

МАССИВЫ

ПОНЯТИЕ

Массив представляет набор однотипных данных. Объявление массива похоже на объявление. Отличие составляет то, что после указания типа ставятся квадратные скобки

тип_переменной[] название_массива;

Например:

```
int[ ] array_of_chisla;
```

СПОСОБЫ ОБЪЯВЛЕНИЯ МАССИВА

Не инициализированный:

```
int[ ] array_of_chisla;
```

Инициализированный

```
int[ ] array_of_chisla = new int[10];
```

Используя операцию **new**, выделяется память для 10 элементов массива: `new int[10]`.

Число 10 называется **длиной массива**.

СПОСОБЫ ОБЪЯВЛЕНИЯ МАССИВА

✓ `int[] my_array = new int[3] { 1, 2, 3};`

✓ `int[] my_array = new int[] { 1, 2, 3 };`

✓ `int[] my_array = new[] { 1, 2, 3};`

✓ `int[] my_array = { 1, 2, 3};`

ОБРАЩЕНИЯ К ЭЛЕМЕНТАМ МАССИВА

Для обращения к элементам массива используются **индексы**.

Индекс представляет номер элемента в массиве.

Нумерация индекса начинается с нуля, поэтому индекс первого элемента будет равен 0.

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

```
my_array[0] = 2004;
```

```
my_array[1] = 2005;
```

```
my_array[2] = 2008;
```

ЦИКЛ FOREACH

Предназначен для перебора элементов в контейнерах (массивах).

Синтаксис:

```
foreach (тип_данных название_переменной in название_массива)
{
    // действия
}
```

Пример:

```
int[] my_array = new int[] {1, 2, 3};
foreach (int i in my_array)
{
    MessageBox.Show(i.ToString());
}
```

ЦИКЛ FOR

Если foreach последовательно извлекает элементы массива, то в цикле for можно перескакивать на несколько элементов вперед в зависимости от приращения счетчика.

```
int[] my_array = new int[] { 1, 2, 3 };  
for (int i = 0; i < my_array.Length; i++)  
{  
    my_array[ i ] = my_array[ i ] + 100;  
    MessageBox.Show(my_array[ i ].ToString());  
}
```

МНОГОМЕРНЫЕ МАССИВЫ

Массивы характеризуются таким понятием как **ранг** или количество измерений.

Ранее рассматривалась работа с одномерными массивы.

Если ранг массива > 1 (2 и выше), данный массив называется многомерным.

Примеры объявления многомерных массивов:

- ✓ `int[,] my_array;`
- ✓ `int[,] my_array = new int[2, 4];`
- ✓ `int[,] my_array = new int[2, 4] { { 0, 1, 2, 3 }, { 0, 1, 2, 3 } };`
- ✓ `int[,] my_array = new int[,] { { 0, 1, 2, 3 }, { 0, 1, 2, 3 } };`

ПРИМЕР ЗАПОЛНЕНИЯ ЗНАЧЕНИЯ ОДНОМЕРНОГО МАССИВА

```
int[] a = new int[5];  
    for (int i = 0; i < 5; i++)  
    {  
        a[i] = 5 * i + 3;  
        MessageBox.Show(a[i].ToString());  
    }
```

Вывод результата заполнения массива в элемент ComboBox:

```
comboBox1.Items.Clear();  
    for (int i = 0; i < 5; i++)  
    {  
        comboBox1.Items.Add(a[i].ToString());  
    }
```

ЗАПОЛНЕНИЕ ОДНОМЕРНОГО МАССИВА С ПОМОЩЬЮ RANDOM

```
int[] my_array = new int[10];  
    Random rand = new Random();  
    for (int i = 0; i < my_array.Length; i++)  
    {  
        my_array[i] = rand.Next();  
        comboBox1.Items.Add(my_array[i].ToString());  
    }
```

ЗАПОЛНЕНИЕ ОДНОМЕРНОГО МАССИВА С ПОМОЩЬЮ RANDOM С ЗАДАНЫМ ДИАПАЗОНОМ

```
int[] my_array = new int[10];  
    Random rand = new Random();  
    for (int i = 0; i < my_array.Length; i++)  
    {  
my_array[i] = rand.Next(10, 15);  
        comboBox1.Items.Add(my_array[i].ToString());  
    }
```

ЗАПОЛНЕНИЕ ДВУМЕРНОГО МАССИВА

```
int[,] my_array = new int[5, 5];  
Random ran = new Random();  
for (int i = 0; i < 5; i++)  
{  
    for (int j = 0; j < 5; j++)  
    {  
        my_array[i, j] = ran.Next(1, 100);  
        MessageBox.Show(my_array[i,j].ToString());  
    }  
}
```