

## **Содержание:**

# **Введение**

Данная работа посвящена языку гипертекстовой разметки HTML (Hypertext Markup Language). Первая версия этого языка была разработана в 1991 году и стала основой для построения всех Web – страниц, которые можно встретить в сети Интернет.

С тех пор HTML не сдал позиций, и по сегодняшний день остается незаменимым средством для создания web – страниц, но уже является не единственным. Дополнением к нему служат такие средства, как:

- Язык сценариев JavaScript;
- Каскадные таблицы стилей (CSS).

Сейчас разработка web – страниц происходит следующим образом:

1. Элементы сайта создаются с помощью HTML – верстки;
2. Стиль этих элементов задается с помощью CSS;
3. «Оживление» элементов осуществляется с помощью JavaScript.

Кроме того, в работе рассматривается расширяемый язык разметки гипертекста XHTML, являющейся модификацией HTML со строгим контролем написания кода.

Целью данной работы является характеристика языка гипертекстовой разметки HTML. Для достижения данной цели ставятся следующие задачи:

1. Обзор истории HTML и XHTML;
2. Анализ возможностей HTML и XHTML;
3. Анализ практического применения HTML и XHTML.

Для написания данной курсовой работы использовалась следующая литература:

1. Сакулин С.А. Основы интернет-технологий: HTML, CSS, Javascript, XML.

В данной книге хорошо раскрыты технология HTML для публикации документов и способы ее применения, теги, элементы, атрибуты, приемы использования таблиц, гиперссылок, списков, способы задания цветов и шрифтов.

1. Чиртик А. HTML: Популярный самоучитель.

В книге хорошо раскрыта история возникновения языка HTML, и простым и кратким языком дано его описание.

1. Муссиано Ч., Кеннеди Б. HTML и XHTML. Подробное руководство.

Самая полная и современная книга по языкам HTML и XHTML, разъясняющая работу и взаимодействие каждого их элемента.

Кроме книг были использованы различные интернет-источники, которые будут указаны в библиографическом списке.

## **1. История HTML**

### **1.1. Создание технологии HTML**

Практически все современные веб-сайты имеют в своей основе HTML. Разработкой данной технологии занималась Европейская организация по ядерным исследованиям (CERN). Причиной разработки данной технологии являлась необходимость быстрого доступа к данным, передачи их от пользователя к пользователю в структурированном виде и с одинаковым отображением информации на экране.[\[1\]](#)

Создать технологию, которая решит данную проблему, решил английский ученый Тим Бернерс-Ли.[\[2\]](#) Основой для разработки послужил язык разметки SGML, который применялся в печатном и издательском деле, но из-за своей сложности не нашел широкого применения. [\[3\]](#)

Принципы, которые лежат в основе HTML:

1. Простота использования.

Создание такого языка, для работы с которым не нужно тратить много времени на обучение, и имеющего понятный синтаксис и небольшое количество структурных элементов.

1. Универсальность.

Документ, с HTML версткой, не должен отображаться по-разному на любом виде устройств, даже если они имеют разные операционные системы, платформы и т.д.

### 1. Расширяемость.

Так как с каждым годом появляются все новые и новые требования, и эти требования невозможно было учесть в процессе разработки технологии, то необходима возможность для модернизации и дополнения. [\[4\]](#)

Главными особенностями HTML является то, что:

1. Можно верстать электронный документ на уровне полиграфического предприятия;
2. В документ можно добавлять различные изображения, аудио- и видеофрагменты;
3. HTML по сути представляет собой обычный текстовый файл. Редактирование можно осуществлять без специального ПО, с помощью текстовых процессоров. [\[5\]](#)

## 1.2. Развитие языка HTML

Появление языка HTML позволило создать внутреннюю сеть CERN. Технологии, которая были разработаны для данной сети, в последствии были модернизированы и стали основой огромного проекта «всемирной паутины».

Данный проект подразумевал использования языка HTML для верстки документов, которые в последствии будут опубликованы в данной сети. Для реализации связей между этими документами используются гиперссылки. [\[6\]](#)

Для контроля за данным проектом, в октябре 1994 года был создан World Wide Web Consortium (сокр. W3C). Т.к. официально версия HTML 1.0 не было, то W3C сразу начал разрабатывать HTML 2.0, показывая этим, что данная версия отличается от предыдущей. Разработка была утверждена в 1995 году. Главной особенностью данной версии стало появление форм, для возможности отправки информации на сервер. [\[7\]](#)

Параллельно с выпуском второй версии, началось обсуждение третьей.

Она была выпущена в марте 1995 года. В третьей версии было предложено много новых возможностей:

- Использование таблиц;
- Возможность обтекания изображения текстом;
- Визуализация сложных математических формул;
- Теги для создания страниц;
- Поддержка каскадных таблиц стилей CSS. Данное нововведение позволило удовлетворить потребность пользователей, которым необходима была настройка визуального представления.

Четвертая спецификация HTML вышла 18 декабря 1997 года. Главной особенностью стало обеспечение корректной работы в разных браузерах и на разных платформах. Данные разработки были усовершенствованы в версии 4.01, которую выпустили 24 декабря 1999 года.[\[8\]](#)

Последней спецификацией на данный момент является HTML 5, вышедшая 28 октября 2014 года. Нововведения ориентированы больше на машины, интерпретирующие код и позволяют лучше индексировать сайт.[\[9\]](#)

### **1.3. Появление языка XHTML**

Кроме разработки языка HTML, W3C также занималось разработкой XHTML. Причиной для этого стало тупик развития HTML. В новых тегах уже нет необходимости, а для разработчиков главным теперь является стиль страницы. [\[10\]](#)

Кроме того, язык HTML начал терять свою популярность по сравнению с расширяемым языком разметки XML, в виду того, что с помощью XML можно создавать свои теги и описывать их структуру.[\[11\]](#)

По мнению W3C XHTML это последняя версия HTML, которая вскоре заменит его.

Первая версия спецификации XHTML появилась в качестве рекомендации 26 января 2000 года. XHTML 1.0 описывала три типа документов:

1. XHTML-1.0-Strict – тип, в котором исключалось использование любых элементов и атрибутов, устанавливающих графическое отображение (например, элемент font или атрибут bgcolor). Это наиболее рекомендуемый тип документов;

2. XHTML-1.0-Transitional – в данном типе допускается использовать элементы и атрибуты, устанавливающие графическое отображение. Данный тип в основном используется для таких операций как, например, облегчения миграции документов с HTML 3.2 на XHTML;
3. XHTML-1.0-Frameset: тип, дополняет возможности XHTML-1.0-Transitional возможностью использования фреймов;[\[12\]](#)

Специально для мобильных устройств были разработаны версии XHTML Basic и XHTML MP:

XHTML Basic – специальная облегчённая версия XHTML для устройств, которые не могут использовать полный набор элементов XHTML;[\[13\]](#)

XHTML MP – основанный на XHTML Basic, добавляет специфические элементы для мобильных телефонов.[\[14\]](#)

Последняя версия XHTML – XHTML 2.0, но эта версия не достигла статуса рекомендации и деятельность группы, занимающейся разработкой XHTML была прекращена в конце 2010 года, а все разработки включены в HTML 5.[\[15\]](#)

Язык гипертекстовой разметки HTML разрабатывался в качестве простого, универсального и расширяемого инструмента для быстрого доступа к данным, с одинаковым отображением структуры на различных устройствах.

Позже данный язык стал основой для «всемирной» паутины и до сих пор используется для структурирования страниц web-сайтов.

