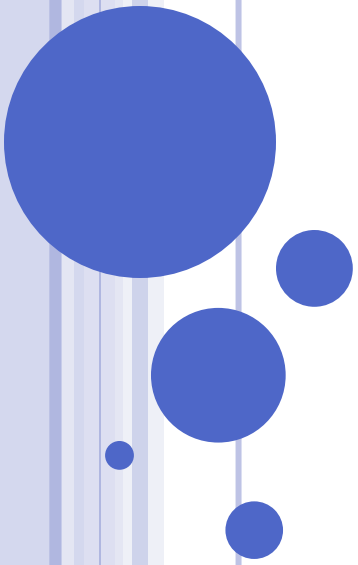
# Язык HTML



# HyperText Markup Language

(язык разметки гипертекста)

- Понятие HTML включает в себя различные способы оформления гипертекстовых документов, дизайн, гипертекстовые редакторы, браузеры и многое другое.
- HTML как основа создания Web-страниц имеет прямое отношение и к новому направлению изобразительного искусства — Web-дизайну.

- Многие называют HTML языком программирования. Это не совсем верно, так как в традиционном понимании HTML является языком *разметки* электронных документов, лишь указывающим программам просмотра HTML-страниц (браузерам) форму представления описанной в документе информации.

# Браузер

- Специальные программы для просмотра веб-сайтов (электронных документов, созданных по правилам языка разметки HTML), называются браузерами.
- Основная функция браузера заключается в интерпретации кода HTML и выводе визуального результата на экран монитора пользователя.
- Сегодня существует большое количество самых разнообразных браузеров

- Самым первым обозревателем с графическим интерфейсом является Mosaic от фирмы NCSA.
- Более продвинутым был второй браузер — Netscape Navigator.

Наибольшей популярностью пользуются программы:

- Internet Explorer,
- Opera,
- Mozilla,
- Google Chrome (На январь 2014 года — 32%)

# Специальные термины

- **Элемент** (element) — конструкция языка HTML. Это контейнер, содержащий данные и позволяющий отформатировать их определенным образом. Любая
- **Web-страница** представляет собой набор элементов. Одна из основных идей гипертекста — возможность вложения элементов.
- **Тег (tag)** — начальный или конечный маркеры элемента. Теги определяют границы действия элементов и отделяют элементы друг от друга. В тексте Web-страницы теги заключаются в угловые скобки, а конечный тег всегда снабжается косой чертой.

- **Атрибут** (attribute) — параметр или свойство элемента. Это переменная, которая имеет стандартное имя и которой может присваиваться определенный набор значений. Атрибуты располагаются внутри начального тега и отделяются друг от друга пробелами.

• **Гиперссылка** — фрагмент текста, который является указателем на другой файл или объект. Гиперссылки необходимы для того, чтобы обеспечить возможность перехода от одного документа к другому.

• **Фрейм** (frame) — этот термин имеет два значения. Первое — область документа со своими полосами прокрутки. Второе значение — одиночное изображение в сложном (анимационном) графическом файле (по аналогии с кадром кинофильма). Вместо термина ≪фрейм≫ в специальной литературе и локализованных программных продуктах иногда можно встретить термин ≪кадр≫ или ≪рамка≫.

- **Апплет** (applet) — программа, передаваемая на компьютер клиента в виде отдельного файла и запускаемая при просмотре Web-страницы.
- **Скрипт или сценарий** (script) — программа, включенная в состав Web-страницы для расширения ее возможностей. Броузер Internet Explorer в определенных ситуациях выводит сообщение: ≪Разрешить выполнение сценариев на странице?≫ В этом случае имеются в виду скрипты.

- **Расширение** (extension) — элемент, не входящий в спецификацию языка, но использующийся, обеспечивая возможность создания нового интересного эффекта форматирования.
- **Программный код** или просто код — аналог понятия ≪текст программы≫.
- **Код HTML** — гипертекстовый документ в своем первоначальном виде, когда видны все элементы и атрибуты.

- **World Wide Web, WWW** или просто Web — Всемирная паутина, распределенная система доступа к гипертекстовым документам, существующая в Интернете.

• **Web-страница** — документ (файл), подготовленный в формате гипертекста и размещенный в World Wide Web.

• **Сайт** (site) — набор Web-страниц, принадлежащих одному владельцу.

- Код **http** указывает на то, что программа должна работать с системой гипертекстовых документов и использовать соответствующий протокол (HyperText Transfer Protocol).

# HTML-файл или HTML-страница

— документ, созданный в виде гипертекста

на основе языка HTML. Такие файлы имеют, как правило, расширения htm или html. В гипертекстовых редакторах и браузерах эти файлы имеют общее название ≪документ≫.

