

Содержание:

Введение

Актуальность. Собственно, появление html и ознаменовало появление интернета в том виде, в котором мы его знаем сейчас, приведя к началу бурного роста сети и количества сайтов в ней.

HTML - это сокращение от англ. "HyperText Markup Language", что в переводе на русский означает "язык гипертекстовой разметки". HTML - это стандартный язык, который и по сей день используется для создания веб-страниц, размещаемых в интернете.

HTML – быстро развивающийся язык разметки веб-документа, который является основой большинства популярных веб-сайтов. Не каждому известно, что HTML5 – это обновление языка HTML, эффективность которого, при создании веб-приложении различных типов и организации представления контента на сайте, на порядок выше предыдущих версий.

Цель работы – изучить языки гипертекстовой разметки.

Достижение указанной цели определило постановку и решение следующих **задач**:

- исследовать историю языков разметки;
- изучить языков разметки на примере HTML;
- привести пример использования HTML.

Предметом исследования можно назвать языки гипертекстовой разметки.

Объектом исследования является язык HTML.

Теоретико-методологическую основу исследования составили научные труды, посвященные анализу языков гипертекстовой разметки. При разработке и решении поставленных задач использовались методы сравнительного анализа, а также группировки.

Структура работы. Работа включает введение, две главы, заключение и список литературы. В первой главе рассматриваются история языков гипертекстовой разметки. Во второй главе рассматриваются основы языка HTML. В третьей главе

приведен пример использования языка HTML.

Научно-методической основой работы служат труды отечественных и зарубежных ученых в области программирования. При выполнении работы использовалась научно-методическая литература, публикации в периодической печати и научных изданиях, материалы Интернет-порталов.

Глава 1. Теоретические основы языков разметки

1.1. История языков разметки

Начало истории HTML было положено Чарльзом Гольдфарбом в 1969 г., когда он создал для компании IBM прототип языка разметки технической документации GML (Generalized Markup Language - обобщенный язык разметки), которому в 1986 г. был придан статус международного стандарта SGML (**S**tandard GML). Этот обобщенный язык предназначен для построения систем логической, структурной разметки любых разновидностей текста. Структурная разметка в данном случае означает, что управляющие коды, вносимые в текст при такой разметке, не несут никакой информации о форматировании документа, а лишь указывают границы и соподчиненность его составных частей[1].

После создания Тимом Бернес-Ли в 1989 г. первого браузера многие коммерческие компании подхватили эту инициативу и стали выпускать свои браузеры и к ним свои собственные расширения языка HTML.

Каждая из компаний стремилась сделать свой браузер самым лучшим и удобным, и вводила в его интерпретатор собственные новые элементы, отвечающие за визуальную разметку страниц. Делалось это с целью привлечь пользователей и разработчиков веб-страниц к собственному продукту. Тем самым основная прелесть HTML - "платформенно-независимое отображение документа" - была утеряна[2].

С начала 1994 г. наблюдается всплеск активности фирмы Netscape. Основатели этой компании, в прошлом разработчики в NCSCA (**N**ational **C**enter for **S**uper**C**omputer **A**pplications - Национальный Центр Суперкомпьютерных приложений), были создателями первого графического браузера Mosaic и имели большой опыт в разработке графических браузеров.

Программа Mosaic была единственной в то время браузером поддерживающим графические возможности. Она была разработана в Национальном центре суперкомпьютерных приложений США (National Center for Supercomputer Applications - NCSA), там же, кстати, была разработана WWW (World Wide Web). По этой причине никаких противоречий между официальными стандартами и их реализацией в браузерах тогда не существовало[3].

Для повышения популярности собственного продукта - Netscape Navigator, были разработаны "Расширения Netscape для HTML". Это дало свои плоды, - к концу 1995 г. компания завоевала до 90% рынка браузеров.

С небольшим запозданием на этот рынок пришла фирма Microsoft. Версия Internet Explorer 2.0 не представляла ничего выдающегося по сравнению с той же версией Netscape Navigator (которая действительно была второй), включавшей в себя интерфейс подключаемых модулей, поддержку Java-апплетов (программ на языке Java, подробнее смотрите раздел **Java**), встроенный язык сценариев JavaScript, возможности разбивки окна браузера на фреймы и многое другое.

В результате назревала проблема несовместимости браузеров. Динамично растущей Сети грозил хаос.

В апреле 1994 года был образован Консорциум W3C (World Wide Web Consortium). Так как официальной спецификации HTML 1.0 не существовало, W3C начал заниматься подготовкой спецификации HTML следующей версии. Но чтобы стандартная версия отличалась от всех предыдущих, ей сразу присвоили номер 2.0. Разработка спецификации HTML 2.0 шла не спеша и лишь в сентябре 1995 года она была утверждена. Из больших дополнений был добавлен лишь механизм форм для отсылки информации с компьютера пользователя на сервер[4].

Тем временем Консорциум W3 в параллель со спецификацией 2.0 занимался обсуждением HTML 3.0. Она была предложена в марте 1995 года. Третья версия предлагала много новых возможностей: поддержка таблиц, обтекание изображения текстом, отображение сложных математических формул, примечания. Поддержка этого стандарта браузерами того времени была не удовлетворена.