Последней версией языка является HTML 5, включающей разработки XHTML 2.0, разрабатываемого параллельно для сокращения разрыва между HTML и XML, и в последствии замены HTML.

## **1.4. Анализ языка HTML**

По своей сути HTML не является языком программирования, хотя многие его функции позволяют сделать простые приложения непосредственно используя код, называемым гипертекстом. [\[16\]](#)

Предназначением языка HTML является разметка информации на странице. Язык HTML позволяет оформить информацию в виде:

- таблиц;
- списков;
- параграфов;
- изображений;
- разбивать документ на части при помощи заголовков;
- формы для работы с пользователями;
- организовывать HTML – документы в виде списков.[\[17\]](#)

Главным предназначением HTML является создание разметки документа или верстки, но также он имеет различные элементы оформления, для этого есть специальные теги, с помощью которых можно сделать текст жирным или курсивным, надстрочным или подстрочным, и т.д. Некоторые HTML теги из-за своей неактуальности считаются запрещенными, а использование некоторых тэгов считается дурным тоном.[\[18\]](#)

Для создания неповторимого стиля HTML документов и для того, чтобы сделать HTML теги неповторимыми были созданы HTML атрибуты, но сейчас уже никто не использует их, так как на их замену были разработаны CSS (каскадные таблицы стилей). Исключением являются некоторые почтовые клиенты, которые не поддерживают CSS и для оформления электронных писем используются HTML атрибуты.

Основой и самой массовой сферой применения HTML является сеть Интернет. Для разработки каждого сайта, представляющего информацию пользователю в графическом виде, использует HTML, но также HTML очень активно используют при разработке мобильных и настольных приложений для различным операционных систем. [\[19\]](#)

Большинство современных программ для работы с HTML-документами, например, такие как Sublime Text или Notepad++, сами создают основу разметки документов, но для того, чтобы понимать принципы работы, правильно читать и создавать ее самостоятельно, необходимо познакомиться с простыми правилами.[\[20\]](#)

Для разметки документа используются теги, в основном парные (открывающий и закрывающий). Теги указываются в угловых скобках, не отображаются пользователю, но при этом обеспечивают строгую разметку страниц.[\[21\]](#)

Браузер, обрабатывающий HTML-документ, визуализирует его в окне, используя последовательность HTML-тегов. В каждом HTML-документе должны присутствовать три главных части:

- 1) Объявление HTML;
- 2) Заголовочная часть;
- 3) Тело документа.[\[22\]](#)

Для объявления HTML кода используются теги `<HTML>` и `</HTML>`. Пара этих тегов сообщает браузеру, что между ними находится документ формата HTML. Тег `<HTML>` является самым первым и должен указываться в самом начале документа, а последним - `</HTML>` (указывается в самом конце). Единственное, что может быть перед этими тегами - это тег документа, в котором указывается стандарт, с использованием которого будет отображаться содержимое (тег DOCTYPE).[\[23\]](#)

Пример:

```
<html>
```

Тут код страницы

```
</html>
```

Заголовочная часть обозначается тегами `<HEAD>` и `</HEAD>`. Эти теги содержат информацию о документе (название, ключевые слова для поиска, описание и т.д.). Кроме того, здесь указывается информация, предназначенная для создания формата документа, используемые скрипты, которые используются для отображения данных, и стили, применяемые к элементам документа.[\[24\]](#)

Однако самым главным из всей этой информации является название документа, которое отображается в верхней строке окна браузера и в списках "Избранное".[\[25\]](#) Специальные программы-спайдеры поисковых систем, занимающиеся индексированием сайтов, используют название документа для построения своих баз данных. Для того чтобы дать название HTML-документу текст помещается между тегами `<TITLE>` и `</TITLE>`. [\[26\]](#) Например, команда:

```
<TITLE>Заголовок страницы</TITLE>
```

будет выглядеть в браузере, как на рисунке 1.

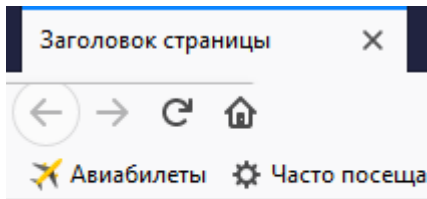


Рис. 1. Заголовок страницы

Третьей важной частью документа является его тело. Оно располагается сразу за заголовком и находится между тегами `<BODY>` и `</BODY>`. Тег `<BODY>` должен находиться сразу после тега `</HEAD>`, а тег `</BODY>` - перед тегом `</HTML>`. Содержимое, указанное между этими тегами будет отображено, кроме элементов, для которых указаны специальные значения, и самих тегов для форматирования документа.[\[27\]](#)

Тело HTML-документа - это область, где автор размещает данные, структурированные средствами HTML.[\[28\]](#) В разделе BODY все символы табуляции и переноса строк браузером игнорируются и никак не влияют на отображение страницы. Поэтому перевод строки в исходном тексте HTML-документа не приведет к началу новой строки в отображаемом обозревателем тексте при отсутствии специальных тегов. Об этом очень важно не забывать и ставить разделяющие строки теги, иначе у текста не будет абзацев, и он станет нечитаемым.[\[29\]](#)

Структуру HTML документа можно представить следующим образом:

```
<!DOCTYPE HTML *****>
```

```
<html>
```

```
<head>
```

```
<title> Заголовок страницы (отражается в окне!) </title>
```

Заголовок документа. Не отображается на экране

```
</head>
```

```
<body>
```

Тело документа. Отображается на экране



</body>

</html>[\[30\]](#)

Для управления плотностью шрифта применяются теги:

<b>текст</b>

<strong>текст</strong>

Отличие тэга <b> от тэга <strong> в том, что тэг <b> указывает браузеру выводить текст жирным шрифтом, а тэг <strong> указывает, что текст надо выделить. Как браузер будет выделять текст, жирным шрифтом, курсивом или жирным курсивом - зависит от конкретного браузера.[\[31\]](#)

Курсивный шрифт выводится с помощью тэгов:

<i>текст</i>

<em>текст</em>

Для подчеркнутого шрифта используются теги:

<u>текст</u>

Создание надстрочного индекса осуществляется с помощью тегов:

<sup>текст</sup>

Подстрочный индекс выводится с помощью тэга:

<sub>текст</sub>[\[32\]](#)

Для любителей ретро стиля html предлагает имитацию стиля печатной машинки (Teletype). Для этого необходимо воспользоваться тегам:

<tt>текст</tt>[\[33\]](#)

Кроме стиля шрифта можно управлять и его размером.

Первым тегом в данном разделе является тег для создания базового шрифта. Размер, цвет и стиль шрифта по умолчанию задается с помощью тега <BASEFONT> и не применяется к заголовкам. Если базовый шрифт не задан по умолчанию используется шрифт размером 3.[\[34\]](#)

Атрибуты элемента <BASEFONT>:

- color (цвет) – цвет шрифта;
- size (целое число от 1 до 7) – размер шрифт;
- face – список разделенных запятыми названий шрифтов.

