

Agenda

- Overview of an Array
- Creating Arrays
 - Declaration
 - Construction
 - Initialization
- Manipulating Arrays



Array

- An array is a data structure that contains a number of variables, which are accessed through computed indices.
- The variables contained in an array, are called the elements of the array. They must all be of the same type, and this type is called the element type of the array.

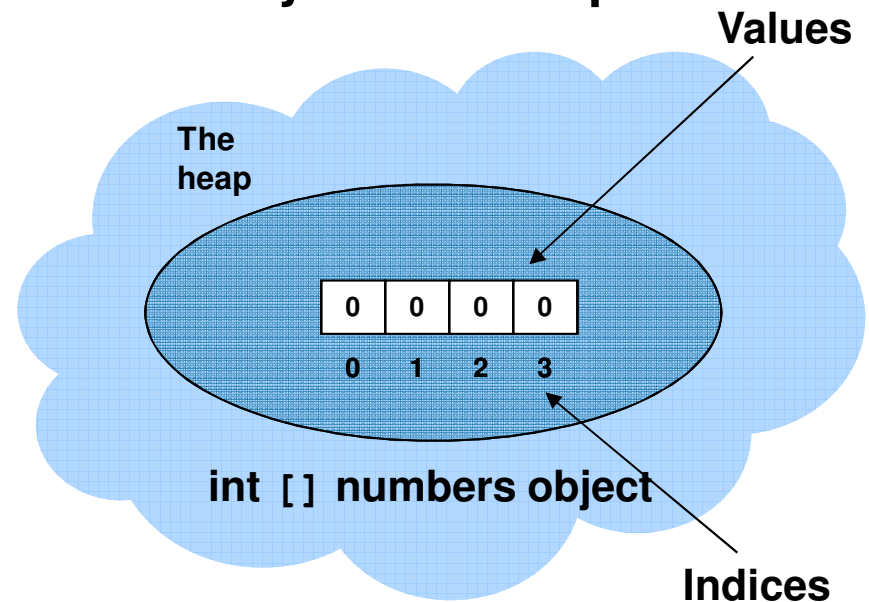
`int [] numbers;`

`int` is the **type**

`[]` is the **Indexing operator**

`numbers` is the **Identifier name**

A one-dimensional array on the heap



```
int [] numbers;  
numbers = new int[4] ;
```

Creating an Array

- There are three steps to create an array:
 - Declaration
 - Construction/Creation
 - Initialization

Declaration →

Construction/Creation →

Initialization →

```
public class ArrayTest
{
    public static void Main(string[] args)
    {
        int[] scores;

        scores = new int[3];

        scores[0] = 10;
        scores[1] = 7;
        scores[2] = 9;
    }
}
```

Declaring an Array

- Declaring an array means providing a name and its data type:

- Single declaration.
- Multiple declaration.
- Array of objects.

Single Declaration

Multiple Declaration

```
public class ArrayTest
{
    public static void Main(string[] args)
    {
        int[] numbers;
        char[] letters, symbols;
        string[] countries;
    }
}
```

Array of Objects

- Note: Arrays actually are created as objects of the System.Array class.

Constructing an Array

- Constructing an array means creating an object of its declared type:
 - Create an array of int type consisting of 3 elements.
 - Declaration and construction at the same time.

```
public class ArrayTest
{
    public static void Main(string[] args)
    {
        int[] numbers;
        char[] letters, symbols;
        string[] countries;

        numbers = new int[3];
        string[] currencies = new string[3];
    }
}
```

An array of int type with three elements.

Declare and Construct at the same time.

Initializing an Array

- Initializing an array means assigning values to its elements:
 - Array index starts with 0.
 - Declaration, construction and initialization at the same time.

Array index starts with 0 (zero).

Declare, Construct, and Initialize at the same time.

```
public class ArrayTest
{
    public static void Main(string[] args)
    {
        int[] numbers;
        char[] letters, symbols;
        string[] countries;
        numbers = new int[3];
        string[] currencies = new string[3];

        numbers[0]=100;
        numbers[1]=200;
        numbers[2]=300;
        int[] newNumbers = {1,2,3};
    }
}
```

Manipulating Arrays

- The following are methods for manipulating arrays:
 - Length gives the size of the array.
 - Printing each element of an array.
 - Assigning array to another array.
 - Passing array to a method.
 - Passing anonymous array.

