



# Массивы

Лекция №8



# Массив

- **Массив** представляет собой совокупность переменных одного типа с общим для обращения к ним именем.



# Работа с массивами

- в C# массивы реализованы в виде объектов. Действия по созданию:
- 1) необходимо объявить переменную, которая может обращаться к массиву.
- 2) нужно создать экземпляр массива, используя оператор new.

# Работа с массивами

```
// Объявляем массив
```

```
int[] myArr = new int[5];
```

```
// Инициализируем каждый элемент массива вручную
```

```
myArr[0] = 100;
```

```
myArr[1] = 23;
```

```
myArr[2] = 25;
```

```
myArr[3] = 31;
```

```
myArr[4] = 1;
```

```
foreach (int i in myArr)
```

```
    Console.WriteLine(i);
```

# Инициализация массива

```
// Синтаксис инициализации массива с использованием
```

```
// ключевого слова new
```

```
int[] myArr = new int[] {10,20,30,40,50};
```

```
// Синтаксис инициализации массива без использования
```

```
// ключевого слова new
```

```
string[] info = { "Фамилия", "Имя", "Отчество" };
```

```
// Используем ключевое слово new и желаемый размер
```

```
char[] symbol = new char[4] { 'X','Y','Z','M' };
```

## Неявно типизированные массивы

```
var arr1 = new[] { 1, 2, 3 };
```

```
Console.WriteLine("Тип массива arr1 - {0}",arr1.GetType());
```

```
var arr2 = new[] { "One", "Two", "Three" };
```

```
Console.WriteLine("Тип массива arr2 - {0}",arr2.GetType());
```

```
Console.ReadLine();
```

# Массив объектов

```
// Объявляем и инициализируем массив объектов
```

```
object[] arrByObject = { true, 10, "Привет", 13.7m };
```

```
// Выведем в консоль тип каждого члена массива
```

```
foreach (object me in arrByObject)
```

```
    Console.WriteLine("Тип {0} - {1}", me, me.GetType());
```



# Свойство Length

- Реализация в С# массивов в виде объектов дает целый ряд преимуществ. Одно из них заключается в том, что с каждым массивом связано **свойство Length**



# Свойство Length

```
int[] myArr = { 1, 2, 3, 4 };  
  
for (int i = 0; i < myArr.Length; i++)  
    Console.WriteLine(myArr[i]);  
  
Console.ReadLine();
```



# Многомерные массивы

- *Многомерным* называется такой массив, который отличается двумя или более измерениями, причем доступ к каждому элементу такого массива осуществляется с помощью определенной комбинации двух или более индексов. Многомерный массив индексируется двумя и более целыми числами.

# Двумерные массивы

- Простейшей формой многомерного массива является **двумерный массив**. Такой массив можно представить в виде таблицы, на строки которой указывает один индекс, а на столбцы — другой.

```
// Объявляем двумерный массив
```

```
int[,] myArr = new int[4, 5];
```

```
Random ran = new Random();
```

```
// Инициализируем данный массив
```

```
for (int i = 0; i < 4; i++)
```

```
{
```

```
    for (int j = 0; j < 5; j++)
```

```
    {
```

```
        myArr[i, j] = ran.Next(1, 15);
```

```
        Console.Write("{0}\t", myArr[i, j]);
```

## Массивы трех и более измерений

- В C# допускаются массивы трех и более измерений. Ниже приведена общая форма объявления многомерного массива:
- *тип[,...]* *имя\_массива* = *new* *тип* [*размер1*, *размер2*, ... *размеры*];

# Массивы трех и более измерений

```
int[,,] myArr = new int[5,5,5];
```

```
for (int i = 0; i < 5; i++)
```

```
    for (int j = 0; j < 5; j++)
```

```
        for (int k = 0; k < 5; k++)
```

```
            myArr[i, j, k] = i + j + k;
```

# Инициализация многомерных массивов

*тип[,]* *имя\_массива* =  
*{{val, val, val, ..., val},*  
*{ val, val, val, ..., val},*  
*{val, val, val, ..., val}};*

```
int[,] myArr = {  
    {1,10},  
    {2,20},  
    {3,30},  
    {4,40}  
};
```

# Ступенчатый массив

Ступенчатый массив представляет собой массив массивов, в котором длина каждого массива может быть разной.

*тип [][] имя массива =  
new тип[размер] [];*

```
// Объявляем ступенчатый массив  
  
int[][] myArr = new int[4][];  
  
myArr[0] = new int[4];  
  
myArr[1] = new int[6];  
  
myArr[2] = new int[3];  
  
myArr[3] = new int[4];
```

# Ступенчатый массив

```
int[][] myArr = new int[3][];  
  
myArr[0] = new int[4];  
  
myArr[1] = new int[10];  
  
myArr[2] = new int[1];  
  
Console.WriteLine("Общая длина всего массива: " + myArr.Length);  
  
Console.WriteLine("\nДлина первой цепи: " + myArr[0].Length);  
  
Console.WriteLine("\nДлина второй цепи: " + myArr[1].Length);  
  
Console.WriteLine("\nДлина третьей цепи: " + myArr[2].Length);
```





# Класс Array

- Класс Array Предоставляет методы для создания, изменения, поиска и сортировки массивов, то есть выступает в роли базового класса для всех массивов в среде CLR.

# Примеры работы

Вывод массива

Поиск максимального

```
var mas = new[] { 1, 2, 5, 7, 10 };  
  
Array.ForEach(mas, x => Console.WriteLine(x));  
int max = mas.Max();  
  
Console.WriteLine("max = {0}", max);
```

# Примеры работы

- Выборка нечетных элементов

```
int[] mas2 = Array.FindAll(mas, x => x % 2 == 1);  
Array.ForEach(mas2, x => Console.WriteLine(x));
```