

# Основные понятия в JavaScript

JSP & Servlets  
7 уровень, 1 лекция

ОТКРЫТА

## 2.1 Переменные и боль

Начнем с самого интересного. В JavaScript есть переменные, но у этих переменных нет типа. Любой переменной можно присвоить абсолютно любое значение. Выглядит безобидно или даже удобно, пока вам не понадобятся типы.

Для объявления переменной используется ключевое слово `var`:

```
var name;  
var name = value;
```

Примеры работы с переменными в JavaScript:

1	<code>var a = 10, b = 20;</code>
2	<code>var c = a*a + b*b;</code>
3	
4	<code>var s = "Диагональ равна:";</code>
5	<code>console.log( s + Math.sqrt(c));</code>

Отличный и понятный код, не правда ли? Возможно, это последний раз в твоей жизни, когда вы видите красивый и понятный JavaScript код. Запомни этот момент :)

## 2.2 Типизация в JavaScript

Как уже говорилось выше, у переменных в языке JavaScript типа нет. Но вот у значений переменных типы есть. Вот 5 самых распространенных типов в JavaScript:

#	Тип	Пример	Описание
1	Number	<code>var pi = 3.14;</code>	Содержит любое число
2	String	<code>var s = "Привет!";</code>	Содержит строку
3	Boolean	<code>var result = true;</code>	Содержит true или false
4	Array	<code>var arr = [1, 2, 3, 4, 5];</code>	Содержит массив элементов
5	Date	<code>var current = new Date();</code>	Содержит дату
6	Object	<code>var o = {   width: 100,   height: 200 }</code>	Содержит объект, который состоит из пар key, value. Чем-то похож на HashMap в Java
7	Function	<code>function sqr(var x) {</code>	Функция

		<pre>        return x*x;     } }</pre>	
--	--	--	--

Для определения типа объекта используется ключевое слово `typeof`, пример:

1	<code>var s = "134";</code>
2	<code>var x = (typeof s == "String") ? s*1 : s;</code>

### 2.3 Функции и return

Ну и конечно же в JavaScript есть функции. Классов там нет, так что объявлять функции можно в любом месте кода. Даже в других функциях. Общий формат имеет вид:

1	<code>function name(var a, var b, var c) {</code>
2	<code>    // код функции</code>
3	<code>    return result;</code>
4	<code>}</code>

Типа у функции нет. А зачем, если в самом языке нет контроля совместимости типов? Параметры функции тоже могут отсутствовать. Как и команда `return`, которая возвращает значение.

При вызове функции можно передавать любое количество параметров любого типа. Лишние отбросится, недостающие будет равны `null`.

Примеры функций:

1	<code>function getValue(name)</code>
2	<code>{</code>
3	<code>    return this[name];</code>
4	<code>}</code>

1	<code>function setValue(name, value)</code>
2	<code>{</code>
3	<code>    this[name] = value;</code>
4	<code>}</code>

### 2.4 Массивы в JavaScript

Массивы в JavaScript очень похожи на массивы в Java. Примеры:

1	<code>var array = [1, 2, 3, 4, 5];</code>
2	<code>array[3] = array[2];</code>
3	<code>console.log (array[0]);</code>

Они могут иметь значения любого типа, даже другие массивы:

1	<code>var array = [1, "Привет", 3.14, [4, 5] ];</code>
2	<code>array[3] = array[2];</code>
3	<code>console.log (array[0]);</code>

Кроме того, массивы также ведут себя как коллекции – в них можно динамически добавлять элементы:

1	<code>var array = [];</code>
2	<code>array.push(100);</code>

```
3 array.push(101);
4 array.push(102);
5
6 array[1] = array[2];
7 console.log (array[0]);
```

## 2.5 Объекты в JavaScript

Объекты в JavaScript очень похожи на HashMap из Java: они содержат пары ключ-значение. Пример:

```
1 var obj = {
2   name: "Bill Gates",
3   age: 67,
4   company: "Microsoft"
5 };
6
7 console.log (obj.age);
```

К полям объекта можно обращаться двумя разными способами:

```
var x = obj.age;
var x = obj["age"];
```

Как и в случае с HashMap, поля можно создавать и удалять. Пример:

```
1 var obj = {};
2 obj.name = "Bill Gates";
3 obj.age = 67;
4 obj.company = "Microsoft";
5
6 delete obj.age;  //удаление поля
```

< Предыдущая лекция

Следующая лекция >

− +15 +

Комментарии (5) + 2

популярные

новые

старые

JavaCoder

Введите текст комментария

Евгений Frontend Developer

14 октября, 09:01

var устаревшее объявление, сейчас используют let и const. В современном мире пишут на языке TypeScript, в котором с типизацией всё в порядке. Тот же JavaScript, только добавили типы. TypeScript изучается за один день.

Ответить

0

Владимир Уровень 68

1 сентября, 13:07

Так что в итоге будет присвоено переменной X ?

```
1 var s = "134";
2 var x = (typeof s == "String") ? s*1 : s;
```

Ответить

− +3 +

Владимир Уровень 68

1 сентября, 13:09 ...

если строка, то присвоено число, а если не строка, то присвоена строка?

Ответить

− +2 +

Andrey Panchenko Моет полы в Яндекс

16 сентября, 10:16 ...

F12 -> Console -> копируешь код -> вставляешь и дописываешь строку вывода типа полученного x -> кликаешь enter -> получаешь вывод "string"

```
var s = "134";
var x = (typeof s == "String") ? s*1 : s;
console.log(typeof x);
```

string

Ответить

− +1 +

Сергей Уровень 68

9 октября, 12:22 ...

```
1 var s = "134";
2 var x = (typeof s == "String") ? s*1 : s;
3 console.log(typeof x);
4 Вывод: string
5
6 var s = 134;
7 var x = (typeof s == "String") ? s*1 : s;
8 console.log(typeof x);
9 Вывод: number
```

Ответить

− 0 +

ОБУЧЕНИЕ

Курсы программирования

Курс Java

Помощь по задачам

Подписки

Задачи-игры

СООБЩЕСТВО

Пользователи

Статьи

Форум

Чат

Истории успеха

Активности

КОМПАНИЯ

О нас

Контакты

Отзывы

FAQ

Поддержка



JavaRush — это интерактивный онлайн-курс по изучению Java-программирования с нуля. Он содержит 1200 практических задач с проверкой решения в один клик, необходимый минимум теории по основам Java и мотивирующие фишки, которые помогут пройти курс до конца: игры, опросы, интересные проекты и статьи об эффективном обучении и карьере Java-девелопера.

ПОДПИСЫВАЙТЕСЬ

ЯЗЫК ИНТЕРФЕЙСА

Русский ▾

СКАЧИВАЙТЕ НАШИ ПРИЛОЖЕНИЯ



