

Programming in C# Lab

BCA-DS-651

**Manav Rachna International Institute of Research and
Studies**

School of Computer Applications

Submitted By	
Student Name	Achsah S Thomas
Roll No	22/FCA/BCA/002
Programme	Bachelor of Computer Applications
Semester	6th Semester
Section	A
Department	School Of Computer Applications
Batch	2022-25
Submitted To	
Faculty Name	Dr. Neerja Negi



INDEX

S. No.	Date	Aim of the Experiment	Signature	Grade
1	30/01/2025	Program 1: Write program to demonstrate the working of C# SDK.		
2	30/01/2025	Program 2: Write program to show the use of various data types available in C#.		
3	31/01/2025	Program 3: Write programs to understand the use of Control statements.		
4	31/01/2025	Program 4: Write programs to understand the use of library Functions.		
5	05/02/2025	Program 5: Write a program to demonstrate the use of various arithmetic, unary, logical, bit-wise, assignment and conditional operators.		
6	06/02/2025	Program 6: Write a program to store 10 elements in an array and display the array elements in increasing order.		
7	06/02/2025	Program 7: Demonstrate the use of pass by value and pass by reference by writing a program.		
8	13/02/2025	Program 8: Write a program to implement recursion.		
9	13/02/2025	Program 9: Write programs to implement one dimensional and two-dimensional arrays.		
10	14/02/2025	Program 10: Write programs to understand the working of predefined string functions like Compare (), CompareTo (), Concat (), a.Copy () and Join().		

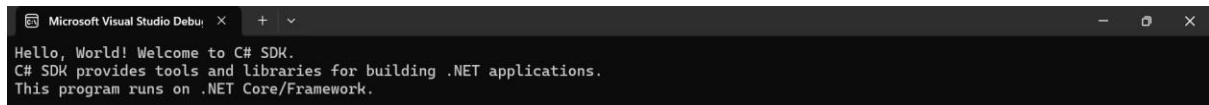
11	27/02/2025	Program 11: Write a program to implement class and its objects.		
12	27/02/2025	Program 12: Write a program to implement constructors.		

1. Write program to demonstrate the working of C# SDK.

```
using System;

class Program
{
    static void Main()
    {
        Console.WriteLine("Hello, World! Welcome to C# SDK.");
        Console.WriteLine("C# SDK provides tools and libraries for building .NET applications.");
        Console.WriteLine("This program runs on .NET Core/Framework.");
    }
}
```

Output:-



The screenshot shows the Microsoft Visual Studio Debug Console window. The title bar reads "Microsoft Visual Studio Debug Console". The console output displays three lines of text: "Hello, World! Welcome to C# SDK.", "C# SDK provides tools and libraries for building .NET applications.", and "This program runs on .NET Core/Framework.".

2. Write program to show the use of various data types available in C#.

```
using System;

class DataTypesExample
{
    static void Main()
    {
        int num = 10;

        float pi = 3.14f;

        double largeDecimal = 123.456;

        char letter = 'A';

        bool isTrue = true;

        string message = "Hello C#";

        Console.WriteLine($"Integer: {num}");

        Console.WriteLine($"Float: {pi}");

        Console.WriteLine($"Double: {largeDecimal}");

        Console.WriteLine($"Character: {letter}");

        Console.WriteLine($"Boolean: {isTrue}");

        Console.WriteLine($"String: {message}");

    }
}
```

Output:-



The screenshot shows the Microsoft Visual Studio Debug Console window. The title bar reads "Microsoft Visual Studio Debug Console". The console output displays the following lines:

```
Integer: 10
Float: 3.14
Double: 123.456
Character: A
Boolean: True
String: Hello C#
```

3. Write programs to understand the use of Control statements.

```
using System;

class ControlStatementsExample
{
    static void Main()
    {
        int num = 5;

        // If-else
        if (num % 2 == 0)
            Console.WriteLine("Even Number");
        else
            Console.WriteLine("Odd Number");

        // For loop
        Console.WriteLine("Numbers from 1 to 5:");
        for (int i = 1; i <= 5; i++)
            Console.Write(i + " ");
        Console.WriteLine();

        // While loop
        int count = 3;
        while (count > 0)
        {
            Console.WriteLine("Countdown: " + count);
            count--;
        }
    }
}
```