- Sample Output:

```
100
200
300
6
6
```

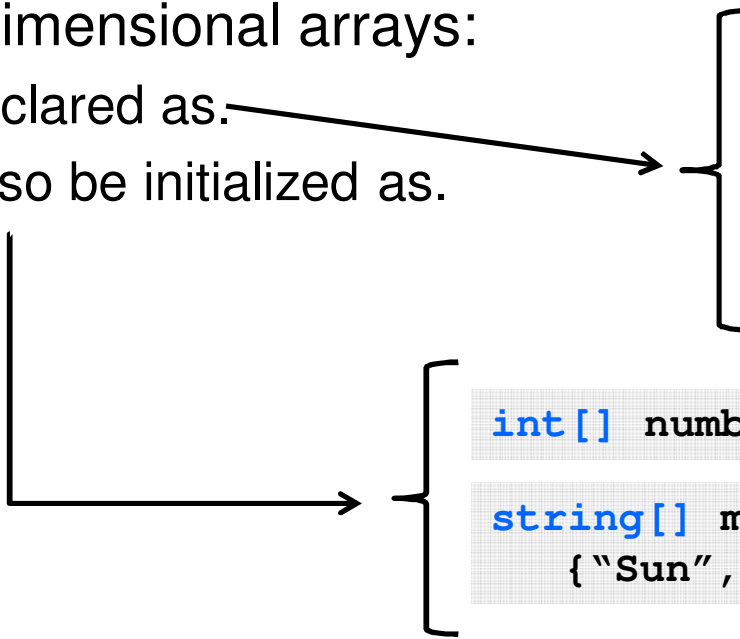
```
public class ArrayTest
{
    public static void Main(string[] args)
    {
        int[] numbers = new int[3];
        numbers[0]=100;
        numbers[1]=200;
        numbers[2]=300;
        int[] newNumbers = {1,2,3};
        for(int i=0; i<numbers.Length; i++)
        {
            System.Console.WriteLine(numbers[i]);
        }
        numbers = newNumbers;
        sumNumbers(numbers);
        sumNumbers(new int[]{3,2,1});
    }

    static void sumNumbers(int[] n)
    {
        int sum=0;
        for (int i=0; i<n.Length; i++)
            sum += n[i];
        System.Console.WriteLine(sum);
    }
}
```

Single-Dimensional Arrays

- Single-dimensional arrays:

- Are declared as.
- Can also be initialized as.



```
int[] numbers ;  
numbers = new int[4] ;
```

```
string[] myStringArray;  
myStringArray = new string[4];
```

```
int[] numbers= new int[] {1,3,5,7,9};
```

```
string[] myStringArray = new string[]  
    { "Sun", "Sat", "Mon", "Tue", "wed", "Fri" };
```

- Note: You can also omit the new operator from the above example. You can assign these values directly without using the new operator.

```
int[ ] numbers = {1,3,5,7,9};
```

```
string [ ] myStringArray = { "Sun", "Sat", "Mon", "Tue", "wed", "Fri" };
```


Multi-Dimensional Arrays

- Multi-dimensional arrays:

- Are declared as.
- Can also be initialized as.

```
int[,] numbers;  
numbers = new int[4,2] ;
```

Two-Dimensional Arrays

```
int[, ,] numbers;  
numbers = new int[4,2,3];
```

Three-Dimensional Arrays

```
int[,] numbers = new int[4,2]  
{{1,2},{3,4},{5,6},{7,8}};
```

Two-Dimensional Arrays

Jagged Arrays

- A jagged array is an array whose elements are arrays. The elements of a jagged array can be of different dimensions and sizes.
- Jagged arrays are also known as an array of arrays.

```
int[][] x = new int[3][];
```

```
x[0] = new int[3]; // first row has three elements
```

```
x[1] = new int[2]; // second row has two elements
```

```
x[2] = new int[4]; // third row has four elements
```

