



Урок 1

Основные понятия в веб-разработке

Что представляет собой веб-страница. Виды сайтов. Процесс разработки сайта. Гипертекст, теги и атрибуты. Структура HTML-документа. Основные теги оформления текста. Горячие клавиши для перемещения по документам.

[Как работает интернет](#)

[Браузеры](#)

[Что представляет собой веб-страница](#)

[Виды сайтов](#)

[Протоколы передачи данных](#)

[Схема HTTP-запроса страницы](#)

[Процесс разработки сайта](#)

[Гипертекст](#)

[Теги и атрибуты](#)

[Структура HTML-документа](#)

[Основные теги оформления текста](#)

[Заголовки](#)

[Параграфы](#)

[Теги выделения текста](#)

[Вложенные теги](#)

[Необходимые инструменты](#)

[Горячие клавиши](#)

[Практика](#)

[Домашнее задание](#)

[Дополнительные материалы](#)

[Используемая литература](#)

Как работает интернет

Прежде всего ответим на вопрос: что такое интернет? В Википедии находим следующее определение:

Интернёт (англ. Internet) – всемирная система объединенных компьютерных сетей, построенная на использовании протокола TCP/IP и маршрутизации пакетов данных. Интернет образует глобальное информационное пространство, служит физической основой для Всемирной паутины (WWW, World Wide Web) и множества других систем (протоколов) передачи данных. Часто упоминается как «Всемирная сеть» и «Глобальная сеть», в обиходе иногда употребляют сокращенное наименование «Инёт».

Другими словами, интернет состоит из множества домашних и корпоративных сетей, принадлежащих различным пользователям, компаниям и предприятиям, работающих по самым разнообразным протоколам, связанных между собой различными линиями связи, которые могут передавать данные по телефонным проводам, оптоволокну, через спутники и радиомодемы.

Интернет – это множество компьютеров по всему миру, объединенных в единую сеть, которые постоянно обмениваются какой-либо информацией.

Структура интернета напоминает паутину, в узлах которой находятся компьютеры, связанные между собой линиями связи. Как и объекты в реальной жизни, каждый компьютер имеет уникальный адрес, называемый IP-адресом. Существуют два типа IP-адресов:

- Постоянные, закрепленные за определенным компьютером.
- Динамические – присваиваются в тот момент, когда пользователь соединяется с интернетом.

Структура IP-адреса позволяет узнать, в какой стране и в каком городе находится компьютер пользователя. Таким образом, например, при настройке объявлений по контекстной рекламе можно задавать регионы показа объявления.

Пример IP-адреса: 192.168.1.2

Чтобы узнать свой IP-адрес, нужно зайти в «Панель управления» – «Сетевые подключения» – выбрать «Подключение по локальной сети» и перейти на вкладку «Поддержка».

Чтобы не запоминать сложные IP-адреса, придумали так называемые доменные имена.

Доменное имя – уникальное имя, которое данный поставщик услуг избрал себе для идентификации, например mail.ru или google.com.

Доменное имя может иметь несколько уровней. Домен первого уровня обычно определяет страну местоположения сервера (ru – Россия; ua – Украина; uk – Великобритания; de – Германия) или вид организации (com – коммерческие организации; edu – научные и учебные организации; gov – правительственные учреждения; org – некоммерческие организации). С недавнего времени стало возможным использование русскоязычных доменов (рф).

Браузеры

Браузер – программа, с помощью которой вы имеете возможность просматривать любые сайты. То есть браузер является проводником между всемирной сетью интернет и пользователем. Именно он позволяет получать доступ ко всему множеству доступной в интернете информации.

Существует большое количество браузеров. Большинство из них практически одинаково отображают информацию, за исключением Internet Explorer, который некоторые теги и свойства стилей отображает иначе – к примеру, ширину блоков. Эти особенности необходимо учитывать при верстке, но подробнее об этом – в конце курса.

Что представляет собой веб-страница



Веб-страница сайта состоит из различных блоков:

- Шапка (header), в которой могут размещаться логотип компании, название сайта, телефоны организации и проч.
- Вертикальная и/или горизонтальная навигация по сайту (меню).
- Основное содержимое (content).
- Нижний колонтитул (footer), где размещается дополнительная информация: автор сайта, счетчики посещаемости.

Виды сайтов

По содержанию

Статические – содержимое подготавливается заранее и выдается пользователю в том виде, в котором хранится на сервере.

Динамические – содержимое генерируется при помощи серверных языков программирования.

По схеме представления информации

Коммерческие – сайты компаний, интернет-магазины и т.д.

Информационные – доносят до пользователя какую-либо информацию.

Веб-сервисы (порталы) – поисковые системы, электронная почта, форумы, социальные сети.