В 1996 г. W3C объявил о сотрудничестве с лидерами рынка для установления стандартов совместимости программного обеспечения. Была предложена версия HTML 3.2, которая стала компромиссным решением - попыткой примирить Microsoft и Netscape и объединить их оригинальные решения в одно целое[5].

Если первая версия языка (HTML 1.0) была направлена на представление языка как такового, где описание его возможностей носило скорее рекомендательный характер, вторая версия языка (HTML 2.0) фиксировала практику использования конструкций языка, версия + (HTML +, она же "замороженная" в силу недоработанности HTML 3.0) представляла новые возможности, расширяя набор элементов HTML в сторону отображения научной информации и таблиц, а также улучшение стиля компоновки изображений и текста, то версия HTML 3.2 призвана упорядочить все нововведения и согласовать их с существующей практикой. Кроме того, в версии 3.2 снова делается попытка формализации интерфейса пользователя гипертекстовой распределенной системы. По сравнению с версией 2.0, HTML 3.2 позволяет реализовать отображение таблиц, выполнение мобильных кодов, обтекание графики текстом, а также отображение верхних и нижних индексов.

Успех W3C оказался спорным, т.к. компании продолжали придумывать новые расширения к языку HTML и внедряли их поддержку в свои браузеры. Разработчики браузеров обвиняют W3C из-за того, что он является своеобразным тормозом в развитии HTML, ввиду отсутствия давления на него со стороны пользователей, и не спешит принять некоторые их расширения, в то время как многие веб-разработчики давно и успешно их применяют на своих страницах. С помощью версии HTML 4.0 W3C предпринял очередную попытку установить стандарты.

В 1997 г. Консорциум W3C опубликовал спецификацию языка HTML 4.0, в которой указано, какие из элементов устарели и не рекомендуются к дальнейшему применению (APPLET, BASEFONT, CENTER, DIR, FONT, ISINDEX, MENU, S, I, STRIKE и U), а также новые элементы и атрибуты (ABBR, ACRONYM, BDO, BUTTON, COLGROUP, DEL, FIELDSET, FRAME, FRAMESET, IFRAME, INS, LABEL, LEGEND, NOFRAMES, NOSCRIPT, OBJECT, OPTGROUP, PARAM, SPAN, TBODY, TFOOT, THEAD и Q), которые предлагается использовать в дальнейшем[\[6\]](#).

При этом W3C обозначил тенденцию развития языка HTML все более в сторону логического форматирования содержимого. Была дана возможность идентификации практически любого элемента или группы элементов по имени, описаны основные атрибуты для событий. Однозначно указано на платформенную независимость HTML, другими словами, было подчеркнуто, что HTML не является языком визуальной верстки текстов, а является языком логической разметки. Почти все атрибуты, определяющие представление документа HTML (цвета, выравнивание, шрифты, графика и т.д.), являются нежелательными, взамен рекомендуется использовать таблицы стилей и классы.

Атрибуты **Id** и **Class** позволяют авторам назначать элементам информацию об имени и классе для таблиц стилей, ссылок, скриптов, объявления объектов, общей обработки документа и т.д., а также ожидается, что задачу представления таблиц в ближайшем будущем возьмут на себя таблицы стилей. HTML 4.0 поддерживает более обширный набор описаний клиентских устройств, так что авторы могут писать таблицы стилей в зависимости от устройств. Многие элементы теперь имеют атрибуты-обработчики событий, это легло в основу повышения интерактивности документов, дало возможность пользователям динамически изменять вид и содержание документов.

Появилась возможность говорить о надстройке к языку HTML - **Dynamic HTML**, позволяющей создавать и визуализировать интерактивные web-страницы без помощи подключаемых модулей, не увеличивая трафика (кол-ва передаваемых по Сети данных). Применение новых решений направлено именно на снижение кол-ва обращений к серверу, т.е. снижение трафика, что так необходимо в условиях роста популярности различных переносных устройств доступа в сеть.

Кроме возможностей разметки текста, включения мультимедиа и формирования гипертекстовых связей, уже существовавших в предыдущих версиях HTML, в версию 4.0 включены дополнительные средства работы с мультимедиа, языки программирования, таблицы стилей, упрощенные средства печати изображений и документов, которые становятся более доступными для всех пользователей HTML 4.0. Эти дополнения служат интернационализации WWW и распространению ее по всему миру. Кроме того, для управления сценариями просмотра страниц web-сайта можно использовать языки программирования этих сценариев типа **JavaScript**, **Java** и **VBScript**.

В 2014 году был завершен стандарт HTML 5, который кардинально отличается от прошлых версий HTML.

Практически с момента основания W3C параллельно с выпуском новых спецификаций HTML шла разработка нового языка XML (eXtensible - расширяемый), который в будущем должен заменить HTML. Пока еще мало кто из непрофессиональных авторов web-страниц пользуется этим языком разметки.

Но уже сейчас стоит соблюдать некоторые правила этого языка (такие как использование парных тэгов и внешнее определение форматирования), поскольку XML из свода правил, из не слишком понятной и сложной для рядового пользователя технологии "дорос" к 2000г. до четырех вариантов реализации[7]:

- Собственно XML как правила построения настраиваемого и расширяемого языка разметки
- XHTML 1.0 (eXtensible HTML) - рекомендован к использованию 26 января 2000г. как переопределение спецификации HTML 4.01 в терминах XML
- XHTML 1.1 - модульный подход, ориентированный на быстрое создание интерпретаторов языка с любым доопределением или уменьшением дескрипторов исходного
- XHTML Basic - облегченная версия, рассчитанная на мобильных пользователей (по оценкам W3C, в 2002г. 75% информации в Internet будет просматриваться с мобильных клиентов: телефоны, карманные компьютеры и т.п.).

