Day 13   30-Aug-2025   Saturday.

Operatols

Arithmetic operatols

Addition	+	→ Add
Subtraction	−	→ Sub
Multiplication	*	→ Multi
division	/	→ quotient
Modulus	%	→ remainer

Exponentiation	**	} not in Java (python)
Floor division	//	

Relational operatols (Boolean output)

Equal to	==
Not Equal to	!=
Greater than	>
less than	<
greater than or equal to	>=
less than or equal to	<=

Logical operator

AND	&&	true && true
OR	||	true || false

NOT	!	true = false
		false = true

Bitwise operator

AND		*
0	0	0
0	1	0
1	0	0
1	1	0

⇒ AND &

Eg: 5 & 2

101 & 010

000 ⇒ $\boxed{0}$

OR		+
0	0	0
0	1	1
1	0	1
1	1	1

⇒ OR |

Eg: 5 | 2

101 | 010

111 ⇒ $\boxed{7}$

$\boxed{8\ 4\ 2\ 1}$

XOR		
0	0	0
0	1	1
1	0	1
1	1	0

⇒ XOR ^

Eg: 5 ^ 2

101 ^ 010 ⇒ ϕϕϕ

$\boxed{7}$

⇒ Not ~

⇒ Left shift <<

⇒ Right shift >>

Eg: Left shift.

5 << 2

101 ~~010~~

~~101~~ ~~101~~

~~010~~ ~~0 10~~

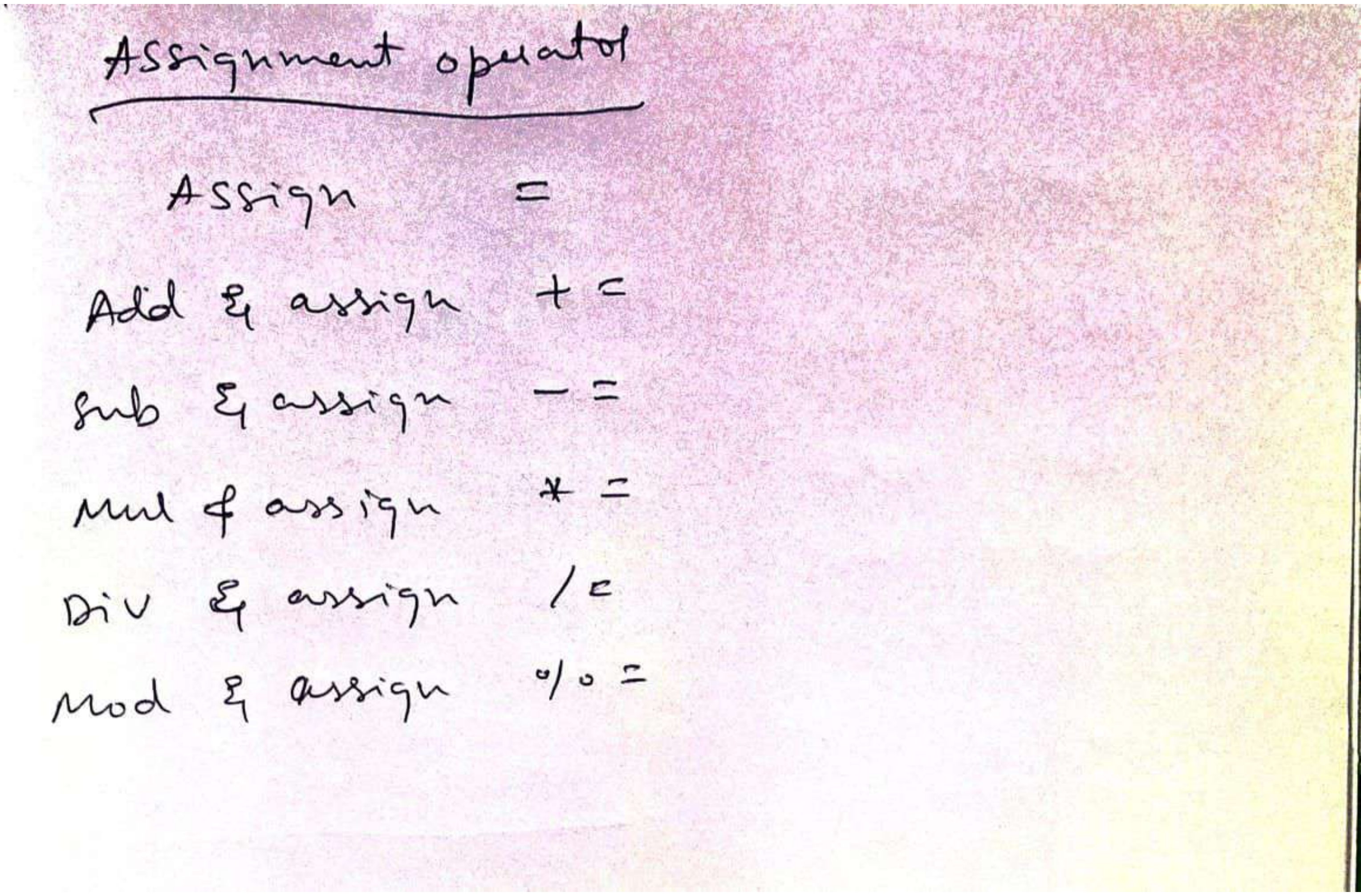
~~1010~~

```
    1 0 1
  1 0 1
1 0 1
---------
1 0 1 0 0  ⇒ 20
```