HTML является основным языком для создания документов в WWW.

# Особенности гипертекста

- Способ создания гипертекста обеспечивает его абсолютную платформенную независимость.
- Одной из основных особенностей HTML является принцип, по которому не только допускается вложение одних элементов в другие, но и декларируется необходимость такого вложения. Это отличает HTML от многих других языков.

- Если рассмотреть исходные тексты различных Web-страниц, то можно легко увидеть схожесть их структур. Это объясняется тем, что документы создаются по определенным правилам.
- В основу синтаксиса языка HTML лег **стандарт** ISO 8879:1986
- ≪Information processing. Text and office systems. Standard Generalized Markup Language (SGML)≫.

# Язык HTML представляет собой набор специальных правил.

- Каждому правилу соответствует свое название, свойство и значение. Например, чтобы задать правило жирного начертания обыкновенного текста, необходимо использовать следующую HTML-конструкцию:

**<B>Обыкновенный текст</B>**

- Как видно из примера, текст, который должен отображаться жирным начертанием, обособлен группами символов <в> и </в>.
- Такие группы принято называть тегами.
- Теги бывают одинарными и парными. В случае с нашим примером тег </в> является парным, т. к. он закрывает HTML-конструкцию вместе с символом "/" (прямой слэш). Иногда теги, которые необходимо закрывать парным тегом, называют тегами-контейнерами.

# Все элементы языка можно условно разделить на три группы.

- К первой относятся элементы, которые создают **структуру** гипертекстового документа.
- Ко второй группе можно отнести элементы, создающие **эффекты форматирования**. Их использование диктуется конкретными требованиями к документу, фантазией разработчика.
- К третьей группе относятся элементы, которые позволяют **управлять** программными средствами, установленными и работающими на компьютере-клиенте.

- Структура любого тега подразумевает указание самого тега, его параметра и значения этого параметра.
- При этом наименование параметра и его значения может писаться как строчными, так и прописными буквами.
- Значение параметра ставится в кавычки.

- Параметры и их значения могут либо вообще отсутствовать у какого-то определенного тега:

<TABLE WIDTH="100%" BORDER>

- В приведенном примере для тега построения таблицы <TABLE> один параметр (WIDTH) указан со значением, другой (BORDER) такового не содержит.

- Любой HTML-документ содержит три основных обязательных раздела: HTML, HEAD и BODY. Рассмотрим подробнее каждый из них.

# Раздел *HTML*

- Раздел HTML описывается тегом-контейнером <HTMLX/HTML> и дает браузеру информацию о том, что документ разработан с помощью языка разметки HTML.
- Сегодня большинство браузеров способно распознать HTML-документ и без указания данного тега, тем не менее, пропускать раздел HTML разработчикам не рекомендуется.

# <HTML> </HTML>

- Данный элемент является самым внешним, так как между его начальным и конечным тегом должна находиться вся Web-страница.

- Он допускает вложение элементов HEAD, BODY, FRAMESET и других, определяющих общую структуру Web-страницы.
- Конечным тегом </html> заканчиваются все подобные документы.

# Раздел *HEAD*

- Раздел HEAD выполняет функцию рабочего заголовка HTML-документа и является, по сути, "бойцом невидимого фронта" - - теги, указываемые внутри этого раздела, чрезвычайно важны и могут сильно влиять на внешний вид документа, но сами остаются незаметными глазу пользователя.
- Данному разделу сопоставлен парный тег <HEADX/HEAD>.
- Рассмотрим теги HTML, которые указываются внутри раздела HEAD.

# <HEAD> </HEAD>

- Область заголовка Web-страницы. Иными словами, ее первая часть. Так же, как предыдущий элемент, HEAD служит только для формирования общей структуры документа. Этот элемент может иметь атрибуты lang и d i r , должен включать элемент TITLE и допускает вложение элементов BASE, META, LINK, OBJECT, SCRIPT, STYLE.

# Название документа <TITLE>

- Парный тег <TITLEX/TITLE> предназначен для указания имени созданному электронному документу. Следует помнить, что под именем документа в данном случае имеется в виду не файловое наименование, а визуальный заголовок HTML-страницы.
- Указание конструкции <TITLEX/TITLE> не является обязательным, однако рекомендуется
- Эта строка часто используется при организации поиска в WWW. Поэтому авторы должны позаботиться о том, чтобы эта строка, не будучи слишком длинной, достаточно точно отражала назначение документа.