Например, для установки размер базового шрифта равным двум, необходимо использовать атрибут SIZE = "2". [\[35\]](#) В результате тег <BASEFONT> примет вид:

```
<BASEFONT SIZE="2">
```

Также размер шрифта можно изменить с помощью тега <FONT> и его атрибута SIZE. Атрибут SIZE может принимать значения от 1 до 7. [\[36\]](#)

Пример:

```
<font size="7"> size=7</font>
```

В результате шрифт в окне браузера будет выглядеть, как на рисунке 1:

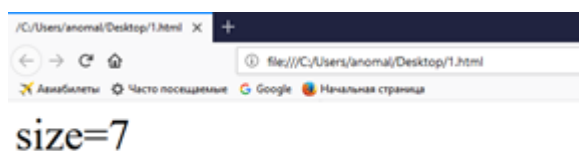


Рис. 1. Изменение размера шрифта

В атрибуте SIZE можно указывать размер шрифта относительно текущего размера SIZE. [\[37\]](#) Например:

```
<font size="+1">size +1</font>
```

```
<font size="+2">size +2</font>
```

```
<font size="-3">size -3</font>
```

Имя шрифта задается с помощью атрибута FACE тега <FONT> [\[38\]](#):

```
<font face="имя шрифта"></font>< /FONT>
```

Можно задавать несколько шрифтов через запятую, в этом случае используется первый найденный шрифт. [\[39\]](#)

Пример:

```
<font face="Impact", "Arial Cyr", "Arial", "MS Sans Serif">
```

текст будет выведен шрифтом "Impact" при наличии его на вашем компьютере.

```
</font>
```

Результат выполнения (рисунок 2):



**текст будет выведен шрифтом "Impact" при наличии его на вашем компьютере.**

Рис. 2. Установка имени шрифта

## Заголовки

Существует 6 уровней заголовков. Для их вывода используют теги:

```
<h1>Самый большой заголовок</h1>
```

...

```
<h6>Самый маленький заголовок</h6> \[40\]
```

Выглядит следующим образом (рисунок 3):

**H1**

**H2**

**H3**

H4

H5

H6

Рис. 3. Заголовки

## Работа с текстом

Поскольку в процессе работы в теле html-документа все символы табуляции и переноса текста игнорируются, то были созданы теги для выполнения этих

операций.[\[41\]](#)

Принудительный перенос строки выполняется с помощью тэга `<br>`. [\[42\]](#)

Пример работы с тегом:

Выполняем`<br>`переход на`<BR>`следующую строку

Результат выполнения показан на рисунке 4:

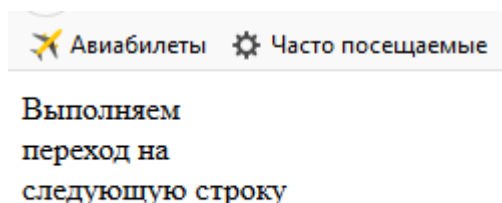


Рис. 4. Перенос строки с помощью тэга `<br>`

Для того чтобы запретить перенос строки, предусмотрены теги:

`<nobr>текст</nobr>`[\[43\]](#)

Данный тег указывает браузеру, что вывод текста между ними должен выполняться одной строкой. Если строка не помещается в окно браузера, появляется горизонтальная линейка прокрутки.[\[44\]](#)

Выравнивание блока текста осуществляется с помощью атрибута ALIGN тэга:

`<DIV>текст</DIV>`

Атрибут ALIGN могут иметь значения:

- ALIGN="LEFT" - выравнивание по левому краю;
- ALIGN="RIGHT"- выравнивание по правому краю;
- ALIGN="CENTER"- выравнивание по центру;
- ALIGN="JUSTIFY" - выравнивание по обоим краям.[\[45\]](#)

Пример:

`<DIV ALIGN="CENTER">Текст, выравненный по центру</DIV>`

Вообще, атрибут ALIGN можно применять во многих тэгах, например:

- `<P ALIGN="JUSTIFY">текст</P>` - выравнивание абзаца
- `<TD ALIGN="CENTER">текст</TD>` - выравнивание текста в ячейке таблицы
- `<H1 ALIGN="CENTER">текст</H1>` - выравнивание заголовка и т.д.[\[46\]](#)

Отцентрировать блок текста можно также и с помощью тега:

`<CENTER>текст</CENTER>`.

Для создания блока текста с отступом используют тег:

`<BLOCKQUOTE>текст</BLOCKQUOTE>`[\[47\]](#)

Как правило, блоки текста разделяют между собой абзацами (параграфами). По умолчанию между параграфами существует небольшой вертикальный отступ, называемый отбивкой. Синтаксис создания абзацев следующий:

`<p>Абзац 1</p>`

`<p>Абзац 2</p>`

Каждый абзац начинается с тега `<p>` и заканчивается необязательным закрывающим тегом `</p>`.[\[48\]](#)

## Управление цветом

Для задания цветов на веб-странице применяется три способа:

1. использует обозначения цветов в шестнадцатеричном коде;
2. использует обозначения цветов в десятичном коде;
3. по названию некоторых цветов.

Преимущественно используется способ, основанный на шестнадцатеричной системе исчисления, как наиболее универсальный. Десятичная система, хотя и более привычна для представления, находит применение только с помощью CSS.  
[\[49\]](#)

Цвет шрифта меняется с помощью атрибута `COLOR` тега `<FONT>`.

`<FONT COLOR="#FF0000">FONT COLOR="#FF0000"</FONT>`

`<FONT COLOR="red">FONT COLOR="red"</FONT>`[\[50\]](#)

Результат показан на рисунке 3:

FONT COLOR="#FF0000" FONT COLOR="red"

Рис. 3. Изменение цвета шрифта

Кроме цвета шрифта также можно изменить цвет фона. Для этого необходимо добавить атрибут bgcolor к тегу body. Тег будет выглядеть следующим образом:

```
<body bgcolor="#333">\[51\]
```

Результат будет следующим:

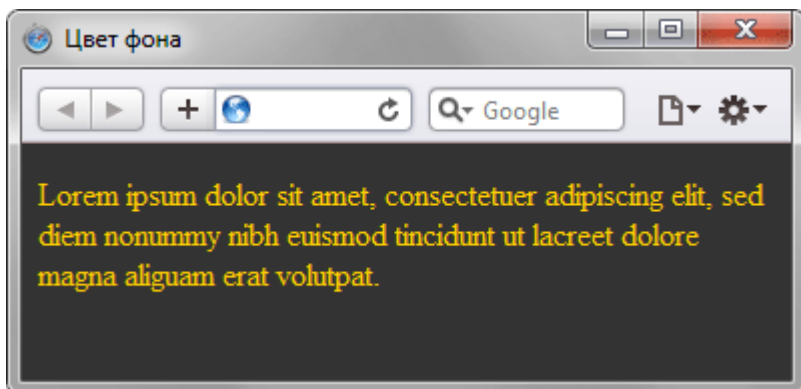


Рис. 3. Изменение цвета фона

### Вставка изображений

Из-за того, что графические данные и текст html невозможно объединить в одном файле, для их отображения на сайте применяется другой подход, нежели с другими элементами html-страниц.[\[52\]](#) Прежде всего, графические изображения, да и другие мультимедийные данные хранятся в отдельных файлах. А для их внедрения в web-страницу используют специальные теги, которые содержат ссылки на эти отдельные файлы. В частности таким тегом является тег <img>. Встретив такой тег с адресом, браузер сначала запрашивает у Web-сервера соответствующий файл с изображением, аудио- или видеороликом, а только затем отображает его на Web-странице.[\[53\]](#)

Все графические изображения и вообще любые данные, которые хранятся в отдельных от web-страницы файлах называются внедренными элементами страницы.[\[54\]](#)

При создании сайтов обычно используют изображения в формате JPEG или GIF, но иногда используют PNG. Главное разобраться в каких случаях какой формат лучше использовать. Если кратко, то:

- JPEG лучше всего использовать для хранения фотографий или полутонных картинок, не содержащих текста;
- GIF используется в основном для анимации;
- PNG — формат для всего остального (иконки, кнопки и др.).[\[55\]](#)

Вставить изображение позволяет одинарный тег `<img>`. Браузер помещает изображение в том месте веб-страницы, где встретит тег `<img>`.

