

# Ruby Essential

Массивы и хеши

# Ruby Essential

Автор курса



Юля Гончаренко

# Ruby Essential

После урока обязательно



Повторите этот урок в видео формате  
на [ITVDN.com](http://itvdn.com)

Доступ можно получить через руководство  
вашего учебного центра



Проверьте как Вы усвоили данный  
материал на [TestProvider.com](http://testprovider.com)

# Ruby Essential

Тема

Массивы и хеши

# Ruby Essential

## Массивы и хеши

1. Понятие массива. Создание массива и доступ к элементам
2. Основные методы для работы с массивами.
3. Многомерные массивы.
4. Понятие хеша. Создание и получение доступа к парам ключ-значение.
5. Методы для работы с хешами.

# Ruby Essential

## Что такое массивы?

Массив- упорядоченная структура данных, которая создает ассоциацию между индексом и самим элементом данных.

Массивы заключаются в квадратные скобки, перечислением элементов идет через запятую.

Массивы индексируются целыми числами, начиная с 0.

Массивы являются динамическим: то есть можно изначально задать размер массива(не является обязательным), но в последствии увеличиваться.

В массивах в Ruby могут храниться объекты различных типов данных.

# Ruby Starter

## Создание массивов

Array.new(array)  
Array.new(size, obj )

```
ar=Array.new(5, "get")  
puts ar
```

Array=[элементы массива]  
Array=%w[Массив Строк]

```
array=[1, "two", 3]  
puts array
```

# Ruby Essential

## Получение доступа к элементам массива

`array[index]`

`array[начало, длина]`

`array[диапазон]`

`array.slice(индекс)`

`array.slice(начало, длина)`

`array.slice(диапазон)`

```
puts arr[0] #=>Первый элемент
```

```
puts arr[0,3] #=>Подмассив с 0 по 4 элемент
```

```
puts arr[1..3] #=>Подмассив с 2 по 4 элемент
```

```
puts arr.slice(1) #=>Второй элемент
```

```
puts arr.slice(1,4) #=>Подмассив с 2 по 5 элемент
```

```
puts arr.slice(1..3) #=>Подмассив с 2 по 4 элемент
```



# Ruby Starter

## Операторы, которые применяются для работы с массивами

<code>array+other_array</code>	Сцепление — возвращает новый массив, созданный из двух массивов путем добавления одного к другому.
<code>array-other_array</code>	Вычитание массивов — возвращает новый массив, который копирует оригинальный массив, но удаляет из него элементы, которые есть в другом массиве.
<code>array&amp;other_array</code>	Пересечение множеств — возвращает новый массив, состоящий из элементов, которые есть в обоих массивах, но без дубликатов.
<code>array other_array</code>	Объединение множеств — возвращает новый массив, который объединяет элементы , но с удаленными дубликатами.
<code>array&lt;=&gt;other_array</code>	Сравнение — возвращает целое число (-1, 0, или +1) если текущий массив меньше, равен или больше другого массива.
<code>array==other_array</code>	Равенство — два массива считаются равными, если количество элементов и соответствующие пары элементов равны
<code>array*int</code>	Повторение массива заданное количество раз
<code>array&lt;&lt; obj</code>	Добавляет передаваемый объект в конец массива. Возвращает массив с уже добавленным элементом.

# Ruby Essential

## Основные методы, применяемые к массивам

<code>array.clear</code>	<code>array.concat(other_array)</code>	<code>array.count</code>	<code>array.delete(obj)</code>	<code>array.empty?</code>
<code>array.length</code> <code>array.size</code>	<code>array.include(obj)</code>	<code>array.insert(index, obj)</code>	<code>array.eql?(other)</code>	<code>array.each {  item  block }</code>
<code>array.reverse</code>	<code>array.zip(other_array)</code>	<code>array.sort</code> <code>array.sort {  a,b  block }</code>	<code>array.delete_at(index)</code>	<code>array.replace(other_array)</code>

# Ruby Essential

## Хеши

Хеш — это ассоциативный массив. То есть это неупорядоченная структура данных, в которой содержатся ключи, к каждому из которых принадлежит значение.

Хеши заключаются в фигурные скобки. Для указания связи между ключем и значением используют знак => или : .

В качестве ключей в хеше могут использоваться любые объекты различных типов данных.

Самым простым способом создания хеша является заключение пар ключ-значение в фигурные скобки.

```
hash = {"1" => "Понедельник", "2" => «Вторник"}
```

Также можно создать хеш с помощью метода new.

```
hash= Hash.new
```

# Ruby Essential

## Получение доступа

`hash.default`  
(key=nil)

Получение элемента — возвращает значение соответствующее ключу key. Если ключа не существует, то возвращается значение по-умолчанию.