# <TITLE> </TITLE>

- Поисковые системы, столкнувшись с безымянной страницей, занесут ее в свои базы данных под заголовком Untitled, что сделает HTML-документ безликим и похожим на миллионы других электронных документов, размещенных в Интернете.

# Связь между документами <*LINK*>

- Часто бывает так, что несколько различных документов структурно или функционально взаимосвязаны между собой. В этом случае используется тег <LINK>, который не является парным, а значит не требует указания закрывающего тега:
- <LINK HREF="styles/main.ess" TYPE="text/css" REL="stylesheet">
- Приведенный пример показывает отношение между HTML-документом, в разделе HEAD которого указана данная кодовая конструкция, и внешним файлом шаблона стилей main.ess.

- **Элементы <*STYLE*> и <*SCRIPT*>**
- В структуру раздела документа HEAD также могут входить теги-контейнеры
- <STYLEX/STYLE> И <SCRIPTX/SCRIPT>.
- первый парный тэг описывает стилевые шаблоны документа, второй — содержит код исполняемых сценариев (скриптов).

# <STYLE> </STYLE>

- **Описание стиля некоторых элементов Web-страницы.**
- Для каждого элемента существует стилевое оформление по умолчанию, поэтому употребление элемента STYLE не обязательно, но желательно.

# <META>

- Этот элемент **содержит служебную информацию**, которая не отражается при просмотре Web-страницы. Внутри него нет текста в обычном понимании, поэтому нет и конечного тега. Каждый элемент META содержит два основных атрибута, первый из которых определяет тип данных, а второй — содержание.

# Вот несколько примеров мета-данных:

- Дата, обозначающая ≪срок годности≫ документа:

name="Expires" content=«Дата"

- Адрес электронной почты:

name="Reply-to" content=*"Имя@Адрес"*

- Имя автора Web-страницы:

name ="Author" content="*Имя автора"*

- Набор ключевых слов для поиска:

name="Keywords" *content="слово1, слово2, слово3 . .  "*

- Краткое описание содержания Web-страницы:

name="Description" content="Содержание *страницы"*

- Описание типа и характеристик Web-страницы:

name="Content-Type" content="*Описание страницы"*

- Указание приложения, в котором была создана Web-страница:

name="Generator" content=«Название *HTML-редактора"*

# <BODY> </BODY>

- **Этот элемент заключает в себе гипертекст**, который определяет собственно Web-страницу. Раздел BODY является одним из самых важных компонентов любого HTML-документа, т. к. в нем располагается содержательная часть, которая выводится браузером на экран монитора пользователя.
- Конечный тег этого элемента надо искать в конце HTML-файла.
- Внутри элемента BODY можно использовать все элементы, предназначенные для дизайна Web-страницы.

- Раздел описывается парным тегом <BODYX/BODY>, внутри которого размещается большинство существующих тегов HTML. Гипертекст, расположенный внутри элемента BODY, может иметь произвольную структуру.
- Тег <BODY> имеет ряд параметров, которые условно можно разделить на четыре основные группы (параметры фона, границ документа, текста и гиперссылок).

**Параметр** **Функция**

- BGCOLOR Определение цвета фона
- BACKGROUND Указание фонового рисунка
- BGPROPERTIES Изменение свойств фона (например, фиксирование фонового рисунка)
- TOPMARGIN, BOTTOMMARGIN Определение размера отступов
- LEFTMARGIN, RIGHTMARGIN,
- MARGINWIDTH, MARGINHEIGHT
- TEXT Определение цвета основного текста
- LINK, ALINK, VLINK Определение цвета гиперссылок

- Один из самых полезных для дизайна — **атрибут, определяющий фон страницы**:

background=«Путь *к файлу фона"*

- Более простое оформление фона сводится к заданию его **цвета**:

bgcolor="#RRGGBB"

- Цвет фона задается тремя двуразрядными шестнадцатеричными числами, которые определяют интенсивность красного, зеленого и синего цветов соответственно.

# Параметры фона

- Параметрами фона документа являются BGCOLOR, BACKGROUND и BGPROPERTIES.
- BGCOLOR устанавливает цвет фона, значение которого может быть введено в символьном эквиваленте, в шестнадцатеричном коде или в формате цветовой модели RGB (Red, Green, Blue).
- Любое значение RGB может быть преобразовано в шестнадцатеричный формат(от 00 до FF с приставкой # (читается "диез")).
- Некоторым значениям упомянутых моделей соответствует символьное название цвета.
- Таким образом, один и тот же цвет можно указать тремя возможными способами.