Код вставки картинки в html страницу имеет такой вид:

```

```

Данный html-код поместит на веб-страницу изображение, хранящееся в файле `image.jpg`, который находится в одной и той же папке с веб-страницей. Адрес картинки указывается в атрибуте `src`.[\[56\]](#)

В атрибуте `src` можно указывать не только относительные ссылки на изображения. Так как изображения хранятся в сети вместе с html-страницами, поэтому у каждого файла с изображением есть свой собственный url-адрес. Поэтому в атрибут `src` можно вставлять url-адрес изображения. Например:

```

```

Данный код вставит на страницу изображение с сайта `mysite.ru`. URL-адрес обычно применяется, если вы указываете на изображение, находящееся на другом сайте. Для изображений, хранящихся на вашем сайте лучше использовать относительные ссылки.[\[57\]](#)

## **Создание таблиц**

Чтобы создать простую таблицу HTML достаточно 3 тега: `<table>`, `<tr>` и `<td>`.

Тег `<table>` является корневым контейнером таблицы. Все содержимое таблицы должно находиться внутри него.[\[58\]](#)

Далее необходимо определить строки и ячейки - структуру таблицы.

В HTML таблицах строка (ряд) `<tr>` является контейнером для ячеек. Колонки таблицы определяются позицией ячеек: первая ячейка `<td>` внутри строки `<tr>` будет в первой колонке, второй элемент `<td>` - во второй колонке и так далее.[\[59\]](#)

Для разделения таблицы на колонтитулы (об этом ниже) и основную часть, как обертку строк `<tr>` основной части таблицы используют тег `<tbody>`. Его использование не обязательно в простых таблицах, однако некоторые браузеры и HTML редакторы добавляют его автоматически, поэтому в примерах ниже мы также будем его использовать. Если ваша таблица не имеет колонтитулов, вы можете не использовать тег `<tbody>`.

Например, обработка нижеприведенного кода будет выглядеть, как показано на рисунке 4.[\[60\]](#)

Код:

```
<table>

<tbody>

<tr>

<td>Ячейка 1</td>

<td>Ячейка 2</td>

<td>Ячейка 3</td>

</tr>

<tr>

<td>Ячейка 4</td>

<td>Ячейка 5</td>

<td>Ячейка 6</td>

</tr>

<tr>

<td>Ячейка 7</td>
```



```
<td>Ячейка 8</td>
<td>Ячейка 9</td>
</tr>
</tbody>
</table>
```

Ячейка 1	Ячейка 2	Ячейка 3
Ячейка 4	Ячейка 5	Ячейка 6
Ячейка 7	Ячейка 8	Ячейка 9

Рис. 4. Таблица, построенная по приведенному коду

HTML нельзя назвать языком программирования, но многие функции позволяют сделать простые приложения непосредственно в коде, называемом гипертекстом. Основной задачей языка HTML является структурирование информации на странице. Язык HTML позволяет структурировать информацию различными способами: создавать таблицы, различные виды списков, параграфы, добавлять в документ изображения, разбивать документ на части при помощи заголовков, создавать формы для взаимодействия с пользователями, связывать разные HTML документы списками. В данном разделе указаны эти способы.

## 2. Анализ языка XHTML

XHTML состоит в семействе языков XML, которые расширяют или продолжают версии HTML. При использовании таких языков нельзя пропускать любые теги или атрибуты. При работе с XHTML важно, чтобы каждый открывающийся тег имел закрывающийся в правильном порядке. Например, если в языке HTML допускается использование одиночного тега `<br>`, то в XHTML так делать уже нельзя и нужно написать тег `<br/>`. Это одно из главных различий языков HTML и XHTML. [\[61\]](#)

Документ XHTML должен иметь только один корневой элемент. Элементы документа, в том числе и переменные, указываются в нижнем регистре, а их значения — указываются в кавычки, закрыты и вложены. В XHTML это является обязательным требованием. Объявление DOCTYPE XHTML определяет правила для

документов, которым необходимо следовать.[\[62\]](#)

HTML и XHTML могут быть задокументированы совместно. И HTML 4.01, и XHTML 1.0 имеют три подспецификации:

- Строгую;
- Нестрогую;
- Фрэймовую. [\[63\]](#)

Отличие файлов HTML и XHTML заключается в декларировании документов. В XHTML есть встроенный язык определения функционала атрибутов и все требования, предъявляемые к синтаксису XML, относятся и к XHTML-документу.

В XHTML используется большое количество разработок, предназначенных для XML. К примеру, XSLT – часть технологии XSL, устанавливающая язык изменений XML-документов. XSLT является рекомендацией W3C.[\[64\]](#)

Также есть возможность применения технологии XPath – языка запросов к различным составляющим XML-документа. Он был придуман для организации доступа к частям документа XML в файлах трансформации XSLT и является стандартом консорциума W3C.

Анализ документа XHTML менее сложен и более быстр, чем HTML. Так как синтаксис XML более строг, по сравнению с SGML, то обработка XHTML возможна даже на малоресурсных мобильных телефонах.[\[65\]](#)

## 2.1. Отличия XHTML от HTML

Главным различием между HTML и XHTML является то, что в XHTML используется синтаксис XML, который разрабатывался для помощи в разработке синтаксически верных документов XML. XHTML являет собой словарь XML, в то время как HTML — это лишь предшествующий XHTML язык разметки.[\[66\]](#)

В связи с тем, что XHTML основан на XML, некоторые действия, которые совершенно правильны в HTML4 на базе SGML, совершенно неверны в XHTML. Опишем правила, которые необходимо соблюдать при переходе от HTML к XHTML [\[67\]](#):

1. Документы должны быть корректно созданы.

Это значит, что каждый открывающийся элемент должен быть закрыт (например, `<r></r>`) или быть указанным в специальной форме (например, `<br />`), а также должна контролироваться правильная вложенность элементов.

Хоть некорректная вложенность элементов не допускалось и в SGML, некоторые браузеры нейтрально к этому относились и все равно показывали то, что хотел разработчик.

Правильное использование вложенных элементов:

```
<r>выделяем это <em>слово</em></r>
```

Неправильное использование вложенных элементов (элементы перекрываются):

```
<r>выделяем это <em>слово.</r></em> [68]
```

2. Имена элементов и атрибутов указываются в нижнем регистре.

В документах XHTML должен быть использовать нижний регистр для всех элементов и атрибутов HTML. Это необходимо, поскольку XML чувствителен к регистру, например, `<ul>` и `<UL>` это различные теги. [69]

3. Правила написания значений атрибутов.

Все значения атрибутов, даже цифры должны быть указаны в двойные (" ") или одинарные(' ') кавычки.