1.2. Языки разметки

Язык разметки (текста) — набор символов или последовательностей, вставляемых в текст для передачи информации о его выводе или строении. Текстовый документ, написанный с использованием языка разметки, содержит не только сам текст (как последовательность слов и знаков препинания), но и дополнительную информацию о различных его участках — например, указание на заголовки, выделения, списки и т. д. В более сложных случаях язык разметки позволяет вставлять в документ интерактивные элементы и содержание других документов.

HTML (от англ. Hypertext Markup Language — «язык разметки гипертекста») — это стандартный язык разметки документов в Интернете. Все веб-страницы создаются при помощи языка HTML (или XHTML). Текстовые документы, содержащие код на языке HTML (такие документы традиционно имеют расширение «html» или «htm»), обрабатываются браузером. Интернет-обозреватель, или браузер, обычно предоставляет пользователю удобный интерфейс для запроса веб-страниц, их просмотра (и вывода на иные внешние устройства) и, при необходимости, отправки введённых пользователем данных на сервер. Наиболее популярными на сегодняшний день браузерами являются Internet Explorer, Mozilla Firefox, Opera и Safari.

С помощью языка HTML вы можете: сделать текст полужирным, курсивным или подчёркнутым; вставить специальные символы (выходящие за рамки ASCII символы пунктуации, математические символы, греческие и готические буквы, стрелки и т.п.); поменять начертание, цвет шрифта; выровнять текст по центру, левому/правому краю, по ширине; оформить текст как гиперссылку на другую

страницу или файл; нарисовать таблицу.

XML (англ. eXtensible Markup Language -расширяемый язык разметки). Свое название XML получил по той причине, что в нем нет фиксированного формата, как в HTML. В то время как язык HTML ограничивается набором твердо закрепленных тегов, пользователи XML могут создавать свои собственные тэги, которые бы отвечали тематике документа. Таким образом, XML — это метаязык. Этот язык используется в качестве средства для описания грамматики других языков и контроля за правильностью составления документов.

Документ XML выглядит во многом похожим на HTML. В XML существуют открывающие, закрывающие и пустые тэги. Однако, в отличие от HTML, правила относительно тегов более строгие. Например, смысл тега зависит от регистра, а каждый открывающий тег должен во всех случаях иметь парный закрывающий тег. Кроме того, теги в документе могут быть вложены друг в друга[8].

XML служит для хранения, передачи и обмена данными. С помощью XML вы можете осуществить обмен данными даже между несовместимыми системами. В настоящее время компьютеры и базы данных хранят информацию в самых разнообразных и несовместимых форматах. Преобразование данных в XML формат сделает данные доступными для самых различных типов программ[9].

Wiki — веб-сайт, структуру и содержимое которого пользователи могут сообщать изменять с помощью инструментов, предоставляемых самим сайтом. В идеале в Wiki пользователь имеет права на редактирование любой страницы и создания своих. При этом не обязательно знать какие-либо сложные языки разметки. Для форматирования текстов может использоваться Wiki-разметка, позволяющая создавать функциональный и корректный гипертекст даже тем, кто не знает html. Чтобы отредактировать текст в Wiki, обычно достаточно нажать кнопку «редактировать» или даже просто дважды кликнуть по тексту. Внесение правок не вызовет никаких проблем и у новичка.

Wiki-разметка максимально проста: основные её правила, достаточные для редактирования и создания новых материалов, можно узнать за считанные минуты. В Wiki сохраняются все изменения, произошедших со всеми статьями с момента их создания. В любой момент можно сравнить две версии одной и той же статьи, увидеть, какие изменения были внесены, откорректировать страницу снова или просто вернуться к старой версии. Исправить нанесенный ущерб проще, чем его нанести. Ну и, конечно же, у администраторов всегда есть возможность

заблокировать изменение той или иной страницы или разрешить редактирование только определенным пользователям. Одной из особенностей Wiki является гипертекстовость: связь страниц и подразделов сайта через контекстные гиперссылки.

BBCode (от англ. Bulletin Board Code) — язык разметки, используемый для форматирования сообщений на многих электронных досках объявлений (BBS), форумах, блогах, гостевых книгах и чатах. При создании сообщений, текст которых необходимо отформатировать, окружают специальными BBCode тегами — которые очень похожи на HTML-теги, но отличаются от последних тем, что заключены в квадратные скобки (“[и]”). BBCode был придуман с целью предоставить более простой, безопасный и ограниченный по сравнению с HTML способ форматирования сообщений. Первоначально на многих форумах использовалась HTML-разметка, что могло приводить к побочным эффектам: искажению разметки на странице или же к запуску JavaScript-сценариев[\[10\]](#).

Отличие BBCode от HTML в том, что язык разметки HTML можно использовать в любом статическом сайте — его по умолчанию понимают и отображают правильно все известные браузеры. С BBCode другая ситуация. Он может использоваться только под управлением какого-нибудь движка — браузеры не воспринимают чистый BBCode. Перед отображением страницы движок форума производит разбор текста, затем преобразует его в HTML-код.