Eg: Right shift

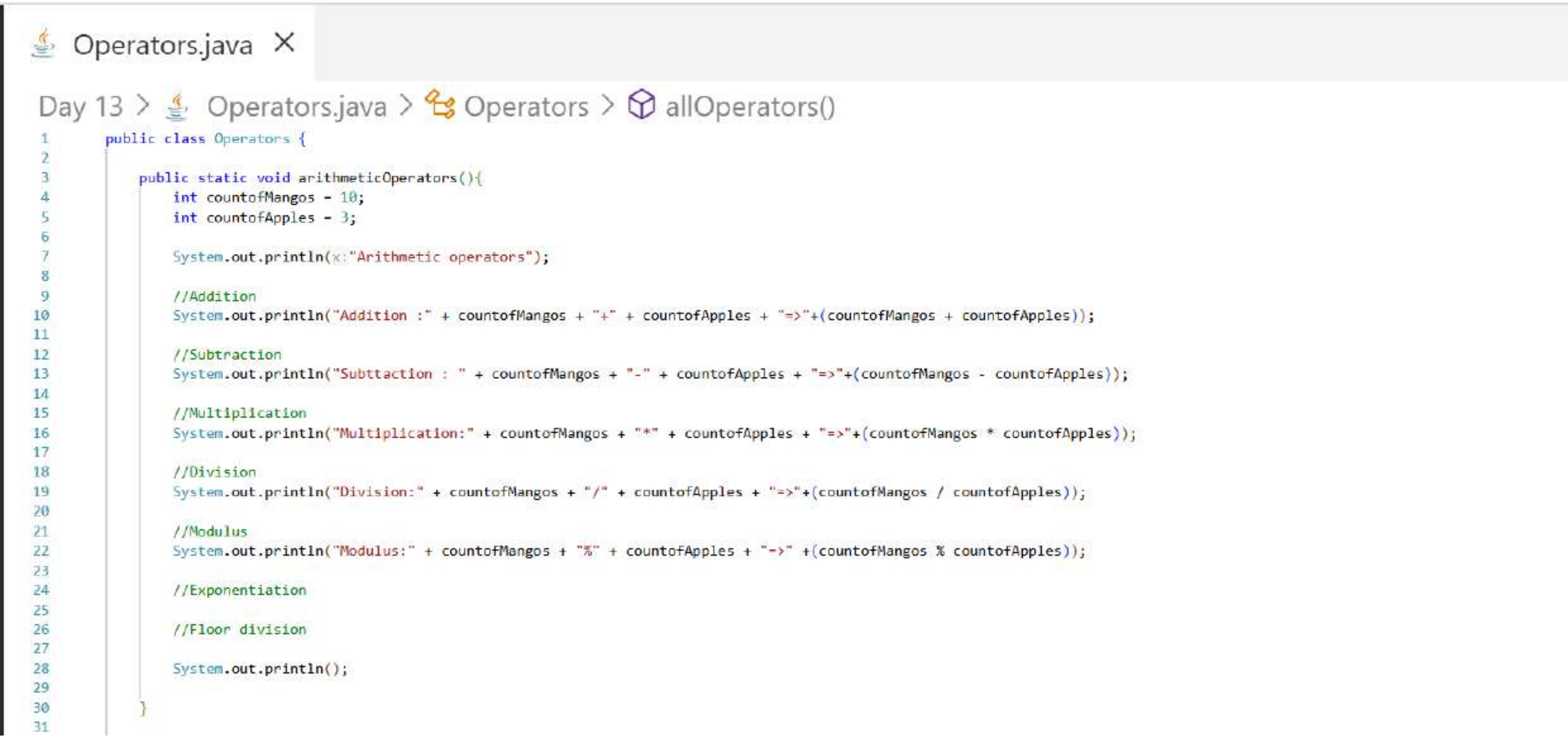
5 >> 2 ⇒ 1.

```
1 0 1
  1 0 | 1 |  x
    1 | 0 1 | .
---------------
0 0 1 ~~0~~
```