Правильно будет указано значение атрибута следующим способом:

```
<td colspan="8">
```

Такое указание будет неверным, так как значение указано не в кавычках:

```
<td colspan=8> [70]
```

Если значение атрибута содержит амперсанд ("&"), то для его отображения должна выражаться мнемоникой ("&amp;"). Например, если атрибут `href` элемента `<a>` ссылается на скрипт CGI, который принимает параметры, то атрибут должен выглядеть так:

```
http://my.site.dom/cgi-bin/myscript.pl?class=guest&name=user
```

Такое использование вызовет ошибку:

<http://my.site.dom/cgi-bin/myscript.pl?class=guest&name=user>[\[71\]](#)

Предопределенные значения атрибутов (например, атрибут `type` элемента `input`) очень зависимы от регистра и должны быть написаны в нижнем регистре. Однако, это не относится к значениям атрибутов, определяемых разработчиками документов (например, значения для атрибутов `id`, `name` или `class`), которые также зависят от регистра, но позволяют использовать символы в верхнем регистре.[\[72\]](#)

Следует принять во внимание, что пробелы в значениях атрибутов удаляются браузерами, а один или несколько последовательных пробелов между словами, а также переводы строк, преобразуются в одиночный пробел. XML не поддерживает минимизацию атрибутов. Пары атрибут/значение обязательно должны быть указаны полностью. Имена атрибутов, такие как `compact` и `checked`, не могут использоваться в элементах без определенных значений. Для того, чтобы код атрибута сработал, необходимо написать его полностью:

```
<dl compact="compact">
```

Но если же атрибут будет иметь, например, такой вид, то ничего не сработает:

```
<dl compact>\[73\]
```

## 6. Элементы `<script>` и `<style>`.

В XHTML элементы `<script>` и `<style>` используются как имеющие содержимое `#PCDATA`. Это резервное имя, описывающее базовые элементы и представляющее тип данных, содержащихся в элементе. Оно означает наличие символьной информации, которая может подвергаться грамматическому анализу. В следствии этого, `<` и `&` будут рассматриваться как начало разметки, а мнемоники, такие как `&lt;` и `&amp;`, будут распознаны процессором XML мнемониками символов `<` и `&` соответственно. [\[74\]](#)

Вообще символ-мнемоника взята из SGML. Данная конструкция ссылается на символ из набора символов документа. В HTML предопределено большое количество спецсимволов. Чтобы вставить определенный символ в разметку, нужно вставить определенную мнемонику в HTML структуру.[\[75\]](#)

Для того, чтобы избежать замены мнемоник символами, необходимо поместить содержимое элементов `<script>` или `<style>` в раздел `CDATA`. Данный раздел говорит обработчику кода о том, что здесь содержатся только символьные данные.

Это необходимо для того, чтобы спрятать от обработчика кода XML код, содержащийся в `script` или `style`, так как для него это обычный namespace.[\[76\]](#)

Namespace – это пространство имен, которое используется для логического объединения классов, функций и переменных. Иначе говоря, namespace дает возможность разделить глобальное пространство имен на подпространства.

Предназначение пространства имен в том, чтобы избежать совпадение имен в глобальном пространстве имен.[\[77\]](#)

Но в `script` или `style` нельзя заменить спец. символы на мнемоники, из-за этого его обработка отменяется с помощью CDATA, и, чтобы javascript не воспринимал CDATA, закомментировать данный раздел.[\[78\]](#)

Комментирование одной строки выполняется с помощью символа `“//”`. В коде выглядит следующим образом:

```
<script type="text/javascript">  
//<![CDATA[  
  
... неэкранированное содержание сценария ...  
  
//]]>  
  
</script>
```

Многострочное комментирование происходит следующим способом:

```
<script type="text/javascript">  
  
/*<![CDATA[*/  
  
... неэкранированное содержание сценария ...  
  
/*]]>*/  
  
</script>
```

Разделы CDATA являются частью объектной модели документа. [\[79\]](#)

7. Исключения SGML.

При работе с SGML разработчики определений типа документа могли исключать элементы как содержимое элемента. Такие исключения отсутствуют в XHTML.[\[80\]](#)

Например, строгое определение типа документа HTML4(Strict DTD) исключает вложенность элемента <a> в другой элемент <a>. Эти исключения не обязательно должны указываться в определении типа документа.[\[81\]](#)

В XHTML:

- В элемент <a> не могут вкладываться другие элементы <a>;
- В элемент <pre> не могут вкладываться элементы <img>, <object>, <big>, <small>, <sub> или <sup>;
- В элемент <button> не могут вкладываться элементы <input>, <select>, <textarea>, <label>, <button>, <form>, <fieldset>, <iframe> или <isindex>;
- В элемент <label> не могут вкладываться другие элементы <label>;
- В элемент <form> не могут вкладываться другие элементы <form>.[\[82\]](#)

## 8. Элементы с атрибутами "id" и "name"

HTML 4 установил атрибут name для элементов a, applet, form, frame, iframe, img и mar, а также внедрил атрибут id. Каждый из этих атрибутов (name и id) создан для использования в качестве идентификаторов фрагмента. В XML, тип ID имеют идентификаторы фрагмента, и элемент может содержать только один атрибут типа ID. Из этого следует, что в XHTML 1.0 атрибут id определён как атрибут типа ID. Файлы XHTML 1.0 считаются правильно структурированными документами XML, если в них используется атрибут id для определения идентификаторов фрагмента.[\[83\]](#)

Значение атрибута id в пределах документа должно быть неповторяющимся.

В XHTML 1.1 для элементов a и mar атрибут name удален, вместо него следует использовать атрибут id.[\[84\]](#)

## 2.2. Валидация XHTML документов

Валидным XHTML-документом является документ, удовлетворяющий всем спецификациям XML. Все браузеры должны придерживаться веб-стандартов и, в соответствии с ними, документы, которые удовлетворяют техническим спецификациям, должны одинаково визуализироваться в любых браузерах.

Использование валидных XHTML-документов рекомендуется W3C, хоть это и не значит, что документ будет корректно отображаться во всех браузерах. Для проверки валидности документа можно воспользоваться службой валидации от W3C. Валидация обнаружит и разъяснит ошибки в XHTML-разметке.[\[85\]](#)

Для того, чтобы документ считался валидным, он должен содержать определение типа документа (DTD). DTD указывается в самом начале документа, перед всеми элементами. [\[86\]](#)

Самыми популярными ошибками в XHTML-разметке являются:

- Использование элементов без закрывающих тегов;
- Не указаны для изображений альтернативные текстов. Для установки альтернативного текста используется атрибут alt. Это необходимо для тех случаев, когда изображение не может быть загружено, и вместо него указывается текст, описывающий изображение;
- Указание текста в теге <body> документа. Текст должен находиться в блочном элементе; [\[87\]](#)
- Вложение внутрискрочных элементов внутрь блочных. Например, такие блочные элементы как <p> или <div> не могут быть вложены внутрь элементов <a>, <span>, <em>;
- Незаключенные в кавычки значения атрибутов;
- Некорректное вложение элементов;[\[88\]](#)
- Некорректное использование ссылок-мнемоник;
- Использование тегов и/или атрибутов, указанных прописными буквами.[\[89\]](#)

Вот образец минимального документа XHTML:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="ru" lang="ru">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=windows-1251" />