Output:-

A screenshot of the Microsoft Visual Studio Debug Console window. The window has a dark background and a title bar that reads "Microsoft Visual Studio Debug Console". The output text is as follows:

```
Odd Number  
Numbers from 1 to 5:  
1 2 3 4 5  
Countdown: 3  
Countdown: 2  
Countdown: 1
```

4. Write programs to understand the use of library functions.

```
using System;

class LibraryFunctionsExample
{
    static void Main()
    {
        double sqrtValue = Math.Sqrt(25);
        string upperCase = "hello".ToUpper();
        string formatted = string.Format("Value: {0}, Square Root: {1}", 100, sqrtValue);
        Console.WriteLine($"Square Root of 25: {sqrtValue}");
        Console.WriteLine($"Uppercase String: {upperCase}");
        Console.WriteLine(formatted);
    }
}
```

Output:-



The screenshot shows a Visual Studio Debug Console window with the following output:

```
Square Root of 25: 5
Uppercase String: HELLO
Value: 100, Square Root: 5
```


5. Write a program to demonstrate the use of various arithmetic, unary, logical, bit-wise, assignment and conditional operators.

```
using System;

class OperatorsExample
{
    static void Main()
    {
        int a = 10, b = 5;

        Console.WriteLine($"Arithmetic: {a} + {b} = {a + b}");

        Console.WriteLine($"Unary: {-a} = {-a}");

        Console.WriteLine($"Logical: {a > b && b < 15}");

        Console.WriteLine($"Bitwise: {a} & {b} = {a & b}");

        Console.WriteLine($"Assignment: a += b -> {a += b}");

        Console.WriteLine($"Conditional: {(a > b ? "A is greater" : "B is greater")}");
    }
}
```

Output:-



```
Microsoft Visual Studio Debug: x + v
Arithmetic: 10 + 5 = 15
Unary: -10 = -10
Logical: True
Bitwise: 10 & 5 = 0
Assignment: a += b -> 15
Conditional: A is greater
```

**6. Write a program to store 10 elements in an array and display the array
A elements in increasing order.**

```
using System;

class ArraySorting
{
    static void Main()
    {
        int[] arr = { 9, 5, 2, 8, 3, 7, 1, 6, 4, 0 };
        Array.Sort(arr);
        Console.WriteLine("Sorted Array:");
        foreach (int num in arr)
            Console.Write(num + " ");
    }
}
```

Output:-



The screenshot shows the Microsoft Visual Studio Debug Console window. The title bar reads "Microsoft Visual Studio Debug Console". The output text is "Sorted Array:" followed by the numbers "0 1 2 3 4 5 6 7 8 9" on the next line.

7. Demonstrate the use of pass by value and pass by reference by writing a program.

```
using System;

class PassExample
{
    static void IncrementByValue(int num)
    {
        num++;
    }

    static void IncrementByReference(ref int num)
    {
        num++;
    }

    static void Main()
    {
        int val = 10;

        IncrementByValue(val);

        Console.WriteLine("After Pass by Value: " + val);

        IncrementByReference(ref val);

        Console.WriteLine("After Pass by Reference: " + val);
    }
}
```

Output:-



The screenshot shows a Visual Studio Debug Console window with the following output:

```
Microsoft Visual Studio Debu: X + v - □ X
After Pass by Value: 10
After Pass by Reference: 11
```

8. Write a program to implement recursion.

```
using System;

class RecursionExample
{
    static int Factorial(int n)
    {
        if (n == 1) return 1;
        return n * Factorial(n - 1);
    }

    static void Main()
    {
        int num = 5;
        Console.WriteLine($"Factorial of {num} is {Factorial(num)}");
    }
}
```

Output:-

A screenshot of a Microsoft Visual Studio Debug Console window. The window has a dark theme and a title bar that reads "Microsoft Visual Studio Debug Console". The console output shows the text "Factorial of 5 is 120".

```
Microsoft Visual Studio Debug Console
Factorial of 5 is 120
```

9. Write programs to implement one dimensional and two dimensional arrays.

1-D Array

```
using System;

class OneDArrayExample
{
    static void Main()
    {
        int[] numbers = { 10, 20, 30, 40, 50 };

        Console.WriteLine("One-Dimensional Array Elements:");

        for (int i = 0; i < numbers.Length; i++)
        {
            Console.Write(numbers[i] + " ");
        }
    }
}
```

Output:-



The screenshot shows the Microsoft Visual Studio Debug Console window. The title bar reads "Microsoft Visual Studio Debug Console". The console output displays the text "One-Dimensional Array Elements:" followed by the numbers "10 20 30 40 50" on the next line.