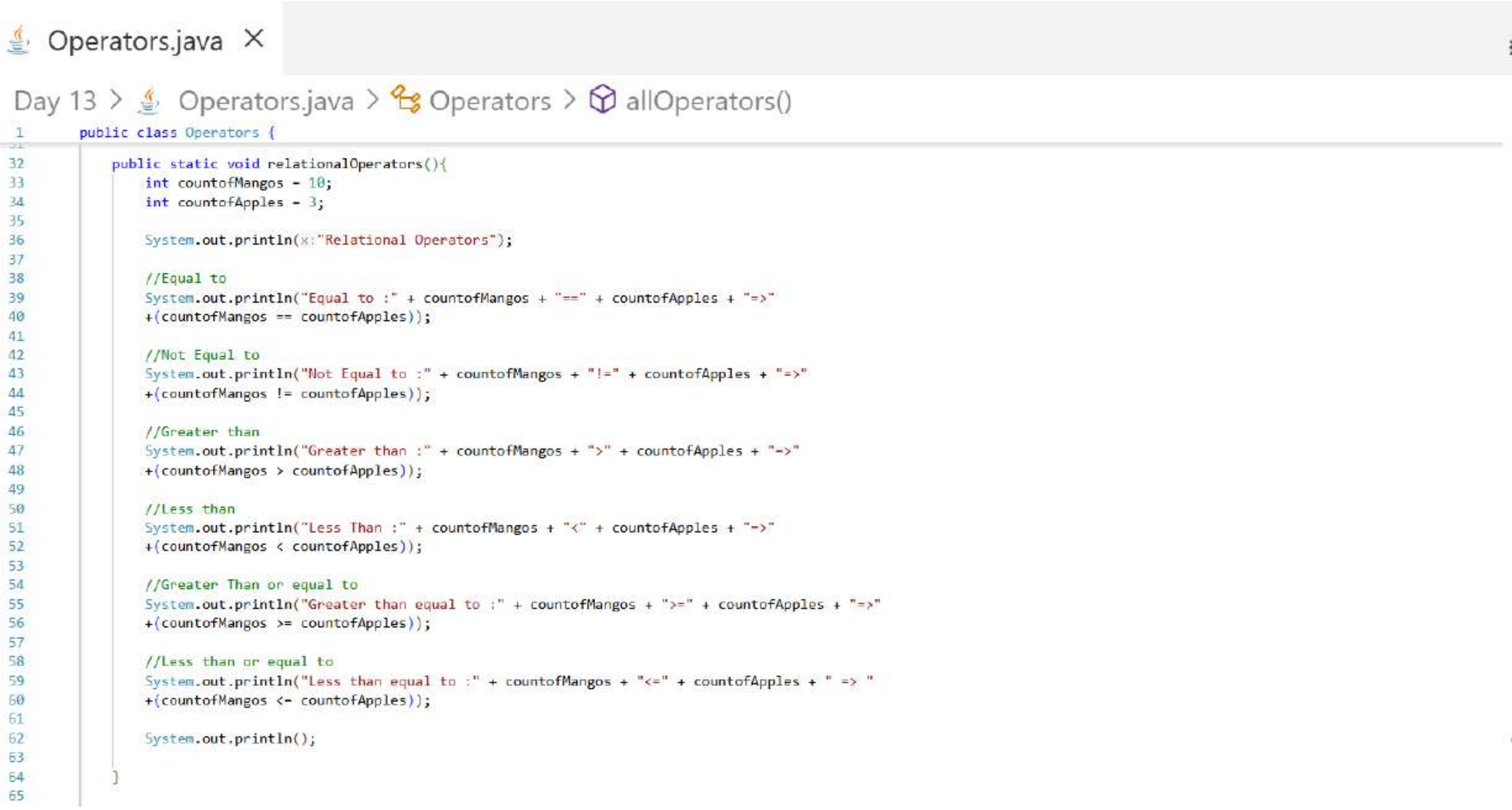


Assignment operatol

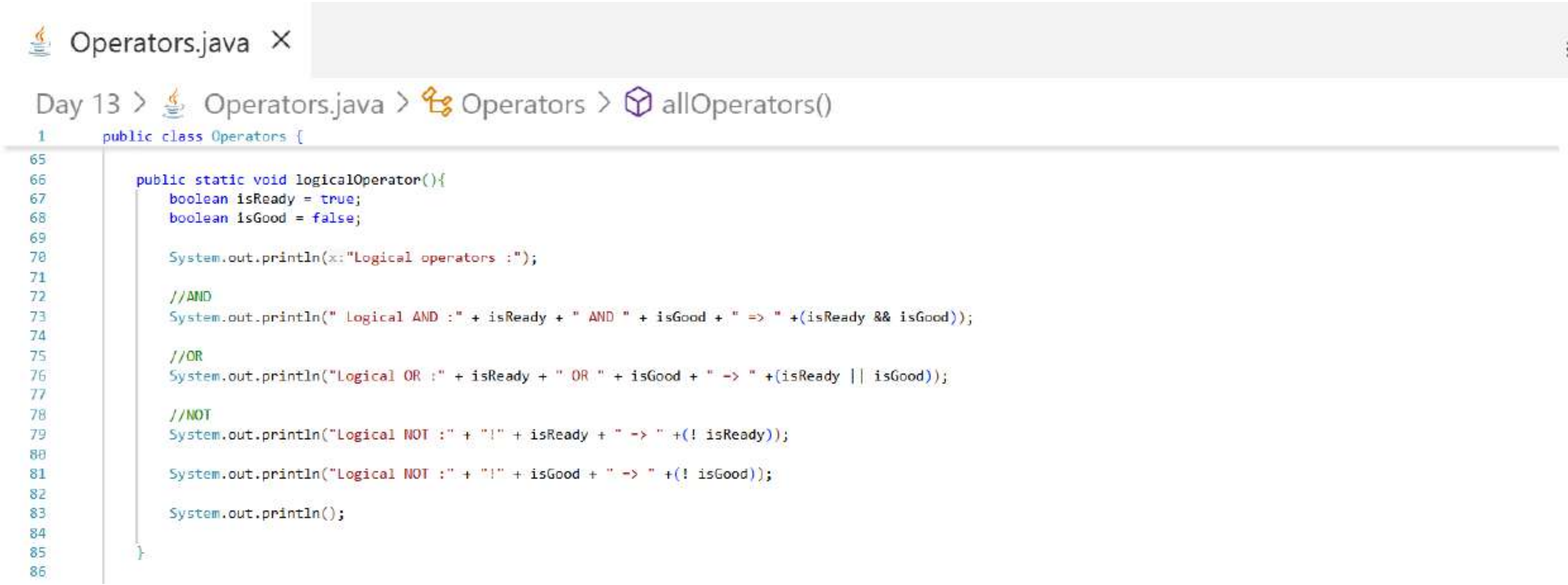
Assign	=
Add & assign	+=
sub & assign	-=
Mul & assign	*=
Div & assign	/=
Mod & assign	%=