<title>Образец минимального документа XHTML</title>

</head>
```

```
<body><p>Образец минимального документа XHTML</p></body>
```

```
</html>\[90\]
```

XHTML представляет собой семейство языков XML, которые расширяют или продолжают версии HTML. Имеет очень строгий синтаксис и не прощает ошибок, в отличие от HTML. Данный язык гипертекстовой разметки более сложен, но зато анализировать код быстрее и проще, и его обработка возможна даже на малоресурсных мобильных телефонах.

## Заключение

В данной курсовой работе были рассмотрены языки гипертекстовой разметки HTML и XHTML.

Язык гипертекстовой разметки HTML разрабатывался в качестве простого, универсального и расширяемого инструмента для быстрого доступа к данным, с одинаковым отображением структуры на различных устройствах.

Позже данный язык стал основой для «всемирной» паутины и до сих пор используется для структурирования страниц web-сайтов.

XHTML разрабатывался для замены HTML и считается его более строгой версией. По сути код XHTML, является неким идеальным кодом веб-страницы, которая не будет отображена, пока все ошибки не будут исправлены. Благодаря этому сайты становятся кросс-платформенными и кросс-браузерными. Иначе говоря, сайты будут верно отображаться как на ПК, так и на мобильных гаджетах.

## Библиографический список

### Книги

1. Сакулин, С.А. Основы интернет-технологий: HTML, CSS, JavaScript, XML: учебное пособие/ С.А. Сакулин. — Москва: МГТУ им. Н.Э. Баумана, 2017. — 112 с.
2. Муссиано Ч. HTML и XHTML. Подробное руководство, 6 - е издание/Ч. Муссиано, Б. Кеннеди - СПб: Символ-Плюс, 2016. - 752 с.



3. Чиртик А. HTML: Популярный самоучитель/ А. Чиртик. -- СПб: Питер, 2014. -320 с.

## Интернет-источники

1. Что такое HTML? URL: <https://zametkinapolyah.ru/verstka-sajtov/chto-takoe-html-dlya-chego-nuzhen-html-kak-sozdat-html-stranicu-vozmozhnosti-yazyka-html.html> (дата обращение: 21.07.2018)
  2. HTML5. URL: <http://htmlbook.ru/html5> (дата обращение: 21.07.2018)
  3. Как изменить цвет фона и текста веб - страницы? URL: <http://htmlbook.ru/faq/kak-izmenit-tsvet-fona-i-teksta-veb-stranitsy> (дата обращения: 21.07.2018)
  4. HTML - история развития языка разметки гипертекста. URL: [http://www.castcom.ru/publications/web/html\\_istoriya\\_razvitiya\\_yazyka\\_razmetki\\_giperteks](http://www.castcom.ru/publications/web/html_istoriya_razvitiya_yazyka_razmetki_giperteks) (дата обращения: 21.07.2018)
  5. Основы XHTML. Основное различие между HTML и XHTML. URL: <http://megaweb.su/osnovi/xhtml/12-xhtml-osnovy-osnovnoe-razlichie-mezhdu-html-i-xhtml.html> (дата обращения: 21.07.2018)
  6. Сравнение HTML и XHTML. URL: <http://www.internet-technologies.ru/articles/sravnenie-html-i-xhtml.html> (дата обращения: 21.07.2018)
  7. XHTML. URL: <http://htmlbook.ru/xhtml> (дата обращения: 21.07.2018)
- 
1. Что такое HTML? URL: <https://zametkinapolyah.ru/verstka-sajtov/chto-takoe-html-dlya-chego-nuzhen-html-kak-sozdat-html-stranicu-vozmozhnosti-yazyka-html.html> (дата обращение: 21.07.2018) [↑](#)
  2. Чиртик А. HTML: Популярный самоучитель/ А. Чиртик. -- СПб: Питер, 2014. -15 с. [↑](#)
  3. HTML - история развития языка разметки гипертекста. URL: [http://www.castcom.ru/publications/web/html\\_istoriya\\_razvitiya\\_yazyka\\_razmetki\\_giperteks](http://www.castcom.ru/publications/web/html_istoriya_razvitiya_yazyka_razmetki_giperteks) (дата обращения: 21.07.2018) [↑](#)
  4. Что такое HTML? URL: <https://zametkinapolyah.ru/verstka-sajtov/chto-takoe-html-dlya-chego-nuzhen-html-kak-sozdat-html-stranicu-vozmozhnosti-yazyka-html.html> (дата обращение: 21.07.2018) [↑](#)

5. Сакулин, С.А. Основы интернет-технологий: HTML, CSS, JavaScript, XML: учебное пособие/ С.А. Сакулин. — Москва: МГТУ им. Н.Э. Баумана, 2017. — 10 с. [↑](#)
6. HTML – история развития языка разметки гипертекста. URL: [http://www.castcom.ru/publications/web/html\\_istoriya\\_razvitiya\\_yazyka\\_razmetki\\_giperteks](http://www.castcom.ru/publications/web/html_istoriya_razvitiya_yazyka_razmetki_giperteks) (дата обращения: 21.07.2018) [↑](#)
7. HTML – история развития языка разметки гипертекста. URL: [http://www.castcom.ru/publications/web/html\\_istoriya\\_razvitiya\\_yazyka\\_razmetki\\_giperteks](http://www.castcom.ru/publications/web/html_istoriya_razvitiya_yazyka_razmetki_giperteks) (дата обращения: 21.07.2018) [↑](#)
8. Что такое HTML? URL: <https://zametkinapolyah.ru/verstka-sajtov/chto-takoe-html-dlya-chego-nuzhen-html-kak-sozdat-html-stranicu-vozmozhnosti-yazyka-html.html> (дата обращение: 21.07.2018) [↑](#)
9. HTML5. URL: <http://htmlbook.ru/html5> (дата обращение: 21.07.2018) [↑](#)
10. Муссиано Ч. HTML и XHTML. Подробное руководство, 6 – е издание/Ч. Муссиано, Б. Кеннеди – СПб: Символ-Плюс, 2016. – 25 с. [↑](#)
11. HTML – история развития языка разметки гипертекста. URL: [http://www.castcom.ru/publications/web/html\\_istoriya\\_razvitiya\\_yazyka\\_razmetki\\_giperteks](http://www.castcom.ru/publications/web/html_istoriya_razvitiya_yazyka_razmetki_giperteks) (дата обращения: 21.07.2018) [↑](#)
12. XHTML. URL: <http://htmlbook.ru/xhtml> (дата обращения: 21.07.2018) [↑](#)
13. Основы XHTML. Основное различие между HTML и XHTML. URL: <http://megaweb.su/osnovi/xhtml/12-xhtml-osnovy-osnovnoe-razlichie-mezhdu-html-i-xhtml.html> (дата обращения: 21.07.2018) [↑](#)
