

01 - Введение в JavaScript

1. Кто такой JavaScript разработчик?

JavaScript разработчик – это специалист, который хорошо знает web программирование клиентской части веб-сайта. Можно сказать, что их задачей является создание привлекательного и удобного веб-дизайна и интерфейса, которые в свою очередь являются показателями качественного программного обеспечения. JavaScript программист должен хорошо владеть HTML разметкой, CSS, языком программирования JavaScript также фреймворком этого языка, такими как JQuery, AngularJS и др. еще он должен знать правила правильного удобного интерфейса.

2. Структура современной ИТ компании.

В современной ИТ компании, кроме JavaScript разработчика вы встретите следующих специалистов:

Front-end разработчик - программист клиентской части приложений.

Back-end разработчик - программист серверной части приложений.

HR менеджер (англ. *human resources management*) - менеджерами по персоналу.

PM менеджер (англ. *project manager*) - это специалист, чьей главной задачей является управление проектом в целом.

Вопрос: Кто принимает на работу ИТ специалистов?

3. Основы работы компьютерной сети.

Компьютерная сеть (англ. *computer network*) – это совокупность компьютеров и других устройств, соединенных линиями связи и обменивающихся информацией между собой в соответствии с определенными правилами – протоколом. Протокол играет очень важную роль, поскольку недостаточно только соединить компьютеры линиями связи.

Домен — это адрес сайта или определенная зона, которая имеет свое имя, не похожее ни на одно другое в системе доменных имен.

IP-адрес (англ. *internet protocol address*) — уникальный идентификатор (адрес) устройства (обычно компьютера), подключенного к локальной сети или интернету.

DNS (англ. *domain name system*) - система доменных имён. Это система, позволяющая преобразовывать символьные имена доменов в IP-адреса.

TCP (англ. *transmission control protocol*) - протокол управления передачей. Один из основных протоколов передачи данных интернета, предназначенный для управления передачей данных.

HTTP (англ. *hypertext transfer protocol*) - протокол обмена гипертекстовой информацией. Это протокол прикладного уровня, разработанный для обмена гипертекстовой информацией в сети Internet.

При работе в Internet для обслуживания HTTP-запросов используется 80 порт TCP/IP. Практика использования протокола такова, что клиент устанавливает соединение и ждет ответа сервера. После отправки ответа сервер инициирует разрыв соединения.

Вопрос: Может ли быть несколько сайтов на одном сервере?

Вопрос: Как сервер передает данные?

Практика: Создание тестового проекта (html файл + js файл в выбранной папке)

4. Что такое JavaScript.

JavaScript – это «безопасный» язык программирования общего назначения. JavaScript изначально создавался для того, чтобы сделать web-страницы «живыми». Программы на этом языке называются скриптами. В браузере они подключаются напрямую к HTML и, как только

загружается страничка – тут же выполняются. Программы на JavaScript – обычный текст. Они не требуют какой-то специальной подготовки.

JavaScript может выполняться не только в браузере, а где угодно, нужна лишь специальная программа – **интерпретатор**. Процесс выполнения скрипта называют «интерпретацией».

Важно: JavaScript чувствителен к регистру.

5. Компиляция и интерпретация

Компиляция – это когда исходный код программы, при помощи специального инструмента, другой программы, которая называется «компилятор», преобразуется в другой язык, как правило – в машинный код. Этот машинный код затем распространяется и запускается. При этом исходный код программы остается у разработчика.

Интерпретация – это когда исходный код программы получает другой инструмент, который называют «интерпретатор», и выполняет его «как есть». При этом распространяется именно сам исходный код (скрипт). Этот подход применяется в браузерах для JavaScript.

Вопрос: Чем же все-таки интерпретация отличается от компиляции?

6. Подключение и порядок исполнения JavaScript.

- Скрипты вставляются на страницу как текст в теге <script>, либо как внешний файл через <script src="путь"></script>
- HTML5 не требует обязательного указания атрибута type, пример: type="text/javascript".
- Если указан атрибут src, то содержимое тега <script> игнорируется.
- Чтобы подключить несколько скриптов, используйте несколько тегов.
- Специальные атрибуты async и defer используются для того, чтобы пока грузится внешний скрипт – браузер показал остальную (следующую за ним) часть страницы. Без них этого не происходит.
- Разница между async и defer: атрибут defer сохраняет относительную последовательность скриптов, а async – нет.
- Скрипт с defer сработает, когда весь HTML-документ будет обработан браузером.
- Атрибуты async/defer – только для внешних скриптов.
- При одновременном указании async и defer в современных браузерах будет использован только async, в IE9 – только defer (не понимает async).

Практика: Сделайте страницу, которая выводит «Я – JavaScript разработчик!».

Сначала подключите скрипт прямо в html документ, потом вынесите скрипт во внешний файл.

Полезные ссылки: <http://learn.javascript.ru/external-script>

7. Комментарии в коде.