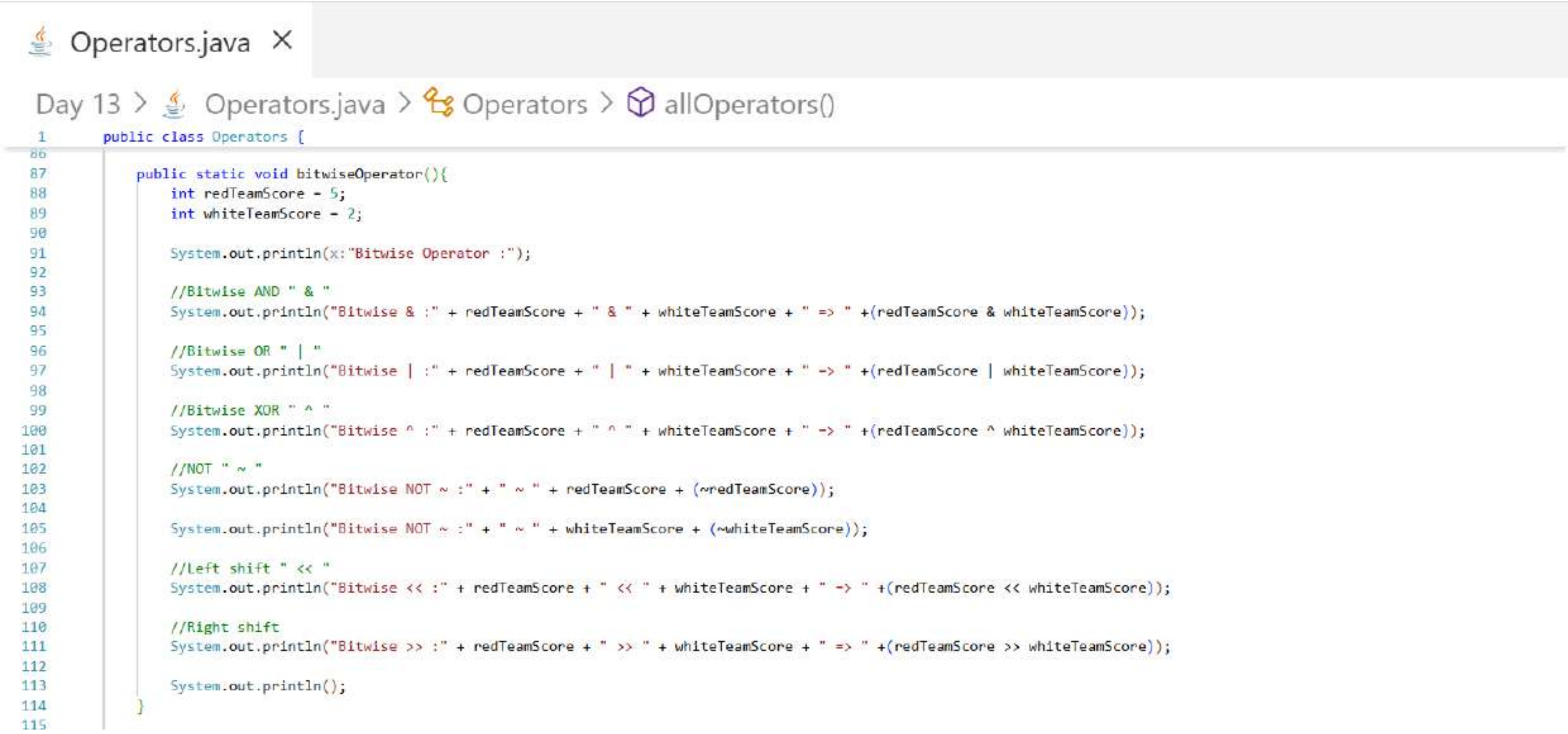
```java
public class Operators {

    public static void arithmeticOperators(){
        int countofMangos = 10;
        int countofApples = 3;

        System.out.println(x:"Arithmetic operators");

        //Addition
        System.out.println("Addition :" + countofMangos + "+" + countofApples + "=>"+(countofMangos + countofApples));

        //Subtraction
        System.out.println("Subttaction : " + countofMangos + "-" + countofApples + "=>"+(countofMangos - countofApples));

        //Multiplication
        System.out.println("Multiplication:" + countofMangos + "*" + countofApples + "=>"+(countofMangos * countofApples));

        //Division
        System.out.println("Division:" + countofMangos + "/" + countofApples + "=>"+(countofMangos / countofApples));

        //Modulus
        System.out.println("Modulus:" + countofMangos + "%" + countofApples + "->" +(countofMangos % countofApples));

        //Exponentiation

        //Floor division

        System.out.println();

    }