По размеру

Фиксированной ширины – задается фиксированная ширина сайта (в настоящее время ширина обычно составляет 1000 пикселей, т.к. разрешение современных мониторов по ширине начинается от 1024 пикселей). Тогда все блоки сайта будут выглядеть так, как было задумано, и не будут изменять размер в зависимости от разрешения монитора или размера экрана браузера.

«Резиновые» – ширина строго не задается, а изменяется в зависимости от размера экрана браузера или разрешения монитора.

Адаптивная верстка/тип макета – дизайн, который подстраивается (адаптируется) под размер экрана, в том числе может происходить перестройка блоков с одного места на другое или их замена блоками, отображаемыми только при определенном разрешении.

Протоколы передачи данных

Взаимодействие компьютеров, серверов, маршрутизаторов, коммутаторов определяется протоколами. Каждый протокол – четкий и определенный набор правил и соглашений, предписывающий, каким образом производится обмен и обработка информации.

Наиболее известные протоколы, используемые в сети интернет:

- HTTP (HyperText Transfer Protocol) – протокол передачи гипертекста. Используется при пересылке веб-страниц с одного компьютера на другой.
- HTTPS (HyperText Transfer Protocol Secure) – тоже протокол для передачи гипертекста, но использует дополнительное шифрование данных для более безопасной передачи информации.
- FTP (File Transfer Protocol) – протокол передачи файлов со специального файлового сервера на компьютер пользователя. FTP дает абоненту возможность обмениваться двоичными и текстовыми файлами с любым компьютером сети. Установив связь с удаленным компьютером, пользователь может скопировать файл с удаленного компьютера на свой и наоборот.

Схема HTTP-запроса страницы



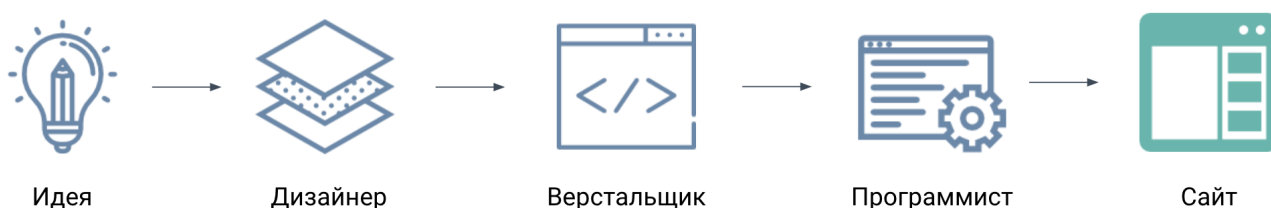
Пользователь набирает в браузере адрес нужного ему сайта, после чего посылается HTTP-запрос серверу. Сервер определяет тип страницы по расширению запрашиваемого файла.

Если расширение `.html`, это статическая страница, и сервер сразу формирует HTML-страницу и отправляет ее обратно пользователю. Если расширение, к примеру, `.php`, то в этом случае страница динамическая. Сервер подключает интерпретатор `php`, который, в свою очередь, может обратиться в базу данных и получить оттуда необходимые данные.

Результат – HTML-код, который веб-сервер посылает обратно на компьютер пользователя. Сгенерированный HTML-код веб-сервер упаковывает в HTTP-пакет, который и передается по сети обратно клиенту. Полученный HTTP-ответ попадает в браузер клиента, который извлекает из него HTML-код и генерирует на его основе графическое представление запрошенной страницы.

Простым языком: если мы обратились к HTML-файлу (`.html`), браузеру этого документа достаточно и Apache в случае такого запроса возвращает нам HTML-разметку. Если используется другое расширение файлов, например, `.php`, в данном файле может содержаться программный код, который должен быть обработан (выполнен) интерпретатором языка. На выходе `php`-интерпретатор должен нам выдать HTML-код, потому что именно его понимает браузер.

Процесс разработки сайта



На схеме представлен процесс разработки сайта на примере веб-студии, где определенные функции выполняет определенный человек.

Когда заказчик приходит со своей идеей сайта в веб-студию, с ним начинает работать менеджер проекта. Очень часто бывает, что идея у заказчика есть, но она настолько размыта, что тяжело представить на этом этапе, что в итоге хочет получить клиент.

В разных организациях используется свой подход к выяснению потребностей заказчика. Один из вариантов – специальная анкета, после составления которой можно уже представить, как будет выглядеть будущий сайт и каким функционалом обладать.

Далее менеджер составляет два технических задания: одно дизайнеру на составление макета сайта (дизайна), а другое – программисту с указанием, какой функционал необходимо реализовать в этом проекте.

После этого верстальщику отправляется макет сайта от дизайнера и различные модули от программиста, и после того, как сайт будет сверстан, копирайтер наполняет его контентом, т.е. заполняет товарами, пишет статьи и т.д. В это же время seo-специалист начинает заниматься продвижением сайта.

