

# Массивы

**№** 29 **Курс:** Процедурное программирование на языке C#  
**урока:**

**Средства обучения:** Visual Studio 2019 Community Edition

## Обзор, цель и назначение урока

Данный урок познакомит вас с такой конструкцией в языках программирования как массивы. Вы получите общее понимание их структуры и простые возможности ее применения на практике.

## Изучив материал данного занятия, учащийся сможет:

- Понимать, что такое массив.
- Уметь заполнять массив элементами при его создании.
- Уметь обходить элементы массива в цикле.
- Уметь применять операторы для досрочной остановки работы цикла и для перехода на новую итерацию цикла, без завершения текущей.

## Содержание урока

1. Рассмотрение практической задачи
2. Что такое массивы
3. Решение практической задачи с применением массивов

## Резюме

- Оператор **break**, при использовании его в теле цикла, заставляет цикл резко прекратить свою работу. Цикл прекращает свою работу и дальнейшее выполнение программы продолжается сразу за телом цикла.
- Оператор **continue** (continue переводится на русский, как «продолжить»), при использовании его в теле цикла, останавливает выполнение цикла, но, в отличие от оператора break, не «убивает» цикл.  
Для текущей итерации код в теле цикла ниже оператора **continue** выполняться не будет. После оператора continue выполнение программы продолжится с начала цикла, а именно - с участка проверки условия цикла.  
Такова задача оператора continue – остановить выполнение текущей итерации цикла и продолжить выполнение цикла с новой итерации.
- **Массив** — это структура данных, содержащая в себе несколько переменных, доступ к которым осуществляется по вычисляемым индексам, которые используются вместо имен переменных. Содержащиеся в массиве переменные именуются элементами этого массива.

- **Индекс** — это некоторое числовое значение, номер элемента массива.
- Все переменные в массиве (элементы массива) имеют одинаковый тип, который называется типом элементов массива.
- При объявлении массива сначала указывается тип всех переменных (а точнее элементов) которые будут храниться в этом массиве. Далее, после типа элементов массива, указывается пара квадратных скобок, в скобках ничего нет. Эта пара квадратных скобок после типа говорит нам о том, что это массив. Далее, после квадратных скобок, указывается имя массива. Завершаем выражение объявления массива точкой с запятой. В этом случае у нас создается только переменная-имя массива нужного типа. Сам массив в оперативной памяти не создается.
- После объявления массива можно сразу создать его и заполнить массив нужными элементами. Для этого, после объявления массива, перед точкой с запятой ставим символ присвоения, а затем создаем блок из операторных скобок, в котором, просто через запятую, указываем несколько значений того типа, который указан при объявлении массива. После блока ставим точку с запятой. Этот блок, правильно называется **«инициализатор»**, «инициализатор массива». В теле блока-инициализатора, мы можем указать несколько значений через запятую. Эти значения и станут элементами массива.
- Если при создании массива мы укажем блок инициализатора, в котором не пропишем ни одного значение – будет создан **пустой массив**. В таком массиве не будет ни одного элемента.
- Индексация (нумерация) элементов массивов в языке C# начинается с нуля.
- Имя массива, которое мы указываем при создании массива — это простая, четырёхбайтовая, переменная, которая в себе содержит, адрес нулевого элемента массива в оперативной памяти. Адрес, который хранится в переменной имени массива называют ссылкой.
- Чтобы получить значение элемента массива по индексу этого элемента, нужно написать имя массива и, после имени массива, в квадратных скобках, указать индекс интересующего нас элемента.
- Чтобы изменить значение элемента в массиве нужно написать имя массива, в квадратных скобках указать индекс элемента и после знака присвоения указать новое значение для этого элемента.

### Закрепление материала

- Что такое массив?
- В чем отличие объявления массива от его создания?
- Что такое пустой массив?
- Какова задача блока инициализатора?
- Как с помощью цикла организовать чтение всех элементов массива?

- Какую функцию выполняют в циклах операторы break и continue?

### Самостоятельная деятельность учащегося

- Задание 1

Ознакомьтесь с дополнительными материалами к уроку.

- Задание 2

Напишите программу, в которой создайте массив строковых элементов. В блоке инициализатора заполните массив названиями месяцев по порядку. Далее создайте цикл, в котором выведите на экран консоли все содержимое массива строк.

- Задание 3

Напишите программу, в которой воспользуйтесь массивом строк из предыдущего задания.

Создайте цикл, в котором обратитесь к пользователю. Пользователь для первой итерации цикла должен ввести строковое значение первого по порядку месяца, для второй итерации – строковое значение второго месяца и так далее до конца массива. Если пользователь введет неправильно текущее значение месяца – итерация цикла должна повториться с тем, чтобы пользователь попробовал ввести значение еще раз. Если пользователь ошибется при вводе значения 5го месяца – цикл должен прекратить свою работу.

### Рекомендуемые ресурсы

[https://ru.wikipedia.org/wiki/Массив\\_\(тип\\_данных\)](https://ru.wikipedia.org/wiki/Массив_(тип_данных))

<https://docs.microsoft.com/ru-ru/dotnet/csharp/tour-of-csharp/arrays>

<https://docs.microsoft.com/ru-ru/dotnet/csharp/programming-guide/arrays/single-dimensional-arrays>