`hash[key]`

Возвращает значение по-умолчанию, т.е. значение, которое будет возвращать `hash[key]`, если ключа key не существует в хеше hash.

`hash.include?`  
(key)

Возвращает true, если заданный ключ находится в хеше hash.

`hash.index`  
(value)

Возвращает ключ для заданного значения. Если значение не найдено, возвращает nil.

`hash.index`  
(value)

Возвращает ключ для заданного значения. Если значение не найдено, возвращает nil.

# Ruby Essential

## Основные методы, применяемые к хэшам

*hash.sort*

*hash.delete(obj)*

*hash.invert*

*hash.merge  
(other\_hash)*

*hash.clear*

*hash.value?(value)*

*hash.update(othr\_h  
as h)*

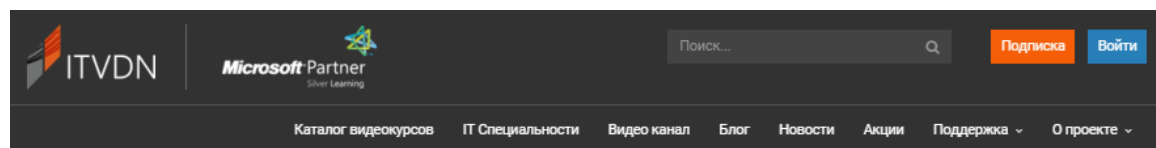
*hash.default*

*hash.replace(othe  
r\_hash)*

*hash.inspect*

# Смотрите наши уроки в видео формате

ITVDN.com



Посмотрите этот урок в видео формате на образовательном портале [ITVDN.com](http://itvdn.com) для закрепления пройденного материала.

### 1 ITVDN 2015. Наши награды

**ITVDN**  
лучший ресурс  
для  
online обучения

### ITVDN 2015. Итоги года

В марте 2015 года ITVDN стал победителем конкурса IT Education Awards, который проходил в рамках IT Jam 2015 и награжден как лучший образовательный ресурс в номинации Online Education. Экспертное жюри, в состав которого вошли представители ведущих IT компаний, отметили такие преимущества ITVDN, как системный подход в обучении, позволяющий удаленно получить качественное образование по наиболее популярным специальностям, высокий профессионализм авторов видео курсов и использование современных методик оценки знаний.

Все курсы записаны сертифицированными тренерами, которые работают в учебном центре CyberBionic Systematics

## Новые видео

Исключения	0
Итераторы и генераторы	0

## Популярные видео курсы

Видео курс C# Стартовый (для начинающих)	9 уроков (16 ч. 3 мин.)
Видео курс по шаблонам проектирования	29 уроков (16 ч. 7 мин.)

## Теги

.NET Developer
Frontend Developer



# Проверка знаний

## TestProvider.com

TestProvider | Мы помогаем людям оценить себя

Регистрация | Войти

Главная | Каталог | Сертификация Microsoft | Поддержка | О нас

### Тестирование

Языки программирования и информационные технологии

**Microsoft**

C# ASP.NET MVC JavaScript Patterns Of Design SQL Architecture Guide WCF HTML&CSS XML SEO WPF HTML5&CSS3 JQuery XNA SharePoint GUI for Android Windows Azure Platform Microsoft Patterns&Practices TFS SCRUM ReSharper TDD WWF LINQ Entity Framework Windows Forms Refactoring Microsoft Expression Blend 4 Windows Phone 8 Windows 8 AppStore Visual Studio Tips&Tricks MSF MEF SilverLight AJAX MEF Service Oriented Architecture

Пройти тест

Наши партнеры

Microsoft Partner CyberBionic ITVDN PROMETRIC TEST CENTER PEARSON VUE Authorized Test Center Windows Azure Cloud Partner EBA

Дополнительные ресурсы:

Очное обучение | On-line обучение | Видео обучение

TestProvider – это online сервис проверки знаний по информационным технологиям. С его помощью Вы можете оценить Ваш уровень и выявить слабые места. Он будет полезен как в процессе изучения технологии, так и общей оценки знаний IT специалиста.

После каждого урока проходите тестирование для проверки знаний на [TestProvider.com](http://TestProvider.com)

Успешное прохождение финального тестирования позволит Вам получить соответствующий Сертификат.



# TRANSACT-SQL

Q&A



# Информационный видеосервис для разработчиков программного обеспечения