Textile — простой язык разметки, позволяющий осуществить форматирование текста без HTML-кодирования, подобно BBCode, но обладающий большими возможностями. Textile используется в системе управления содержимым Textpattern, также встроен в некоторые Wiki. Textile автоматически преобразует текст в разметку, отслеживая абзацы в тексте. В тексте также можно непосредственно использовать HTML-разметку, когда возможностей Textile не хватает для необходимого форматирования в документе.

Глава 2. Изучение языков разметки на примере HTML

2.1.Преимущества и недостатки языка HTML

Использование языка программирования нового поколения позволяет создавать веб-документы с более простой структурой и чистым кодом. В отличие от предыдущей версии, пятое поколение языка программирования использует вместо контейнеров div семантические элементы `nav`, `article`, `header`, `footer`, `aside` и `section`. Ко всему прочему, HTML5 существенно облегчает работу с файлами мультимедиа: так, пользователи могут слушать музыку и смотреть видео прямо в браузере без использования Adobe Flash и прочих приложений. Дополнительные программы не потребуются и при размещении фотографий и прочей графики. Метаморфозы затронули также процесс ввода данных, который в пятом поколении языка программирования стал намного проще и безопаснее[\[11\]](#).

Если в предыдущей версии система сообщала об ошибке ввода данных постфактум, то теперь пользователь будет оповещён незамедлительно, что снимет массу проблем, связанных с отправкой неверно составленных документов. Что касается внешнего вида, то новый язык программирования существенно расширяет возможности оформления сайтов и приложений, за счёт чего html 5 игры и сайты станут ещё более оригинальными и интересными. Благодаря новой функции Canvas, которая реализована в пятой версии веб-языка, можно будет размещать на сайтах небольшие html 5 игры, анимацию и динамическую графику без использования дополнительных программ. К тому же данная функция позволяет создавать качественно новые html 5 игры и пользовательские интерфейсы[\[12\]](#).

Одним из тонких мест нового поколения веб-языка является защита данных. При использовании HTML 5 на винчестере пользователя долгое время хранится большой объём данных, что позволяет злоумышленникам осуществлять несанкционированный доступ к ресурсам компьютера с целью сбора конфиденциальной информации. Некоторая уязвимость HTML 5 имеет место быть и в сфере игр. Так, контент не может похвастаться высокой производительностью, платформы весьма ограничены в своём применении, а на создание контента с использованием HTML 5 требует больших затрат ресурсов и времени[\[13\]](#).

Ко всему прочему, до недавнего времени поддержкой HTML 5 могли похвастаться далеко не все браузеры. Несмотря на то, что сегодня ситуация в корне изменилась, и теперь абсолютное большинство программ распознают пятое поколение языка программирования, всё же остаются единичные экземпляры, которые с ним до сих пор не знакомы. Самым совместимым с HTML 5 считается браузер Chrome, а именно его 18-я версия. Программа, создателем которой является Google, имеет модуль Canvas 2D, который существенно облегчает работу в ресурсами, разработанными с помощью HTML 5. Впрочем, недостатки новой версии языка программирования не

снижают его популярности даже среди таких гигантов, как YouTube, Google и т.д., которые активно осваивают данную веб-технологию.

Огромный вклад в расширение практического использования нового языка программирования сделала корпорация Google, которая любезно предоставила свои стартовые страницы, чтобы рекламировать на них html 5 игры, сайты и веб-приложения. Ярким примером использования расширенных возможностей языка является страничка <http://www.google.com.hk/intl/zh-CN/landing/shuixia/>, которая предназначена для поздравления жителей Гонконга. Если пощёлкать компьютерной мышкой ниже воторлинии, то обычная стартовая страница Google превратится в островок удовольствия.

Ещё один пример применения возможностей HTML 5 в Google – это визуализация главной странички в форме аркадной игрушки в духе Кибериады, что было сделано в честь дня рождения Станислава Лемма. Для того, чтобы освоить весь безграничный потенциал нового языка Интернета, необходимо пройти курс обучения. Желательно, чтобы обучение происходило в игровой форме. Такую уникальную возможность предоставляет сайт <http://html5game.ru/>, который радуется пользователей с самой первой страницы – наводя мышку на шапку, можно с удовольствием наблюдать, как она разлетается на сотни шариков, а потом вновь возвращается в своё исходное состояние[14].

В учебных целях сайт предлагает широкий инструментарий по созданию html 5 игр, а это, как известно, лучший способ освоить новый язык программирования. Немалую обучающую ценность несёт в себе графический редактор <http://www.picozu.com/editor>, который позволяет рисовать, пользуясь инструментарием, разработанным в HTML5.

2.2. Правила построения и интерпретации HTML

Теговая модель определяет документ как некую совокупность элементов, каждый из которых окружен тегами. Теги, по своему значению, близки к представлению скобок «begin/end» в универсальных языках программирования, которые задают области действия имен локальных переменных, а именно динамику распределения памяти блока. Теги определяют область действия неких правил интерпретации текстовых элементов документа. Контейнерами принято называть элементы документа, которые размечены с помощью HTML. В данном виде может быть записана схема расположения элемента текста в формате HTML:

"элемент":= <"имя элемента" "список атрибутов">, сущность элемента

Конструкция перед сущностью элемента называется тегом начала документа, а конструкция, расположенная после сущности элемента, -тег конца элемента.