14. Муссиано Ч. HTML и XHTML. Подробное руководство, 6 – е издание/Ч. Муссиано, Б. Кеннеди – СПб: Символ-Плюс, 2016. – 26 с. [↑](#)
15. HTML5. URL: <http://htmlbook.ru/html5> (дата обращение: 21.07.2018) [↑](#)

16. Что такое HTML? URL: <https://zametkinapolyah.ru/verstka-sajtov/chto-takoe-html-dlya-chego-nuzhen-html-kak-sozdat-html-stranicu-vozmozhnosti-yazyka-html.html> (дата обращение: 21.07.2018) [↑](#)
17. Чиртик А. HTML: Популярный самоучитель/ А. Чиртик. -- СПб: Питер, 2014. -31 с. [↑](#)
18. Сакулин, С.А. Основы интернет-технологий: HTML, CSS, JavaScript, XML: учебное пособие/ С.А. Сакулин. — Москва: МГТУ им. Н.Э. Баумана, 2017. — 16 с. [↑](#)
19. Сакулин, С.А. Основы интернет-технологий: HTML, CSS, JavaScript, XML: учебное пособие/ С.А. Сакулин. — Москва: МГТУ им. Н.Э. Баумана, 2017. — 22 с. [↑](#)
20. Чиртик А. HTML: Популярный самоучитель/ А. Чиртик. -- СПб: Питер, 2014. -38 с. [↑](#)
21. Что такое HTML? URL: <https://zametkinapolyah.ru/verstka-sajtov/chto-takoe-html-dlya-chego-nuzhen-html-kak-sozdat-html-stranicu-vozmozhnosti-yazyka-html.html> (дата обращение: 21.07.2018) [↑](#)
22. Что такое HTML? URL: <https://zametkinapolyah.ru/verstka-sajtov/chto-takoe-html-dlya-chego-nuzhen-html-kak-sozdat-html-stranicu-vozmozhnosti-yazyka-html.html> (дата обращение: 21.07.2018) [↑](#)
23. Чиртик А. HTML: Популярный самоучитель/ А. Чиртик. -- СПб: Питер, 2014. -40 с. [↑](#)
24. Сакулин, С.А. Основы интернет-технологий: HTML, CSS, JavaScript, XML: учебное пособие/ С.А. Сакулин. — Москва: МГТУ им. Н.Э. Баумана, 2017. — 112 с. [↑](#)
25. Сакулин, С.А. Основы интернет-технологий: HTML, CSS, JavaScript, XML: учебное пособие/ С.А. Сакулин. — Москва: МГТУ им. Н.Э. Баумана, 2017. — 41 с. [↑](#)

26. Чиртик А. HTML: Популярный самоучитель/ А. Чиртик. -- СПб: Питер, 2014. -41 с. [↑](#)
27. Что такое HTML? URL: <https://zametkinapolyah.ru/verstka-sajtov/chto-takoe-html-dlya-chego-nuzhen-html-kak-sozdat-html-stranicu-vozmozhnosti-yazyka-html.html> (дата обращение: 21.07.2018) [↑](#)
28. Что такое HTML? URL: <https://zametkinapolyah.ru/verstka-sajtov/chto-takoe-html-dlya-chego-nuzhen-html-kak-sozdat-html-stranicu-vozmozhnosti-yazyka-html.html> (дата обращение: 21.07.2018) [↑](#)
29. Муссиано Ч. HTML и XHTML. Подробное руководство, 6 - е издание/Ч. Муссиано, Б. Кеннеди - СПб: Символ-Плюс, 2016. - 35 с. [↑](#)
30. Чиртик А. HTML: Популярный самоучитель/ А. Чиртик. -- СПб: Питер, 2014. -43 с. [↑](#)
31. Чиртик А. HTML: Популярный самоучитель/ А. Чиртик. -- СПб: Питер, 2014. -51 с. [↑](#)
32. Сакулин, С.А. Основы интернет-технологий: HTML, CSS, JavaScript, XML: учебное пособие/ С.А. Сакулин. — Москва: МГТУ им. Н.Э. Баумана, 2017. — 49 с. [↑](#)
33. Что такое HTML? URL: <https://zametkinapolyah.ru/verstka-sajtov/chto-takoe-html-dlya-chego-nuzhen-html-kak-sozdat-html-stranicu-vozmozhnosti-yazyka-html.html> (дата обращение: 21.07.2018) [↑](#)
34. Сакулин, С.А. Основы интернет-технологий: HTML, CSS, JavaScript, XML: учебное пособие/ С.А. Сакулин. — Москва: МГТУ им. Н.Э. Баумана, 2017. — 50 с. [↑](#)
35. Муссиано Ч. HTML и XHTML. Подробное руководство, 6 - е издание/Ч. Муссиано, Б. Кеннеди - СПб: Символ-Плюс, 2016. - 39 с. [↑](#)

36. Чиртик А. HTML: Популярный самоучитель/ А. Чиртик. -- СПб: Питер, 2014. -53 с. [↑](#)
37. Чиртик А. HTML: Популярный самоучитель/ А. Чиртик. -- СПб: Питер, 2014. - 54 с. [↑](#)
38. Сакулин, С.А. Основы интернет-технологий: HTML, CSS, JavaScript, XML: учебное пособие/ С.А. Сакулин. — Москва: МГТУ им. Н.Э. Баумана, 2017. — 52 с. [↑](#)
39. Муссиано Ч. HTML и XHTML. Подробное руководство, 6 - е издание/Ч. Муссиано, Б. Кеннеди - СПб: Символ-Плюс, 2016. - 42 с. [↑](#)
40. Муссиано Ч. HTML и XHTML. Подробное руководство, 6 - е издание/Ч. Муссиано, Б. Кеннеди - СПб: Символ-Плюс, 2016. - 44 с. [↑](#)
41. Чиртик А. HTML: Популярный самоучитель/ А. Чиртик. -- СПб: Питер, 2014. -55 с. [↑](#)
42. Сакулин, С.А. Основы интернет-технологий: HTML, CSS, JavaScript, XML: учебное пособие/ С.А. Сакулин. — Москва: МГТУ им. Н.Э. Баумана, 2017. — 53 с. [↑](#)
43. Сакулин, С.А. Основы интернет-технологий: HTML, CSS, JavaScript, XML: учебное пособие/ С.А. Сакулин. — Москва: МГТУ им. Н.Э. Баумана, 2017. — 57 с. [↑](#)
44. Чиртик А. HTML: Популярный самоучитель/ А. Чиртик. -- СПб: Питер, 2014. -54 с. [↑](#)
45. Что такое HTML? URL: <https://zametkinapolyah.ru/verstka-sajtov/chto-takoe-html-dlya-chego-nuzhen-html-kak-sozdat-html-stranicu-vozmozhnosti-yazyka-html.html> (дата обращения: 21.07.2018) [↑](#)
46. Чиртик А. HTML: Популярный самоучитель/ А. Чиртик. -- СПб: Питер, 2014. -54 с. [↑](#)

47. Сакулин, С.А. Основы интернет-технологий: HTML, CSS, JavaScript, XML: учебное пособие/ С.А. Сакулин. — Москва: МГТУ им. Н.Э. Баумана, 2017. — 59 с. [↑](#)
48. Муссиано Ч. HTML и XHTML. Подробное руководство, 6 - е издание/Ч. Муссиано, Б. Кеннеди - СПб: Символ-Плюс, 2016. - 45 с. [↑](#)
49. Как изменить цвет фона и текста веб - страницы? URL: <http://htmlbook.ru/faq/kak-izmenit-tsvet-fona-i-teksta-veb-stranitsy> (дата обращения: 21.07.2018) [↑](#)
50. Сакулин, С.А. Основы интернет-технологий: HTML, CSS, JavaScript, XML: учебное пособие/ С.А. Сакулин. — Москва: МГТУ им. Н.Э. Баумана, 2017. — 63 с. [↑](#)
51. Чиртик А. HTML: Популярный самоучитель/ А. Чиртик. -- СПб: Питер, 2014. - 56 с. [↑](#)
52. Чиртик А. HTML: Популярный самоучитель/ А. Чиртик. -- СПб: Питер, 2014. -60 с. [↑](#)