```



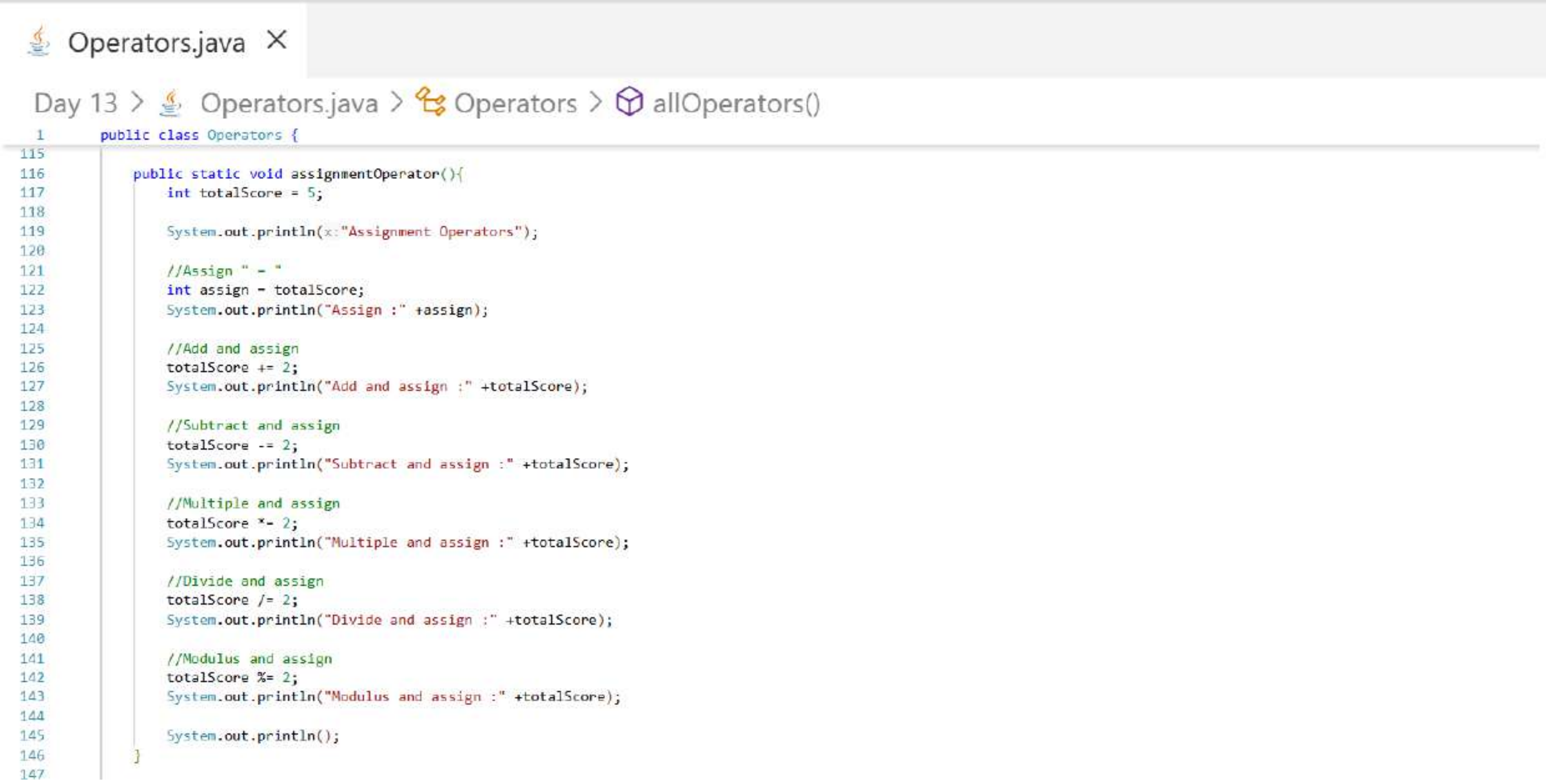
```java
public class Operators {

    public static void relationalOperators(){
        int countofMangos - 10;
        int countofApples - 3;

        System.out.println(x:"Relational Operators");

        //Equal to
        System.out.println("Equal to :" + countofMangos + "==" + countofApples + "=>"
        +(countofMangos == countofApples));

        //Not Equal to
        System.out.println("Not Equal to :" + countofMangos + "!=" + countofApples + "=>"
        +(countofMangos != countofApples));

        //Greater than
        System.out.println("Greater than :" + countofMangos + ">" + countofApples + "->"
        +(countofMangos > countofApples));

        //Less than
        System.out.println("Less Than :" + countofMangos + "<" + countofApples + "->"
        +(countofMangos < countofApples));

        //Greater Than or equal to
        System.out.println("Greater than equal to :" + countofMangos + ">=" + countofApples + "=>"
        +(countofMangos >= countofApples));

        //Less than or equal to
        System.out.println("Less than equal to :" + countofMangos + "<=" + countofApples + " => "
        +(countofMangos <- countofApples));

        System.out.println();

    }

