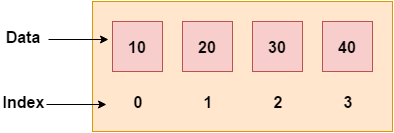
**C# Arrays**

Like other programming languages, array in C# is a group of similar types of elements that have contiguous memory location. In C#, array is an *object* of base type **System.Array**. In C#, array index starts from 0. We can store only fixed set of elements in C# array.



**Advantages of C# Array**

* Code Optimization (less code)
* Random Access
* Easy to traverse data
* Easy to manipulate data
* Easy to sort data etc.

**Disadvantages of C# Array**

* Fixed size

**C# Array Types**

There are 3 types of arrays in C# programming:

1. Single Dimensional Array
2. Multidimensional Array
3. Jagged Array

**C# Single Dimensional Array**

To create single dimensional array, you need to use square brackets [] after the type.

1. int[] arr = new int[5];//creating array

You cannot place square brackets after the identifier.

1. int arr[] = new int[5];//compile time error

Let's see a simple example of C# array, where we are going to declare, initialize and traverse array.

1. using System;
2. public class ArrayExample
3. {
4. public static void Main(string[] args)
5. {
6. int[] arr = new int[5];//creating array
7. arr[0] = 10;//initializing array
8. arr[2] = 20;
9. arr[4] = 30;
11. //traversing array
12. for (int i = 0; i < arr.Length; i++)
13. {
14. Console.WriteLine(arr[i]);
15. }
16. }
17. }

Output:

10

0

20

0

30

**C# Array Example: Declaration and Initialization at same time**

There are 3 ways to initialize array at the time of declaration.

1. int[] arr = new int[5]{ 10, 20, 30, 40, 50 };

We can omit the size of array.

1. int[] arr = new int[]{ 10, 20, 30, 40, 50 };

We can omit the new operator also.

1. int[] arr = { 10, 20, 30, 40, 50 };

Let's see the example of array where we are declaring and initializing array at the same time.

1. using System;
2. public class ArrayExample
3. {
4. public static void Main(string[] args)
5. {
6. int[] arr = { 10, 20, 30, 40, 50 };//Declaration and Initialization of array
8. //traversing array
9. for (int i = 0; i < arr.Length; i++)
10. {
11. Console.WriteLine(arr[i]);
12. }
13. }
14. }

Output:

10

20

30

40

50

**C# Array Example: Traversal using foreach loop**

We can also traverse the array elements using foreach loop. It returns array element one by one.

1. using System;
2. public class ArrayExample
3. {
4. public static void Main(string[] args)
5. {
6. int[] arr = { 10, 20, 30, 40, 50 };//creating and initializing array
8. //traversing array
9. foreach (int i in arr)
10. {
11. Console.WriteLine(i);
12. }
13. }
14. }

Output:

10

20

30

40

50