JavaScript поддерживает два способа оформления комментариев. Любой текст между символами // и концом строки рассматривается как комментарий и игнорируется JavaScript. Любой текст между символами /* и */ также рассматривается как комментарий. Эти комментарии могут состоять из нескольких строк, но не могут быть вложенными. Следующие строки представляют собой корректные JavaScript-комментарии:

```
// Это однострочный комментарий.
```

```
/*
```

```
 * Это еще один комментарий.
```

```
 * Он располагается в нескольких строках.
```

```
*/
```

Не бойтесь комментариев. Чем больше кода в проекте – тем они важнее. Что же касается увеличения размера кода – это не страшно, т.к. существуют инструменты сжатия JavaScript, которые при публикации кода легко их удалят.

Практика: Добавьте комментарии к ранее созданному скрипту для вывода «Я – JavaScript разработчик!»

Полезные ссылки: <http://learn.javascript.ru/structure>

8. Структура кода, команды и их разделение

- В JavaScript всегда рекомендуется разделять команды точкой с запятой. Сейчас это, фактически, стандарт, которому следуют все большие проекты.
- Каждая новая команда начинается с новой строки.
- Логические блоки разделяются пустой строкой.
- Вложенные команды начинаются с отступа в 2 или четыре пробела.

Практика: Отредактируйте ваш скрипт и документ согласно правил.

Полезные ссылки: <http://learn.javascript.ru/structure>

9. Современный стандарт, "use strict"

Очень долго язык JavaScript развивался без потери совместимости. Новые возможности добавлялись в язык, но старые – никогда не менялись, чтобы не «сломать» уже существующие HTML/JS-страницы с их использованием.

Однако, это привело к тому, что любая ошибка в дизайне языка становилась «вмороженной» в него навсегда.

Так было до появления стандарта ECMAScript 5 (ES5), который одновременно добавил новые возможности и внёс в язык ряд исправлений, которые могут привести к тому, что старый код, который был написан до его появления, перестанет работать.

Чтобы этого не случилось, решили, что по умолчанию эти опасные изменения будут выключены, и код будет работать по-старому. А для того, чтобы перевести код в режим полного соответствия современному стандарту, нужно указать специальную директиву use strict.

Полезные ссылки: <http://learn.javascript.ru/strict-mode>

10. Переменные.

В процессе работы компьютерные программы манипулируют значениями, такими как число 3,14 или текст «Hello World». Когда в программе необходимо сохранить значение, чтобы использовать его позже, это значение присваивается (или сохраняется в) переменной. Переменная определяет символическое имя для значения и обеспечивает возможность получить это значение по имени.

- Переменные в JavaScript не имеют типа: переменной может быть присвоено значение любого типа и позднее этой же переменной может быть присвоено значение другого типа.
- Объявление переменных выполняется с помощью ключевого слова var.
- Технически, можно просто записать значение и без объявления переменной с использованием ключевого слова var, однако по ряду причин это не рекомендуется.
- В старом стандарте JavaScript разрешалось создавать переменную и без var, просто присвоив ей значение. В режиме "use strict" так делать уже нельзя.
- Вместе с объявлением можно сразу присвоить значение: var x = 10.
- На имя переменной в JavaScript наложены всего два ограничения:
 - имя может состоять из: букв, цифр, символов \$ и _.
 - первый символ не должен быть цифрой.

- Существует список зарезервированных слов, которые нельзя использовать для переменных, так как они используются самим языком, например: var, class, return, export и др.
- Регистр букв имеет значение. Переменные apple и AppLE – две разные переменные.
- Переменные, которые названы БОЛЬШИМИ_БУКВАМИ, являются константами, то есть никогда не меняются. Как правило, они используются для удобства, чтобы было меньше ошибок.

11. Взаимодействие с пользователем: alert, prompt, confirm.

В этом разделе мы рассмотрим базовые UI операции: alert, prompt и confirm, которые позволяют работать с данными, полученными от пользователя.

alert(аргумент) - принимает один аргумент и выводит на экран окно с сообщением и приостанавливает выполнение скрипта, пока пользователь не нажмет «ОК».

Окно сообщения, которое выводится, является **модальным окном**. Слово «модальное» означает, что посетитель не может взаимодействовать со страницей, нажимать другие кнопки и т.п., пока не разберется с окном.

prompt(title, default) - принимает два аргумента 'заголовок' и 'строка по умолчанию' соответственно и выводит модальное окно с заголовком title, полем для ввода текста, заполненным строкой по умолчанию default и кнопками OK/CANCEL. Вызов prompt возвращает то, что ввел посетитель – строку или специальное значение null, если ввод отменён.

confirm(question) - confirm выводит окно с вопросом question с двумя кнопками: OK и CANCEL.

Практика: Работа с переменными. Объявите две переменные: admin и name. Запишите в name строку "Василий". Скопируйте значение из name в admin. Выведите admin (должно вывести «Василий»).

Практика: Создайте страницу, которая запрашивает имя и выводит его.

Полезные ссылки: <http://learn.javascript.ru/variables>