2-D Array

```
using System;

class TwoDArrayExample
{
    static void Main()
    {
        int[,] matrix = { { 1, 2, 3 }, { 4, 5, 6 } };

        Console.WriteLine("Two-Dimensional Array Elements:");

        for (int i = 0; i < 2; i++)
        {
            for (int j = 0; j < 3; j++)
            {
                Console.Write(matrix[i, j] + " ");
            }

            Console.WriteLine();
        }
    }
}
```

Output:-



```
Microsoft Visual Studio Debug Console
Two-Dimensional Array Elements:
1 2 3
4 5 6
```

10. Write programs to understand the working of predefined string functions like Compare(), CompareTo(), Concat(), Copy() and Join().

```
using System;

class StringFunctions
{
    static void Main()
    {
        string str1 = "Hello";
        string str2 = "World";

        // Compare()
        int result = string.Compare(str1, str2);

        Console.WriteLine("Compare(): " + result); // Returns -1, 0, or 1

        // CompareTo()
        int result2 = str1.CompareTo(str2);

        Console.WriteLine("CompareTo(): " + result2); // Similar to Compare()

        // Concat()
        string concatenated = string.Concat(str1, " ", str2);

        Console.WriteLine("Concat(): " + concatenated);

        // Copy()
        string copiedString = string.Copy(str1);

        Console.WriteLine("Copy(): " + copiedString);

        // Join()
        string[] words = { "C#", "is", "awesome" };

        string joinedString = string.Join(" ", words);

        Console.WriteLine("Join(): " + joinedString);
    }
}
```

Output:-

A screenshot of the Microsoft Visual Studio Debug Console window. The window has a dark background and a title bar that reads "Microsoft Visual Studio Debug Console". The output text is as follows:

```
Compare(): -1  
CompareTo(): -1  
Concat(): Hello World  
Copy(): Hello  
Join(): C# is awesome
```


11. Write a program to implement class and its objects.

```
using System;

class Car
{
    // Fields (Attributes)
    public string Brand;
    public string Model;
    public int Year;


    // Constructor to initialize values
    public Car(string brand, string model, int year)
    {
        Brand = brand;
        Model = model;
        Year = year;
    }

    // Method to Display Car Information
    public void DisplayCarInfo()
    {
        Console.WriteLine("Car Brand: " + Brand);
        Console.WriteLine("Car Model: " + Model);
        Console.WriteLine("Manufacturing Year: " + Year);
    }
}

class Program
{
    static void Main()
    {
        // Creating objects of the Car class
```

```
Car car1 = new Car("Toyota", "Corolla", 2022);  
Car car2 = new Car("Honda", "Civic", 2023);  
  
// Displaying car details  
Console.WriteLine("Car 1 Details:");  
car1.DisplayCarInfo();  
  
Console.WriteLine("\nCar 2 Details:");  
car2.DisplayCarInfo();  
  
}  
}
```

Output:-



The screenshot shows the Microsoft Visual Studio Debug Console window. It displays the output of the program, which is the details of two cars. The first car is a Toyota Corolla from 2022, and the second car is a Honda Civic from 2023. The output is formatted with line breaks and indentation to match the program's logic.

```
Microsoft Visual Studio Debu: x + v - □ x  
Car 1 Details:  
Car Brand: Toyota  
Car Model: Corolla  
Manufacturing Year: 2022  
  
Car 2 Details:  
Car Brand: Honda  
Car Model: Civic  
Manufacturing Year: 2023
```

12. Write a program to implement constructors.

```
using System;
class Student
{
    public string Name;
    public int Age;
    // Constructor
    public Student(string name, int age)
    {
        Name = name;
        Age = age;
    }
    public void Display()
    {
        Console.WriteLine("Student Name: " + Name);
        Console.WriteLine("Student Age: " + Age);
    }
}
class Program
{
    static void Main()
    {
        Student student1 = new Student("Ram", 20);
        student1.Display();
    }
}
```

Output:-



The screenshot shows a dark-themed window titled "Microsoft Visual Studio Debug Console". The window has a tab bar with a single tab labeled "Microsoft Visual Studio Debug Console" and a close button. The main area of the window displays the following text:

```
Student Name: Ram  
Student Age: 20
```