Структура гипертекстовой сети задается гипертекстовыми ссылками.

Атрибуты всегда помещаются в начальном теге элемента HTML. Значения атрибутов всегда полезно заключать в кавычки. Наиболее широко используются двойные кавычки, но одиночные кавычки также допустимы.

В некоторых редких ситуациях, когда, например, значение атрибута само содержит кавычки, необходимо использовать одиночные кавычки:

```
<html>
<body>
<abbr title='проект "Интернет-Университет
Информационных Технологий" - INTUIT.ru'>ИНТУИТ</abbr>
</body>
</html>
```

Кроме атрибутов, записываемых вышеописанным способом, для некоторых элементов определены специальные флаги, которые просто указываются как

<тег имя_флага>.

```
<html>
<body>
<p>Попробуйте исправить текст в этих полях ввода</p>
<form action="index.php" method="GET">
<input type="text" name="blocked"

value="Пример поля ввода, у которого указан флаг readonly"
readonly size="100"><br/><br/>
<input type="text" name="unblocked"
value="Обычное поле ввода у которого нет флагов"
size="100">
</form>
</body>
</html>
```

Заголовки определяются с помощью тегов от <h1> до <h6>. <h1> определяет заголовок самого большого размера, а <h6> определяет заголовок самого маленького размера.

```
<h1>Это заголовок первого уровня</h1>
<h2>Это заголовок второго уровня</h2>
<h3>Это заголовок третьего уровня</h3>
<h4>Это заголовок четвертого уровня</h4>
<h5>Это заголовок пятого уровня</h5>
<h6>Это заголовок шестого уровня</h6>
```

Тег комментария используется для вставки комментариев в исходный код HTML. Комментарии будут проигнорированы браузером. Комментарии можно использовать для пояснения кода, что может помочь при редактировании исходного кода в будущем.

```
<!-- Это комментарий -->
```

Пример:

```
<html>
<body>
Этот текст будет показан в окне браузера.
<!-- Этот текст не будет показан, это комментарий. -->
</body>
</html>
```

Гипертекстовая ссылка – это адрес другого HTML документа, который тематически, логически или каким-либо другим способом связан с документом, в котором ссылка определена. Для записи гипертекстовых ссылок в системе WWW была разработана форма, которая называется Universe Resource Locator.

В HTML они (гипертекстовые ссылки) делятся на два класса. Бывают контекстные ссылки и общие. Контекстные ссылки как-бы вмонтированы в тело документа, в то же время, когда общие ссылки связаны со всем документом в целом и при просмотре любого фрагмента документа могут быть использованы. Оба класса ссылок существуют в стандарте языка с самого основания, однако, приоритетной популярностью по началу пользовались именно контекстные ссылки. Эта привело к «атрофированию» механизма задействия общих ссылок.

Но по мере стандартизации интерфейса пользователя и стилей представления информации, разработчики языка вновь вернулись к общим ссылкам для приспособления их к задачам управления этим интерфейсом. Структура HTML документа дает возможность использовать вложенные друг в друга элементы. Можно сказать, что сам документ и является одним большим элементом с именем «HTML».

Новая версия языка гипертекстовой разметки, потенциальные возможности которого гораздо превышают предшественника – это HTML 5. Новые теги и новая семантика упрощают работу написания приложения. HTML-страницы становятся все более и более логичными: их структура отделяется от визуального оформления. Такой код легко разбирают и читают люди и, благодаря стандартизации, более точно выполняют устройства, какими бы они ни были. Стандартизация и логичность - это то, к чему стремится язык HTML и что требуется от браузеров и других устройств, работающих с ним.

Глава 3. Примеры использования языка HTML

Создание сайта HTML.

Создадим сайт в блокноте при помощи HTML. HTML-документ должен иметь следующую структуру:

```
<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8">

<title>Как создать сайт с нуля - "Нубекс"</title>

</head>

<body>

<h1>Мой первый сайт в блокноте HTML</h1>

<p>Привет! Это мой первый сайт.</p>

</body>

</html>
```

Скопируйте приведенный код в блокнот (Notepad) и сохраните на компьютере в формате .html. Чтобы теперь посмотреть на нашу веб-страницу, откройте сохраненный файл с помощью браузера (Правой кнопкой мыши -> Открыть с

помощью).

Большинство тегов, которые используются в нашем коде уже были рассмотрены в предыдущих статьях:

html - корневой тег документа, является контейнером веб-страницы;

head - в этом теге указываются основные данные, касающиеся генерации веб-страницы (заголовок, стили, скрипты, мета-теги и так далее);

body - основной тег страницы. Представляет собой "тело" документа, внутри которого помещаются остальные теги;

<!DOCTYPE html> - указывает на тип документа (Doctype);

meta - используется для указания метаданных страницы (в нашем случае тег указывает на кодировку страницы. Подробнее о мета-тегах читайте в статье [Мета-теги](#));

title - задает непосредственный заголовок страницы (Тег title);

H1 - заголовок страницы/статьи (подзаголовки задаются тегами H2-H6);

P - тег, с помощью которого выделяются абзацы в тексте страницы;