```



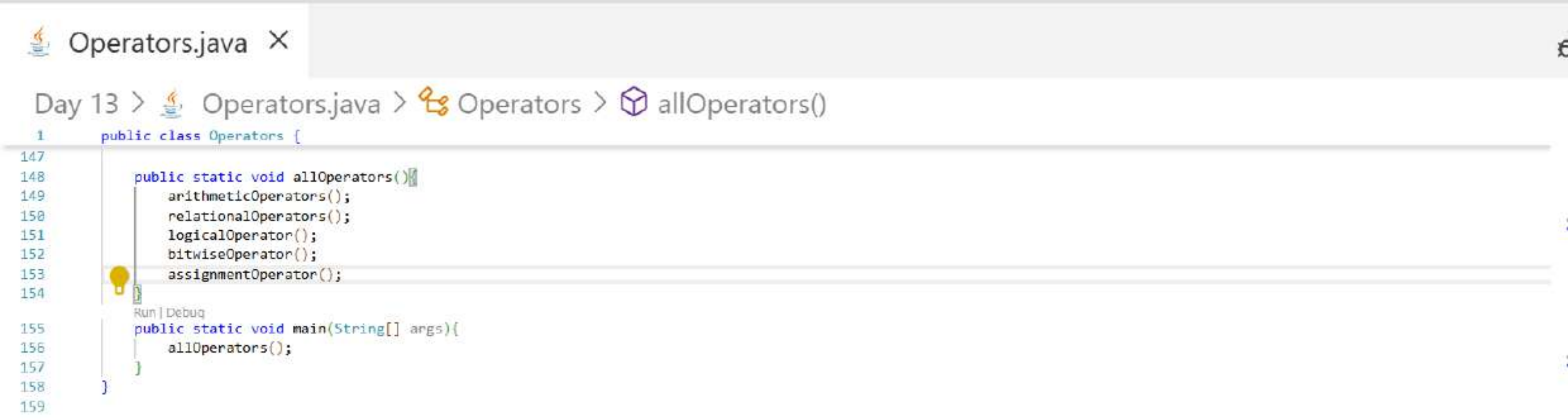
```java
public class Operators {

    public static void logicalOperator(){
        boolean isReady = true;
        boolean isGood = false;

        System.out.println(x:"Logical operators :");

        //AND
        System.out.println(" Logical AND :" + isReady + " AND " + isGood + " => " +(isReady && isGood));

        //OR
        System.out.println("Logical OR :" + isReady + " OR " + isGood + " -> " +(isReady || isGood));

        //NOT
        System.out.println("Logical NOT :" + "!" + isReady + " -> " +(! isReady));

        System.out.println("Logical NOT :" + "!" + isGood + " -> " +(! isGood));

        System.out.println();

    }

