

```

using System;

namespace _02_OPERATORS
{
    class Program
    {
        static void Main(string[] args)
        {
            #region 01. ARITHMETIC OPERATORS
            Console.WriteLine("01. Arithmetic Operators:");

            int result;
            // Addition
            result = 20 + 10;
            Console.WriteLine($"Addition: 20 + 10 = {result}"); // Output: 30

            // Subtraction
            result = 20 - 10;
            Console.WriteLine($"Subtraction: 20 - 10 = {result}"); // Output: 10

            // Multiplication
            result = 20 * 10;
            Console.WriteLine($"Multiplication: 20 * 10 = {result}"); // Output: 200

            // Division
            result = 20 / 10;
            Console.WriteLine($"Division: 20 / 10 = {result}"); // Output: 2

            // Modulo
            result = 20 % 10;
            Console.WriteLine($"Modulo: 20 % 10 = {result}"); // Output: 0

            Console.WriteLine();
            #endregion

            #region 02. ASSIGNMENT OPERATORS
            Console.WriteLine("02. Assignment Operators:");

            int x = 15;
            x += 10;
            Console.WriteLine($"x += 10 → x = {x}"); // Output: 25

            x = 20;
            x -= 5;
            Console.WriteLine($"x -= 5 → x = {x}"); // Output: 15

            x = 15;
            x *= 5;
            Console.WriteLine($"x *= 5 → x = {x}"); // Output: 75

            x = 25;
            x /= 5;
            Console.WriteLine($"x /= 5 → x = {x}"); // Output: 5

            x = 25;
            x %= 5;
            Console.WriteLine($"x %= 5 → x = {x}"); // Output: 0

            Console.WriteLine();
            #endregion

            #region 03. RELATIONAL OPERATORS
            Console.WriteLine("03. Relational Operators:");

            bool resultBool;
            int num1 = 5, num2 = 10;

            resultBool = (num1 == num2);
            Console.WriteLine($"Equal (==): {resultBool}"); // Output: False

            resultBool = (num1 > num2);
            Console.WriteLine($"Greater Than (>): {resultBool}"); // Output: False

            resultBool = (num1 < num2);
            Console.WriteLine($"Less Than (<): {resultBool}"); // Output: True
        }
    }
}

```

```

resultBool = (num1 >= num2);
Console.WriteLine($"Greater Than or Equal (>=): {resultBool}"); // Output: False

resultBool = (num1 <= num2);
Console.WriteLine($"Less Than or Equal (<=): {resultBool}"); // Output: True

resultBool = (num1 != num2);
Console.WriteLine($"Not Equal (!=): {resultBool}"); // Output: True

Console.WriteLine();
#endregion

#region 04. LOGICAL OPERATORS
Console.WriteLine("04. Logical Operators:");

bool lx = true, ly = false, lz;

lz = lx && ly;
Console.WriteLine($"AND (true && false): {lz}"); // Output: False

lz = lx || ly;
Console.WriteLine($"OR (true || false): {lz}"); // Output: True

lz = !lx;
Console.WriteLine($"NOT (!true): {lz}"); // Output: False

Console.WriteLine();
#endregion

#region 05. BITWISE OPERATORS
Console.WriteLine("05. Bitwise Operators:");

int a = 12;    // 00001100
int b = 25;    // 00011001

result = a & b; // 00001000 = 8
Console.WriteLine($"Bitwise AND (12 & 25): {result}"); // Output: 8

result = a | b; // 00011101 = 29
Console.WriteLine($"Bitwise OR (12 | 25): {result}"); // Output: 29

result = a ^ b; // 00010101 = 21
Console.WriteLine($"Bitwise XOR (12 ^ 25): {result}"); // Output: 21

Console.WriteLine();
#endregion

#region 06. INCREMENT/DECREMENT OPERATORS
Console.WriteLine("06. Increment/Decrement Operators:");

// Post-Increment
x = 10;
int postInc = x++;
Console.WriteLine($"Post-Increment: x = {x}, result = {postInc}"); // Output: x = 11, result = 10

// Pre-Increment
int y = 10;
int preInc = ++y;
Console.WriteLine($"Pre-Increment: y = {y}, result = {preInc}"); // Output: y = 11, result = 11

// Post-Decrement
x = 10;
int postDec = x--;
Console.WriteLine($"Post-Decrement: x = {x}, result = {postDec}"); // Output: x = 9, result = 10

// Pre-Decrement
y = 10;
int preDec = --y;
Console.WriteLine($"Pre-Decrement: y = {y}, result = {preDec}"); // Output: y = 9, result = 9

Console.WriteLine();
#endregion

#region 07. TERNARY OPERATOR
Console.WriteLine("07. Ternary Operator:");

```

```
a = 20;
b = 10;
result = (a > b) ? a : b;
Console.WriteLine($"Ternary Result (a > b ? a : b): {result}"); // Output: 20

Console.WriteLine();
#endregion

Console.ReadKey();
    }
}
```