Как показано выше, HTML-страницы сайта имеют древовидную структуру. Одни теги могут быть вложены в другие, а те, в свою очередь, - в третьи. Это означает, что структура документа определена достаточно четко. Открывающиеся теги обязательно должны иметь закрывающую пару. Немного усложним нашу страницу:

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<meta charset="utf-8">
```

```
<title>ООО Василий Пупкин</title>
```

```
</head>
```

```
<body>
```

```
<h1>Информация о нашей компании</h1>
<center></center>
<h2>Кто мы?</h2>
<p>Мы - команда профессионалов.</p>
<h2>Наши услуги</h2>
<h3>Создание сайтов</h3>
<p>Мы создаем по-настоящему крутые сайты.</p>
<h3>Продвижение сайтов</h3>
<p>Ваш сайт в ТОП-3 поисковых систем через 2 дня.</p>
<h3>Посадка картошки</h3>
<p>20 соток в час.</p>
<h2>Контакты</h2>
<div id="map">Карта проезда:
...
</div>
<p>Телефон: 0000</p>
</body>
</html>
```

Создание фреймов.

Для создания фрейма используется тег `<frameset>`, который заменяет тег `<body>` в документе и применяется для деления экрана на области. Внутри данного тега находятся теги `<frame>`, которые указывают на HTML-документ, предназначенный для загрузки в область (рис. 1).

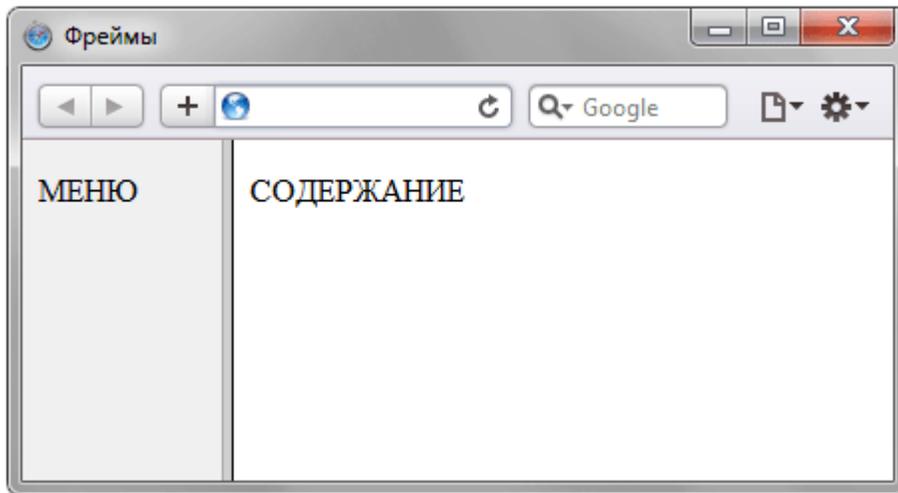


Рисунок 1. Пример разделения окна браузера на два фрейма

При использовании фреймов необходимо как минимум три HTML-файла: первый определяет фреймовую структуру и делит окно браузера на две части, а оставшиеся два документа загружаются в заданные окна. Количество фреймов не обязательно равно двум, может быть и больше, но никак не меньше двух, иначе вообще теряется смысл применения фреймов.

Рассмотрим этапы создания фреймов на основе страницы, продемонстрированной на рис. 1. Нам понадобится три файла: `index.html` — определяет структуру документа, `menu.html` — загружается в левый фрейм и `content.html` — загружается в правый фрейм. Из них только `index.html` отличается по структуре своего кода от других файлов.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Frameset//EN" "http://www.w3.org/TR/html4/frameset.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Фреймы</title>
</head>
<frameset cols="100,*">
<frame src="menu.html" name="MENU">
<frame src="content.html" name="CONTENT">
</frameset>
</html>
```

В случае использования фреймов в первой строке кода пишется следующий тип документа.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Frameset//EN" "http://www.w3.org/TR/html4/frameset.dtd">
```

Данный `<!DOCTYPE>` указывает браузеру, что он имеет дело с фреймами, эта строка кода является обязательной. Контейнер `<head>` содержит типовую информацию вроде кодировки страницы и заголовка документа. Вот только учтите,

что заголовок остается неизменным, пока HTML-файлы открываются внутри фреймов.

В данном примере окно браузера разбивается на две колонки с помощью атрибута `cols`, левая колонка занимает 100 пикселей, а правая — оставшееся пространство, заданное символом звездочки. Ширину или высоту фреймов можно также задавать в процентном отношении, наподобие таблиц.

В теге `<frame>` задается имя HTML-файла, загружаемого в указанную область с помощью атрибута `src`. В левое окно будет загружен файл, названный `menu.html`, а в правое — `content.html`. Каждому фрейму желательно задать его уникальное имя, чтобы документы можно было загружать в указанное окно с помощью атрибута `name`.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
  <title>Навигация по сайту</title>
</head>
<body style="background: #f0f0f0">
  <p>МЕНЮ</p>
</body>
</html>
```