```



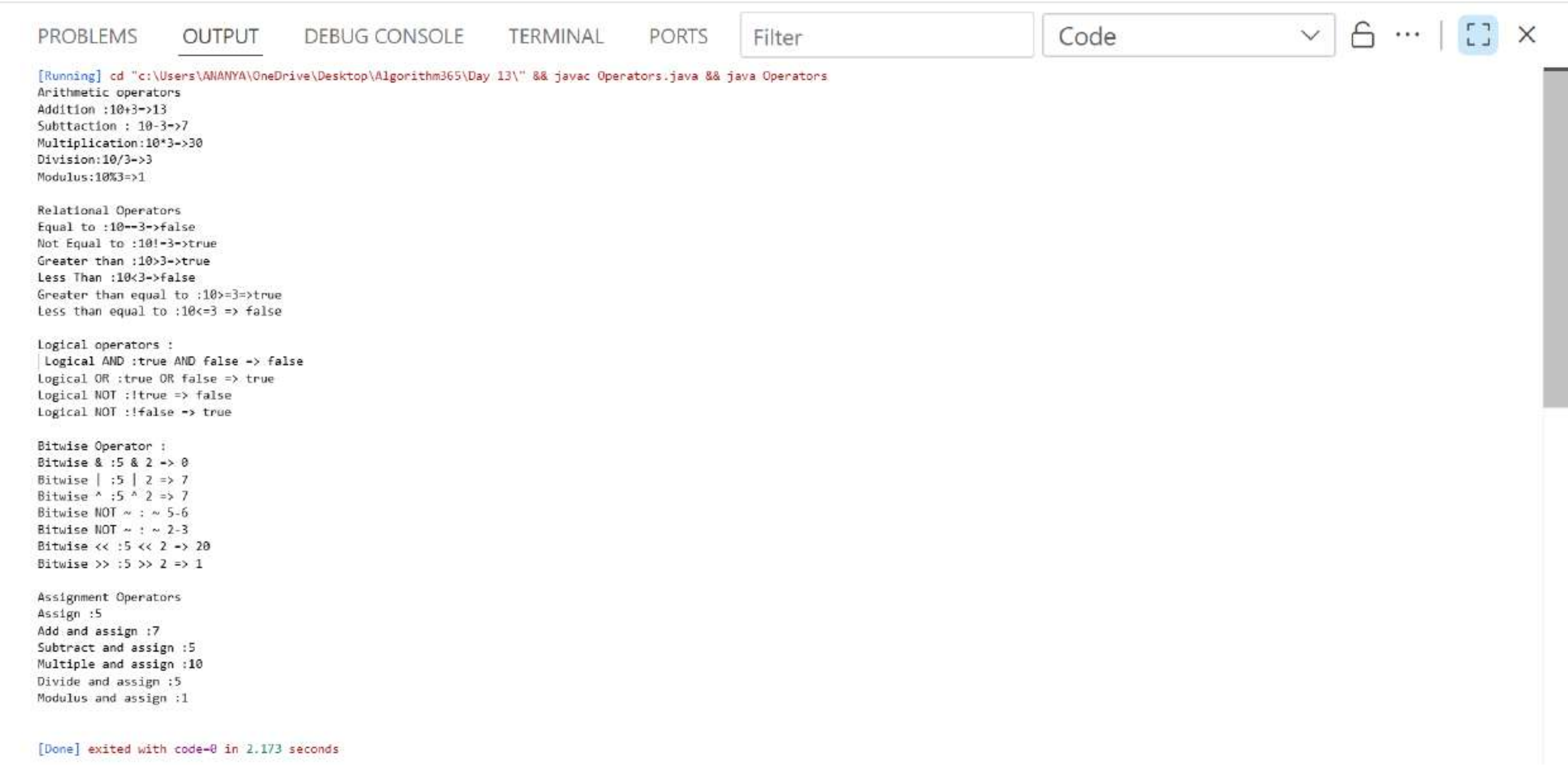
```java
public class Operators {

    public static void bitwiseOperator(){
        int redTeamScore = 5;
        int whiteTeamScore = 2;

        System.out.println(x:"Bitwise Operator :");

        //Bitwise AND " & "
        System.out.println("Bitwise & :" + redTeamScore + " & " + whiteTeamScore + " => " +(redTeamScore & whiteTeamScore));

        //Bitwise OR " | "
        System.out.println("Bitwise | :" + redTeamScore + " | " + whiteTeamScore + " -> " +(redTeamScore | whiteTeamScore));

        //Bitwise XOR " ^ "
        System.out.println("Bitwise ^ :" + redTeamScore + " ^ " + whiteTeamScore + " -> " +(redTeamScore ^ whiteTeamScore));

        //NOT " ~ "
        System.out.println("Bitwise NOT ~ :" + " ~ " + redTeamScore + (~redTeamScore));

        System.out.println("Bitwise NOT ~ :" + " ~ " + whiteTeamScore + (~whiteTeamScore));

        //Left shift " << "
        System.out.println("Bitwise << :" + redTeamScore + " << " + whiteTeamScore + " -> " +(redTeamScore << whiteTeamScore));

        //Right shift
        System.out.println("Bitwise >> :" + redTeamScore + " >> " + whiteTeamScore + " => " +(redTeamScore >> whiteTeamScore));

        System.out.println();
    }
```



```java
public class Operators {

    public static void assignmentOperator(){
        int totalScore = 5;

        System.out.println(x:"Assignment Operators");

        //Assign " - "
        int assign - totalScore;
        System.out.println("Assign :" +assign);

        //Add and assign
        totalScore += 2;
        System.out.println("Add and assign :" +totalScore);

        //Subtract and assign
        totalScore -= 2;