Гипертекст

Гипертекст – термин, введенный Тедом Нельсоном в 1965 году для обозначения «текста ветвящегося или выполняющего действия по запросу». Обычно гипертекст представляется набором текстов, содержащих узлы перехода между ними, которые позволяют избирать читаемые сведения или последовательность чтения.

Общеизвестным и ярким примером гипертекста служат веб-страницы – документы HTML (язык разметки гипертекста), размещенные в сети. В более широком понимании термина гипертекстом является любая повесть, словарь или энциклопедия, где встречаются отсылки к другим частям данного текста, имеющим отношение к данному термину. В компьютерной терминологии гипертекст – текст, сформированный с помощью языка разметки, потенциально содержащий в себе гиперссылки.

Теги и атрибуты

Язык HTML – язык тегов. Теги описывают структуру HTML-документа. Они оформляются угловыми скобками <имя тега>, между которыми прописывается имя тега. Теги HTML-документа предназначены для управления конструкциями разметки: заголовками, абзацами, списками, таблицами и картинками.

Теги бывают парные и одиночные.

Парные теги сначала открываются, потом закрываются. Все, что описывают парные теги, находится внутри них. Так, тег делает текст полужирным. Вот как это выглядит:

```
<b>Какое прекрасное утро</b>
```

Одиночные теги стоят сами по себе и, как правило, не меняют контент вокруг себя. Пример:

```
<br> -- тег переноса строки  
<hr> -- тег горизонтальной линии
```

Атрибуты тегов определяют какие-либо дополнительные, уточняющие параметры того или иного тега. Вы можете указать внутри тега столько атрибутов, сколько вам необходимо. Атрибуты разделяются пробелами.

`<название_тега атрибут1="значение_атрибута1"> текст внутри тега </название_тега >`

Пример:

```
<a href="contacts.html">Контакты</a>

```

Структура HTML-документа

Каждая страница в HTML-документе состоит из трех обязательных элементов:

1. Тип документа.
2. Раздел `<head></head>` с технической информацией о странице: заголовок, описание, ключевые слова для поисковых машин, кодировка. Введенная в нем информация в основном не отображается в окне браузера, однако содержит данные, которые указывают браузеру, как следует обрабатывать страницу.
3. Раздел `<body></body>`, где располагаются все элементы, которые видит пользователь. Именно с этим разделом мы в основном и будем работать.

DOCTYPE отвечает за корректное отображение веб-страницы браузером. DOCTYPE определяет не только версию HTML (например, html), но и соответствующий DTD-файл в интернете.

Элементы, находящиеся внутри тега `<html>`, образуют дерево документа, так называемую объектную модель документа, DOM (document object model). При этом элемент `<html>` является корневым.

Пример структуры HTML 5:

```
<!doctype html>
<html>
<head>
  <title>Hello HTML</title>
  <meta charset="UTF-8">
</head>
<body>
...
</body>
</html>
```

Основные теги оформления текста

Заголовки

Как и в газетах и журналах, в HTML-странице любая статья или новость должна начинаться с заголовка. Предусмотрено 6 уровней заголовков: первый из них будет отображен самым крупным шрифтом, а далее чем выше уровень, тем меньше будет размер шрифта.

.Для отображения заголовков существует тег <h> и указывается цифра от 1 до 6, которая соответствует уровню заголовка. Тег заголовка – парный, не забудьте его закрыть.

<h1>Заголовок первого уровня</h1>

<h2>Заголовок второго уровня</h2>

<h3>Заголовок третьего уровня</h3>

<h4>Заголовок четвертого уровня</h4>

<h5>Заголовок пятого уровня</h5>

<h6>Заголовок шестого уровня</h6>

Параграфы

После заголовка обычно следует какой-нибудь текст, который необходимо заключать в параграфы, или абзацы. При составлении документа выделяйте блоки текста в отдельные параграфы, как это сделано в книгах – в противном случае может получиться сплошной текст, который очень трудно будет читать посетителю сайта.

В HTML для параграфов используется парный тег <p>, и внутри него помещается тот текст, который нужно отобразить. Пример:

```
<p>Здесь мы напишем первый параграф</p>
```

Теги выделения текста

Иногда необходимо выделить слово, словосочетание, предложение или целый участок текста, чтобы привлечь внимание читателя или поискового робота. Чтобы это сделать, можно выделить фрагмент жирным, курсивом или подчеркнуть.

Аккуратнее с подчеркиванием текста: по стандартам принято, что подчеркнуты гиперссылки, и многие пользователи уже настолько к этому привыкли, что, видя подчеркнутый текст, хотят «кликнуть» по этому участку текста, ожидая перехода на другую страницу.