В данном примере серый фон на странице задается с помощью стилей, о которых речь пойдет далее.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
  <title>Содержание сайта</title>
</head>
<body>
  <p>СОДЕРЖАНИЕ</p>
</body>
</html>
```

Рассмотрим более сложный пример уже с тремя фреймами.

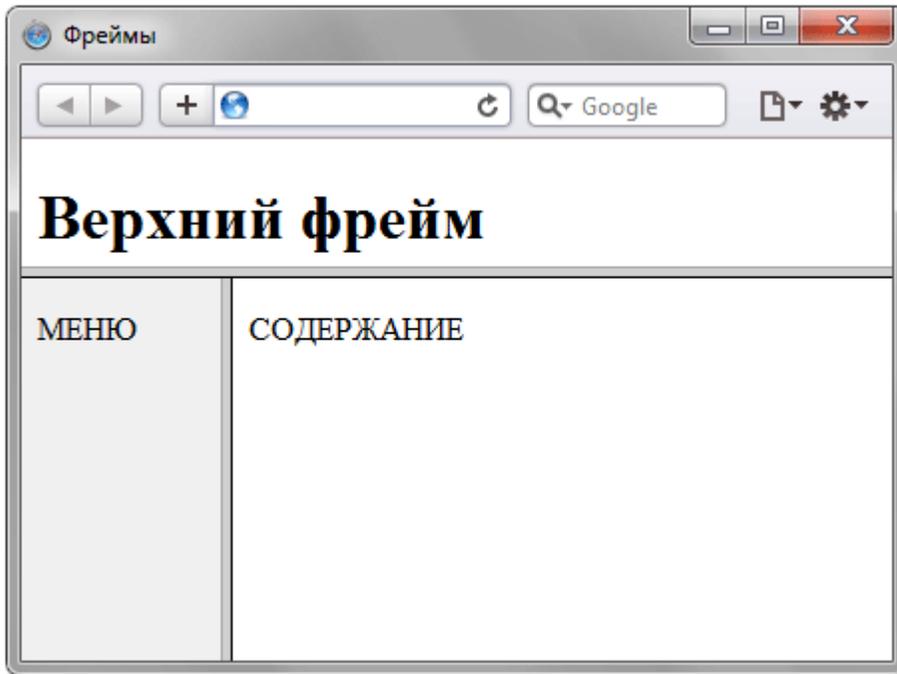


Рисунок 2. Разделение страницы на три фрейма

В данном случае опять используется тег `<frameset>`, но два раза, причем один тег вкладывается в другой. Горизонтальное разбиение создается через атрибут `rows`, где для разнообразия применяется процентная запись.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Frameset//EN" "http://www.w3.org/TR/html4/frameset.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Фреймы</title>
</head>
<frameset rows="25%,75%">
<frame src="top.html" name="TOP" scrolling="no" noresize>
<frameset cols="100,*">
<frame src="menu.html" name="MENU">
<frame src="content.html" name="CONTENT">
</frameset>
</frameset>
</html>
```

Как видно из данного примера, контейнер `<frameset>` с атрибутом `rows` вначале создает два горизонтальных фрейма, но вместо второго фрейма подставляется еще один `<frameset>`, который повторяет уже известную вам структуру из примера. Чтобы не появилась вертикальная полоса прокрутки, и пользователь не мог самостоятельно изменить размер верхнего фрейма, добавлены атрибуты `scrolling="no"` и `noresize`.

Заключение

Язык разметки (текста) — набор символов или последовательностей, вставляемых в текст для передачи информации о его выводе или строении.

Мы рассмотрели 5 языков разметки: HTML, XML, Wiki, BBcod и Textile.

Более подробно рассмотрели самый известный язык гипертекстовой разметки. В 1991 году, британским ученым Тимом Бернерсом Ли, сотрудником Европейского института физики частиц (CERN) в Женеве, была разработана система передачи гипертекстовой информации через интернет. А за основу нового языка был взят SGML. Язык разметки гипертекста был назван - HTML (Hyper Text Markup Language) и он является до сих пор самым известным из приложений SGML. HTML в первую очередь был разработан для обмена научной и технической документации для использования людьми, не являющимися специалистами в области верстки. Последняя версия языка это HTML 5.0.

Новый язык программирования обладает как неопровержимыми преимуществами перед предыдущими версиями, так и грешит некоторыми недостатками. К неоспоримо сильным сторонам HTML 5 принадлежат невероятно большие возможности по созданию качественно новой графики. Особую благодарность в этом следует выразить тегу canvas, который уже стал внедряться во многие браузеры. Благодаря функции canvas веб-разработчики получили расширенные возможности по созданию графики нового поколения.

Основным элементом языка HTML являются теги. Тег - это, как правило, парная конструкция, предназначенная для задания свойств текста или другой информации, расположенной внутри него.

В третьей главе мы рассмотрели пример создания сайта и фреймов в HTML, а также рассмотрели теги которые использовали при написании.

Список литературы

1. HTML 5: нововведения, преимущества и недостатки <http://ua-blogger.com/html-5-novovvedeniya-preimuschestva-i-nedostatki.html> дата обращения (01.05.2019)

HTML 5: эволюция мутаций, или куда доведет язык [Электронный ресурс]/Режим доступа URL: http://sotoman.ru/reviews/ID_987.html дата обращения (01.05.2019)

1. HTML- история развития языка разметки гипертекста – [Электронный ресурс]. URL:
http://www.castcom.ru/publications/web/html_istoriya_razvitiya_yazyka_razmetki_giperteksta
обращения: 01.05.2019)
2. Дакетт, Д. HTML и CSS. Разработка и дизайн веб-сайтов / Д. Дакетт. - М.: Эксмо, 2015. - 480с.