        System.out.println("Subtract and assign :" +totalScore);

        //Multiple and assign
        totalScore *- 2;
        System.out.println("Multiple and assign :" +totalScore);

        //Divide and assign
        totalScore /= 2;
        System.out.println("Divide and assign :" +totalScore);

        //Modulus and assign
        totalScore %= 2;
        System.out.println("Modulus and assign :" +totalScore);

        System.out.println();
    }

```



```
public class Operators {

    public static void allOperators(){
        arithmeticOperators();
        relationalOperators();
        logicalOperator();
        bitwiseOperator();
        assignmentOperator();
    }
    public static void main(String[] args){
        allOperators();
    }
}
```



```
[Running] cd "c:\Users\ANANYA\OneDrive\Desktop\Algorithm365\Day 13\" && javac Operators.java && java Operators
Arithmetic operators
Addition :10+3->13
Subttaction : 10-3->7
Multiplication:10*3->30
Division:10/3->3
Modulus:10%3=>1

Relational Operators
Equal to :10--3->false
Not Equal to :10!-3->true
Greater than :10>3->true
Less Than :10<3->false
Greater than equal to :10>=3=>true
Less than equal to :10<=3 => false

Logical operators :
 Logical AND :true AND false -> false
Logical OR :true OR false => true
Logical NOT :!true => false
Logical NOT :!false -> true

Bitwise Operator :
Bitwise & :5 & 2 -> 0
Bitwise | :5 | 2 => 7
Bitwise ^ :5 ^ 2 => 7
Bitwise NOT ~ : ~ 5-6
Bitwise NOT ~ : ~ 2-3
Bitwise << :5 << 2 -> 20
Bitwise >> :5 >> 2 => 1

Assignment Operators
Assign :5
Add and assign :7
Subtract and assign :5
Multiple and assign :10
Divide and assign :5
Modulus and assign :1


[Done] exited with code=0 in 2.173 seconds
```