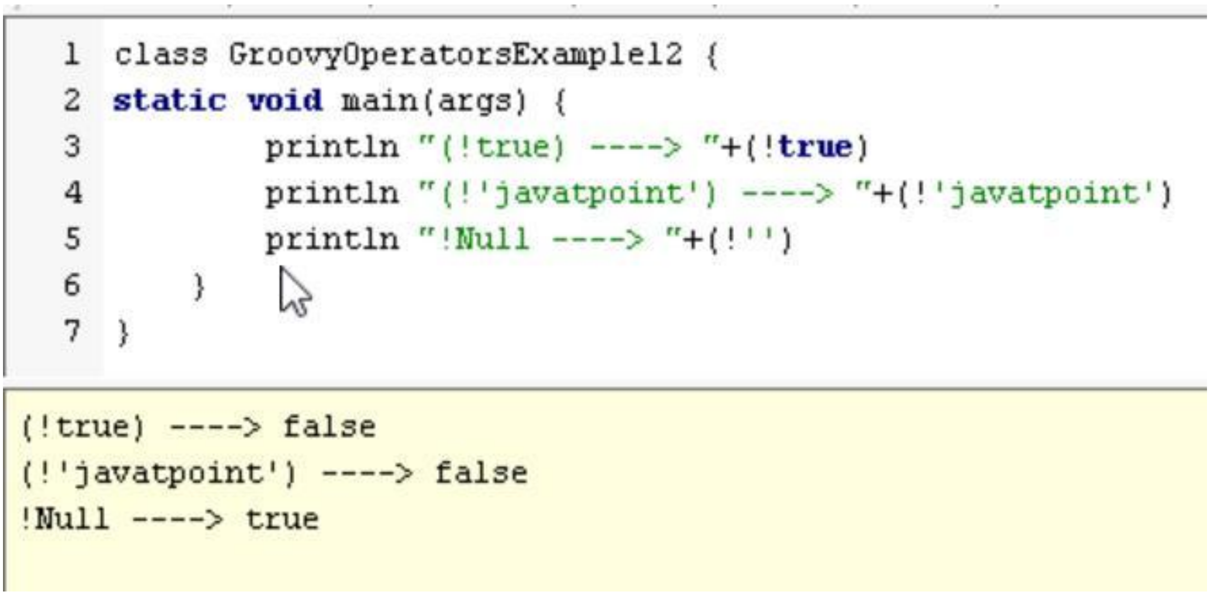
```
class GroovyOperatorsExample10 {

    static void main(args) {
        int a = 0b00101111
        println "a = 0b00101111 ----> "+a
        int b = 0b000010101
        println "b = 0b000010101 ----> "+b
        println "(a & a) ----> "+(a & a)
        println "(a & b) ----> "+(a & b)
        println "(a | a) ----> "+(a | a)
        println "(a | a) ----> "+(a | b)

        int c = 0b11111111
        println "c = 0b11111111"
        println "((a ^ a) & c) ----> "+((a ^ a) & c)
        println "((a ^ b) & c) ----> "+((a ^ b) & c)
        println "((~a) & c) ----> "+((~a) & c)
    }
}
```

```
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
```

7. **Conditional operators**



```
class GroovyOperatorsExample12 {
static void main(args) {
        println "(!true) ----> "+(!true)
        println "(!'javatpoint') ----> "+(!'javatpoint')
        println "!Null ----> "+(!'')
    }
}
```

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