Теги и внешне делают одно и то же: выделяют текст полужирным. Разница в том, что указывает на важность текста, а просто делает текст полужирным. Так, текст в тегах устройство для чтения текста вслух будет выделять интонацией, а – нет.

Аналогично с тегами и <i>, делающими текст курсивным: указывает на важность текста, а <i> нет.

<small> уменьшает размер шрифта на единицу по отношению к обычному тексту.

`<sub>` используется для создания нижних индексов. Сдвигает текст ниже уровня строки, уменьшая его размер.

`<sup>` используется для создания степеней и верхних индексов. Сдвигает текст выше уровня строки, уменьшая его размер.

`<ins>` выделяет текст в новой версии документа, подчеркивая его.

`` перечеркивает текст. Используется для выделения текста, удаленного из документа.

`<code>` служит для выделения фрагментов программного кода. Отображается моноширинным шрифтом.

`<pre>` позволяет вывести текст на экран, сохранив изначальное форматирование. Пробелы и переносы строк при этом не удаляются.

`<q>` используется для выделения коротких цитат. Браузерами заключается в кавычки.

Вложенные теги

Очень часто при верстке веб-страниц необходимо вкладывать одни теги в другие. Эта вложенность может достигать несколько уровней. Поэтому следует запомнить одно простое правило: закрывать теги необходимо именно в той последовательности, в которой вы их открывали. Первый открытый тег закрывается последним.

```
<p>Использовать вложенность тегов
  <b>легко
    <i>и просто</i>
  </b>
</p>
```

Тег `<i>` был открыт последним – закрываем его первым, а далее уже все остальные теги по очереди.

Необходимые инструменты

- Чтобы начать создавать свои веб-странички, необходим текстовый редактор. Можно воспользоваться встроенным в Windows блокнотом, но для начала удобнее использовать более функциональные текстовые редакторы Visual Studio Code, Sublime text, Notepad++ или Brackets.
- Браузер. Выберите для себя наиболее удобный браузер. Я рекомендую пользоваться браузером Google Chrome.

Горячие клавиши

Alt + tab – перемещение между открытыми окнами.

Ctrl + z – отменить действие.

Ctrl + y – вернуть действие.

Ctrl + x – вырезать.

Ctrl + c – копировать.

Ctrl + v – вставить.

Ctrl + s – сохранить.

Ctrl + a – выделить все.

Ctrl + tab – перемещение м/у открытыми вкладками.

F5 / Ctrl + R – обновить страницу.

F2 – переименовать.

Shift + Tab – вернуть одну табуляцию назад.

Shift + home – выделить с указанного места до начала строки.

Shift + end – выделить с указанного места до конца строки.

Shift + стрелка – при зажатом Shift выделять в указанную стрелками сторону.

Практика

- Создаем на любом из жестких дисков новую папку.
- Заходим в эту папку, щелкаем правой кнопкой мыши, выбираем пункт «Создать текстовый документ» с расширением «.txt».
- Переименовываем этот файл, даем ему любое имя, а расширение меняем на «.html».
- Щелкаем по файлу правой кнопкой мыши, выбираем пункт «Открыть с помощью», выбираем текстовый редактор и открываем.
- Напишем теперь первое приветствие миру. Переходим в текстовый редактор и набираем: `<h1>Hello world!</h1> <p>I'm here</p>`.
- Чтобы посмотреть результат в браузере, щелкаем по этому файлу.

Домашнее задание

Создание первой html страницы

1. Создать главную страницу вашего сайта
2. Настроить редактор кода
3. Включить автосохранение файлов
4. Изменить размер шрифта, на удобный для вас
5. Включить автоформат кода, при сохранении
6. Добавить структуру html документа (Для всех последующих страниц также нужно будет добавить структуру html документа, при создании)
7. Создать заголовок с произвольным текстом, не более 3-х слов(h1)

8. Создать параграф с произвольным текстом
9. Создать заголовок, меньше уровнем чем предыдущий (h3), добавить не более 3-х слов
10. Создать заголовок (h4), добавить не более 5-ти слов.
11. Добавить параграф с произвольным текстом не более 15 слов
12. Создать заголовок (h3) добавить не более 3-х слов
13. Добавить параграф с произвольным текстом, не более 10 слов
14. Создать заголовок (h3), добавить не более 3-х слов
15. Добавить параграф с произвольным текстом, не более 10 слов
16. Добавить разделительную линию (hr)
17. Создать параграф с текстом “Все права защищены”

Используемая литература

Для подготовки данного методического пособия были использованы следующие ресурсы:

1. <http://htmlbook.ru/samhtml/struktura-html-koda>.
2. <https://ru.wikipedia.org/wiki/%D0%91%D1%80%D0%B0%D1%83%D0%B7%D0%B5%D1%80>.
3. <http://html.manual.ru/book/html/hierarchy.php>.
4. <https://webref.ru/>.