# Getting and Setting Array Values

In this lesson you'll learn how to access and change array values.

WE'LL COVER THE FOLLOWING

## Accessing Array Values

We use *indexing* to access arrays values just like we did in the previous lesson.

Suppose you declared an *array* of **10** elements.

You can use the array members from `arr[0]` to `arr[9]`.

> **Note:** If you try to access array elements *outside of its bound*, let's say `arr[14]`, the compiler may throw an *"System.IndexOutOfRangeException"*.

Look at the example below for better understanding.

```
using System;

public class MainClass {

   public static void Main(String [] args)
   {
        int[] arr = new int[3] {6,8,5};
        Console.WriteLine(arr[0]);  //outputs 6
        Console.WriteLine(arr[14]); //throws "IndexOutOfRangeException"
   }
}
```

### Iterate over an Array

We can also iterate over the whole array and access any value easily using a `for` loop.

Let’s take a look at the example below.

```
using System;

public class MainClass {

   public static void Main(String [] args)
   {
        int[] arr = new int[] {1, 6, 3, 3, 9};
        for (int i = 0; i < arr.Length; i++)
        {
           Console.WriteLine(arr[i]);
        }
   }
}
```

You can also do the same via `foreach` loop:

```
using System;

public class MainClass {

   public static void Main(String [] args)
   {
        int[] arr = new int[] {1, 6, 3, 3, 9};
        foreach (int element in arr)
        {
           Console.WriteLine(element);
        }
   }
}
```

## Changing Array Values

In order to change the *array* values, we first access them using *indexing* and then change the value at that specified *index* then change the value at that specified *index*.

Take a look at the example below to better understand this concept:

```
using System;

public class MainClass {

   public static void Main(String [] args)
   {
        int[] arr = new int[] { 0, 10, 20, 30};
        // Set
        arr[2] = 100;
        // Get the updated value
        Console.WriteLine(arr[2]); // 100
   }
}
```

As you can see in the example above, originally the value of `arr[2]` is **10**. In line **9** we access the value at *index* **2** and then set it **100**. You can see that the value gets changed from the output of the code above.

These were some of the basics on arrays in C#. Let's delve into other details about arrays in the next lesson.