

HTML - Урок 11

Переменные в JavaScript



Геннадий Караченцев

www.weblaboratory.in.ua

Имена переменных

Для имен переменных в JavaScript существует 2 ключевых ограничения:

1. Имя может состоять из: букв, цифр, символов \$ и _
2. Первый символ не должен быть цифрой.

Хорошо

```
var myName;  
var my_name;  
var $ = 1;  
var _ = 2;  
var __ = 3;
```

Плохо

```
var 1a;  
var my-name;  
var my name;  
var мое имя = 'Геннадий';
```

Имя переменного должно быть осмысленным и отображать содержимое.

Типы данных

В JavaScript существует несколько основных типов данных.

1. Число «number»
2. Строка «string»
3. Булевый (логический) тип «boolean»
4. Специальное значение «null»
5. Специальное значение «undefined»
6. Объекты «object»

Оператор позволяет узнать какой тип у переменной `typeof`

Число «number»

Используется для целых и дробных чисел!

```
var n = 123;  
n = 12.345;
```

Так же есть специальные числовые значения Infinity (бесконечность) и NaN (ошибка вычислений).

К примеру Infinity можно получить, при делении на 0:

```
alert(1 / 0); // Infinity
```

NaN будет выведена при некорректном математическом действии:

```
alert("строка" * 2); // Nan, ошибка
```

Число «string»

String хранить текстовые данные

```
var str = "Двойные кавычки";
```

```
str = 'Одинарные кавычки';
```

В JavaScript одинарные и двойные кавычки одинаковые.

Булевый (логический) тип «boolean»

Имеет два значения: true (истина) и false (ложь).

Такой тип используется для хранения значения типа да/нет, например:

```
var checked = true; // Истина, 1  
checked = false; // Лож, 0
```

Этот тип, очень часто используется в логический и условных операторах.

Специальное значение «null»

Значение null не относится ни к одному из типов выше, а образует свой отдельный тип, состоящий из единственного значения null:

```
var age = null;
```

Это просто специальное значение, которое имеет смысл «ничего» или «значение неизвестно».

Специальное значение «undefined»

Значение `undefined`, как и `null`, образует свой собственный тип, состоящий из одного этого значения. Оно имеет смысл «значение не присвоено».

```
var x;  
console.log( x );
```

Можно присвоить `undefined` и в явном виде, хотя это делается редко:

```
var x = 123;  
x = undefined;  
console.log( x );
```


Объекты «object»

Он используется для коллекций данных и для объявления более сложных сущностей.

Объявляются объекты при помощи фигурных скобок {...}, например:

```
var user = { name: "Вася" };
```

Оператор typeof

Оператор typeof возвращает тип значения переменной.

У него есть два синтаксиса со скобками и без:

1. Синтаксис оператора: typeof x
2. Синтаксис функции: typeof(x)

```
typeof undefined; // undefined
```

```
typeof 0; // number
```

```
typeof true; // boolean
```

```
typeof "string"; // string
```

```
typeof {}; // object
```

Операции или действия с переменными

Математические операции:

1. Умножение *
2. Деление /
3. Сложение +
4. Вычитание -
5. Присвоение =
6. Инкремент ++
7. Декремент --
8. Остаток от деления %

Функции и свойства по работе с переменными:

length, indexOf(), replace(), parseInt(), parseFloat(), toString()