53. Сакулин, С.А. Основы интернет-технологий: HTML, CSS, JavaScript, XML: учебное пособие/ С.А. Сакулин. — Москва: МГТУ им. Н.Э. Баумана, 2017. — 64 с. [↑](#)
54. Муссиано Ч. HTML и XHTML. Подробное руководство, 6 - е издание/Ч. Муссиано, Б. Кеннеди - СПб: Символ-Плюс, 2016. - 69 с. [↑](#)
55. Муссиано Ч. HTML и XHTML. Подробное руководство, 6 - е издание/Ч. Муссиано, Б. Кеннеди - СПб: Символ-Плюс, 2016. - 69 с. [↑](#)
56. Сакулин, С.А. Основы интернет-технологий: HTML, CSS, JavaScript, XML: учебное пособие/ С.А. Сакулин. — Москва: МГТУ им. Н.Э. Баумана, 2017. — 65 с. [↑](#)
57. Чиртик А. HTML: Популярный самоучитель/ А. Чиртик. -- СПб: Питер, 2014. -61 с. [↑](#)

58. Муссиано Ч. HTML и XHTML. Подробное руководство, 6 – е издание/Ч. Муссиано, Б. Кеннеди – СПб: Символ-Плюс, 2016. – 72 с. [↑](#)
59. Сакулин, С.А. Основы интернет-технологий: HTML, CSS, JavaScript, XML: учебное пособие/ С.А. Сакулин. — Москва: МГТУ им. Н.Э. Баумана, 2017. — 70 с. [↑](#)
60. Чиртик А. HTML: Популярный самоучитель/ А. Чиртик. -- СПб: Питер, 2014. -84 с. [↑](#)
61. Сравнение HTML и XHTML. URL: <http://www.internet-technologies.ru/articles/sravnenie-html-i-xhtml.html> (дата обращения: 21.07.2018) [↑](#)
62. XHTML. URL: <http://htmlbook.ru/xhtml> (дата обращения: 21.07.2018) [↑](#)
63. Муссиано Ч. HTML и XHTML. Подробное руководство, 6 – е издание/Ч. Муссиано, Б. Кеннеди – СПб: Символ-Плюс, 2016. – 372 с. [↑](#)
64. Муссиано Ч. HTML и XHTML. Подробное руководство, 6 – е издание/Ч. Муссиано, Б. Кеннеди – СПб: Символ-Плюс, 2016. – 374 с. [↑](#)
65. Основы XHTML. Основное различие между HTML и XHTML. URL: <http://megaweb.su/osnovi/xhtml/12-xhtml-osnovy-osnovnoe-razlichie-mezhdu-html-i-xhtml.html> (дата обращения: 21.07.2018) [↑](#)
66. Сравнение HTML и XHTML. URL: <http://www.internet-technologies.ru/articles/sravnenie-html-i-xhtml.html> (дата обращения: 21.07.2018) [↑](#)
67. Муссиано Ч. HTML и XHTML. Подробное руководство, 6 – е издание/Ч. Муссиано, Б. Кеннеди – СПб: Символ-Плюс, 2016. – 390 с. [↑](#)
68. Основы XHTML. Основное различие между HTML и XHTML. URL: <http://megaweb.su/osnovi/xhtml/12-xhtml-osnovy-osnovnoe-razlichie-mezhdu-html-i-xhtml.html>

xhtml.html (дата обращения: 21.07.2018) [↑](#)

69. Сравнение HTML и XHTML. URL: <http://www.internet-technologies.ru/articles/sravnenie-html-i-xhtml.html> (дата обращения: 21.07.2018) [↑](#)
70. Сравнение HTML и XHTML. URL: <http://www.internet-technologies.ru/articles/sravnenie-html-i-xhtml.html> (дата обращения: 21.07.2018) [↑](#)
71. Основы XHTML. Основное различие между HTML и XHTML. URL: <http://megaweb.su/osnovi/xhtml/12-xhtml-osnovy-osnovnoe-razlichie-mezhdu-html-i-xhtml.html> (дата обращения: 21.07.2018) [↑](#)
72. XHTML. URL: <http://htmlbook.ru/xhtml> (дата обращения: 21.07.2018) [↑](#)
73. Сравнение HTML и XHTML. URL: <http://www.internet-technologies.ru/articles/sravnenie-html-i-xhtml.html> (дата обращения: 21.07.2018) [↑](#)
74. Муссиано Ч. HTML и XHTML. Подробное руководство, 6 – е издание/Ч. Муссиано, Б. Кеннеди – СПб: Символ-Плюс, 2016. – 393 с. [↑](#)
75. Основы XHTML. Основное различие между HTML и XHTML. URL: <http://megaweb.su/osnovi/xhtml/12-xhtml-osnovy-osnovnoe-razlichie-mezhdu-html-i-xhtml.html> (дата обращения: 21.07.2018) [↑](#)
76. Муссиано Ч. HTML и XHTML. Подробное руководство, 6 – е издание/Ч. Муссиано, Б. Кеннеди – СПб: Символ-Плюс, 2016. – 752 с. [↑](#)
77. XHTML. URL: <http://htmlbook.ru/xhtml> (дата обращения: 21.07.2018) [↑](#)
78. Основы XHTML. Основное различие между HTML и XHTML. URL: <http://megaweb.su/osnovi/xhtml/12-xhtml-osnovy-osnovnoe-razlichie-mezhdu-html-i-xhtml.html>



- xhtml.html (дата обращения: 21.07.2018) [↑](#)
79. Муссиано Ч. HTML и XHTML. Подробное руководство, 6 – е издание/Ч. Муссиано, Б. Кеннеди – СПб: Символ-Плюс, 2016. –402 с. [↑](#)
80. XHTML. URL: <http://htmlbook.ru/xhtml> (дата обращения: 21.07.2018) [↑](#)
81. Сравнение HTML и XHTML. URL: <http://www.internet-technologies.ru/articles/sravnenie-html-i-xhtml.html> (дата обращения: 21.07.2018) [↑](#)
82. Сравнение HTML и XHTML. URL: <http://www.internet-technologies.ru/articles/sravnenie-html-i-xhtml.html> (дата обращения: 21.07.2018) [↑](#)
83. Основы XHTML. Основное различие между HTML и XHTML. URL: <http://megaweb.su/osnovi/xhtml/12-xhtml-osnovy-osnovnoe-razlichie-mezhdu-html-i-xhtml.html> (дата обращения: 21.07.2018) [↑](#)
84. Муссиано Ч. HTML и XHTML. Подробное руководство, 6 – е издание/Ч. Муссиано, Б. Кеннеди – СПб: Символ-Плюс, 2016. – 410 с. [↑](#)
85. Муссиано Ч. HTML и XHTML. Подробное руководство, 6 – е издание/Ч. Муссиано, Б. Кеннеди – СПб: Символ-Плюс, 2016. – 752 с. [↑](#)
86. XHTML. URL: <http://htmlbook.ru/xhtml> (дата обращения: 21.07.2018) [↑](#)
87. Основы XHTML. Основное различие между HTML и XHTML. URL: <http://megaweb.su/osnovi/xhtml/12-xhtml-osnovy-osnovnoe-razlichie-mezhdu-html-i-xhtml.html> (дата обращения: 21.07.2018) [↑](#)
88. Основы XHTML. Основное различие между HTML и XHTML. URL: <http://megaweb.su/osnovi/xhtml/12-xhtml-osnovy-osnovnoe-razlichie-mezhdu-html-i-xhtml.html> (дата обращения: 21.07.2018) [↑](#)

89. XHTML. URL: <http://htmlbook.ru/xhtml> (дата обращения: 21.07.2018) [↑](#)

90. Муссиано Ч. HTML и XHTML. Подробное руководство, 6 - е издание/Ч. Муссиано, Б. Кеннеди - СПб: Символ-Плюс, 2016. - 521 с. [↑](#)