Джон Дакетт «Основы веб-программирования с использованием HTML, XHTML и CSS» - Эксмо, 2016. - 768 с.

Дронов, В. HTML 5, CSS 3 и Web 2.0. Разработка современных Web-сайтов / В. Дронов. - М.: БХВ-Петербург, 2014. - 138 с.

Егорова Ю.Н., Мытникова Е.А., Мытников А.Н., Егорова О.А. Программный комплекс оценки угроз информационной безопасности информационных систем как эффективное средство формирования профессиональных компетентностей бакалавров по дисциплине «Информационная безопасность»//Современные наукоемкие технологии. — 2016. — № 4-1. — С. 109-113.

Краткая история HTML- [Электронный ресурс]. URL:
<http://htmleditors.ru/Rasnoe/history/history1.html>(дата обращения: 01.05.2019)

Краткая история языков HTML и XHTML- [Электронный ресурс]. URL: <http://html5.ru/istoriya-html5> (дата обращения: 01.05.2019)

Помощь по BBCode- [Электронный ресурс]. URL: <http://eko-fazenda.ru/forum/faq.php?mode=bbcode> (дата обращения: 01.05.2019)

Прохоренок, Н. А. HTML, JavaScript, PHP и MySQL. Джентльменский набор Web-мастера / Н.А. Прохоренок, В.А. Дронов. - Москва: СПб. [и др.] : Питер, 2015. - 768 с.

Ташков П, «Веб-мастеринг HTML, CSS, JavaScript, PHP, CMS, AJAX, раскрутка» - Книга по Требованию, 2017. - 512 с.

Что такое HTML и XHTML? История. - [Электронный ресурс]. URL:
<http://monetavinternete.ru/sozdaem-sajt-s-nulya/osnovy-html/chto-takoe-html-xhtml-istoria/> (дата обращения: 01.05.2019)

Язык HTML. История развития – [Электронный ресурс]. URL:
<http://preal.ru/under/yazyk-html-istoriya-razvitiya/>(дата обращения: 01.05.2019)

1. HTML- история развития языка разметки гипертекста – [Электронный ресурс]. URL:
http://www.castcom.ru/publications/web/html_istoriya_razvitiya_yazyka_razmetki_gipertekst
обращения: 01.05.2019)
[↑](#)
2. Краткая история HTML– [Электронный ресурс]. URL:
<http://htmleditors.ru/Rasnoe/history/history1.html>(дата обращения: 01.05.2019) [↑](#)
3. Прохоренок, Н. А. HTML, JavaScript, PHP и MySQL. Джентльменский набор Web-мастера / Н.А. Прохоренок, В.А. Дронов. - Москва: СПб. [и др.] : Питер, 2015. - 768 с. [↑](#)
4. Дронов, В. HTML 5, CSS 3 и Web 2.0. Разработка современных Web-сайтов / В. Дронов. - М.: БХВ-Петербург, 2014. - 138 с. [↑](#)
5. Язык HTML. История развития – [Электронный ресурс]. URL:
<http://preal.ru/under/yazyk-html-istoriya-razvitiya/>(дата обращения: 01.05.2019) [↑](#)
6. Краткая история языков HTML и XHTML– [Электронный ресурс]. URL: <http://html-5.ru/istoriya-html5> (дата обращения: 01.05.2019)
[↑](#)
7. Что такое HTML и XHTML? История. – [Электронный ресурс]. URL:
<http://monetavinternete.ru/sozdaem-sajt-s-nulya/osnovy-html/chto-takoe-html-xhtml-istoria/> (дата обращения: 01.05.2019) [↑](#)
8. Джон Дакетт «Основы веб-программирования с использованием HTML, XHTML и CSS» - Эксмо, 2016. - 768 с. [↑](#)
9. Ташков П, «Веб-мастеринг HTML, CSS, JavaScript, PHP, CMS, AJAX, раскрытка» - Книга по Требованию, 2017. - 512 с. [↑](#)

10. Помощь по BBCode– [Электронный ресурс]. URL: <http://eko-fazenda.ru/forum/faq.php?mode=bbcode> (дата обращения: 01.05.2019) [↑](#)
11. Егорова Ю.Н., Мытникова Е.А., Мытников А.Н., Егорова О.А. Программный комплекс оценки угроз информационной безопасности информационных систем как эффективное средство формирования профессиональных компетентностей бакалавров по дисциплине «Информационная безопасность»//Современные наукоемкие технологии. — 2016. — № 4-1. — С. 109-113. [↑](#)
12. HTML 5: эволюция мутаций, или куда доведет язык [Электронный ресурс]/Режим доступа URL: http://sotoman.ru/reviews/ID_987.html дата обращения (01.05.2019) [↑](#)
13.
HTML 5: нововведения, преимущества и недостатки <http://ua-blogger.com/html-5-novovvedeniya-preimuschestva-i-nedostatki.html> дата обращения (01.05.2019)
[↑](#)
14. Дакетт, Д. HTML и CSS. Разработка и дизайн веб-сайтов / Д. Дакетт. - М.: Эксмо, 2015. - 480 с. [↑](#)