



Подробный курс

Современные веб-технологии

Лекция 9. Создание динамического наполнения страницы. Основы JavaScript

Понятие «JavaScript»

- **JavaScript** —скриптовый язык программирования
- Первоначально назывался LiveScript
- 4 декабря 1995 года LiveScript переименовали в JavaScript
- JavaScript обладает рядом свойств:
 - объектно-ориентированного языка
 - функционального языка

Особенности языка

- Объекты, с возможностью интроспекции
- Функции как объекты первого класса
- Автоматическое приведение типов
- Автоматическая сборка мусора
- Анонимные функции
- Все идентификаторы зависят от регистра
- В названиях переменных можно использовать буквы, подчеркивание, символ доллара, арабские цифры
- Названия переменных не могут начинаться с цифры

Структура JavaScript

- Ядро (ECMAScript)
- Объектная модель браузера (BOM)
- Объектная модель документа (DOM)

Основной синтаксис

- Числа
- Строки
- Другие типы (например, Boolean)
- Переменные
- Операторы
- Объекты:
 - Функции
 - Массивы
 - Дата
 - Пользовательские объекты
- Управляющие структуры
- Внутренние функции
- Null, Undefined, Error

Числа

- Все числа в JavaScript – 64-битные двойной точности
- Поддержаны стандартные арифметические операторы, такие как сложение, деление, остаток от деления и т.д.
- Встроенная функция `parseInt()` – преобразование строки в число
- Результатом математической операции с NaN всегда будет NaN
- В JavaScript есть значения `Infinity` и `-Infinity`

Строки

- Строки в JavaScript это последовательность СИМВОЛОВ.
- Для получения длины строки и объектов есть свойство `length`
- Строка имеет набор полезных методов:
 - `charAt` – получение символа по его номеру вхождения
 - `replace` – замена подстроки
 - `toUpperCase` – перевод всех символов в верхний регистр

Логический тип

- Возможные значения – true и false
- Любое значение может быть преобразовано к логическому по правилам:
 - false, 0, пустая строка (""), NaN, null, и undefined трактуются как false
 - все остальное как true
- Преобразование можно провести в явном виде с использованием функции Boolean()
- Поддерживаются стандартные логические операторы && (логическое *и*), || (логическое *или*) и ! (логическое *не*).

Переменные и Операторы

- ❑ Переменная объявляется с использованием ключевого слова `var`
- ❑ Стандартные математические операторы `+`, `-`, `*`, `/`, `%`
- ❑ Значения присваиваются оператором `=`, также есть операторы составного присваивания `+=` и `-=`
- ❑ `++` и `--` для увеличения или уменьшения значения переменной на единицу
- ❑ Оператор `«+»` применяется для конкатенации строк
- ❑ Операторы сравнения: `<`, `>`, `<=` и `>=`
- ❑ Оператор двойного равенства производит преобразование типов
- ❑ Оператор тройного равенства учитывает типы:

Управляющие структуры

- Условные операторы: if и else
- Циклы while и do-while
- Цикл for
- Операторы && и || «закорачиваются»
- Тернарный оператор
- Конструкция switch для разветвления на основании числового или строчного значения

Объекты

- Объекты — это коллекции пар «ключ-значение»
- В JavaScript почти все является объектами (за исключением базовых типов)
- «Ключ» должен быть в JavaScript строкой, а «значение» может быть любым JavaScript типом
- Два способа создать объект:
 - `var obj = new Object();` — стандартный
 - `var obj = {};` — объектный литерал
- Объектно-литеральный синтаксис может быть использован для определения свойств объекта в момент его создания

Массивы

- Массивы – это специальный тип объектов
- Имеют свойство «length», значение которого всегда больше на единицу наибольшего индекса в массиве
- Два способа задания массивов:
 - `var a = new Array();`
 `a[0] = "собака";`
 `a[1] = "кошка";`
 - `var a = ["собака", "кошка"];`
- Массивы имеют набор встроенных методов:
 - `concat` возвращает новый массив с элементами, добавленными к нему
 - `pop` удаляет из массива и возвращает последний элемент
 - `push` добавляет один или несколько элементов в конец массива и др.

Функции

□ Объявление самой простой функции:

```
function add(x, y) {  
    var total = x + y;  
    return total;  
}
```

□ В JavaScript:

- функции могут принимать от 0 и больше именованных параметров
- тело функции может содержать любые выражения JavaScript и объявления внутренних переменных
- выражение return может быть использовано в любое время для прекращения выполнения функции и возврата значения
- если return отсутствует или не возвращает значения, функция возвращает undefined

Добавление на страницу

- Для добавления JavaScript-кода на страницу, можно использовать теги `<script></script>`
- JavaScript можно внедрять в страницу следующими способами:
 - расположение внутри тега
 - отделение от разметки
 - вынесение в отдельный файл

Область применения JavaScript

- Веб-приложения
- AJAX
- Comet
- Браузерные операционные системы
- Пользовательские скрипты в браузере
- Серверные приложения
- Мобильные приложения
- Виджеты
- Прикладное программное обеспечение
- Манипуляция объектами приложений
- Офисные приложения

Безопасность и JavaScript

- В целях безопасности JavaScript имеет два ограничения:
 - JavaScript-программы выполняются в песочнице (sandbox)
 - для JavaScript-кода применяется политика общего происхождения
- JavaScript имеет проблемы безопасности, связанные с:
 - межсайтовыми уязвимостями
 - ошибками в браузере, плагинах и расширениях
 - ошибками реализации песочницы