# Три варианта установки цвета фона (белого):

- П <BODY BGCOLOR="white">
- П <BODY BGCOLOR="#FFFFFF">
- П <BODY BGCOLOR="255,255,255">

- Параметр BACKGROUND позволяет накладывать на фон документа графическое изображение:

  <BODY BACKGROUND="images/bg.gif">

- Наряду с графическим изображением фона рекомендуется использовать и параметры цвета на тот случай, если рисунок не загрузится (тогда браузер отобразит цвет).

# Параметры границ документа

- Параметры границ HTML-документа создают отступы заданного размера от верхнего, нижнего, левого и правого краев документа. Этим отступам соответствуют параметры TOPMARGIN, BOTTOMMARGIN, LEFT-MARGIN и RIGHTMARGIN.
- Значения для них задаются в пикселах:

```
<BODY TOPMARGIN="5" BOTTOMMARGIN="5"
   LEFTMARGIN="10" RIGHTMARGIM="10">
```

# Параметры текста

- Из параметров текста документа реально применяется только один — TEXT.
- Он задает цвет основного текста на странице (значение параметра может быть введено аналогично цвету фона документа):

<BODY TEXT="black">

# Параметры гиперссылок

- Параметры гиперссылок (связей с внутренними или внешними документами) определяют цвет активных (ALINK), не посещенных (LINK) и посещенных (VLINK) ссылок:

```
<BODY LINK="#OOOOFF" ALINK="#OOOOFF"
                    VLINK="blue">
```

- Для задания **цвета текста** имеется следующий атрибут:

text="#RRGGBB"

- Для задания **цвета текста гиперссылок** используется атрибут:

link="#RRGGBB"

- Точно так же можно задать **цвет для просмотренных гиперссылок**:

vlink="#RRGGBB"

- Можно также указать изменение **цвета для последней выбранной пользователем гиперссылки**:

alink="#RRGGBB"

# <!-- Комментарий — >

- В любом языке программирования есть конструкции, позволяющие создавать произвольные ремарки.
- Текст, введенный внутри этого элемента, игнорируется браузером. Эти элементы могут располагаться в любом месте Web-страницы.
- Комментарий должен быть отделен от основного текста угловыми скобками.
- Признаком комментария служит восклицательный знак, а текст комментария должен обрамляться двойными дефисом. Например:

**<!-- Начало вывода таблицы -->**

# <Hlx/hl>

- Элемент заголовка. Существует шесть уровней заголовков, которые обозначаются H1…H6.
- Заголовок уровня 1 самый крупный, а уровень 6 обеспечивает самый маленький заголовок.
- Для заголовков можно использовать атрибут, задающий выравнивание влево, по центру или вправо:
- align="left"
- align="center"
- align="right"

# Заголовки, создаваемые при помощи элементов H1…H6.

- **Заголовок 1**
- ***Заголовок 2***
- **Заголовок 3**
- **Заголовок 4**
- **Заголовок 5**

Шрифт второго заголовка переопределен

# <HR>

- Горизонтальная линия (horizontal rule) — очень часто используемый элемент.
- Во-первых, потому что с его помощью очень удобно делить страницу на части.
- Во-вторых, потому что выбор подобных элементов оформления у автора страницы очень небольшой. Действительно, в HTML практически отсутствуют похожие конструкции, только для горизонтальной линии почему-то было сделано исключение.

- Элемент не имеет конечного тега, но допускает ряд атрибутов для выравнивания
- влево, по центру, вправо, по ширине:
- align="left"
- align="center"
- align="right"
- align="justify"

- Можно задавать толщину линии:

*Size=толщина в пикселах*

- Можно управлять длиной линии:

width= длина в пикселах

width=длина в процентах

- Можно выбрать цвет:

co1or="цвет"

# Правила синтаксиса

- При использовании каждого элемента важно знать, какие элементы могут располагаться *внутри него* и *внутри каких* элементов может находиться он сам. Так, взаимное расположение элементов HTML, HEAD, TITLE и BODY должно быть стандартным на любой странице.
- Если же страница представляет собой документ планировки фреймов, то вместо элемента BODY используется элемент FRAMESET.

- Существуют группы элементов, которые используются совместно. К ним относятся элементы для создания таблиц, списков, фреймов.
- Таблицы и фреймы часто используются для того, чтобы разместить детали страницы (рисунки, текст и т. д.) в определенном порядке. Например, располагая рисунок *внутри* ячейки таблицы, можно добиться определенного его положения.

- В приведенном ниже примере есть два абзаца (первый в зеленой рамке) и таблица:
- <P style="border: 3px solid green">Текст абзаца 1</p>
- <TABLE> . . . </table>
- <P> Текст абзаца 2</p>
- Таблица в данном случае — независимый элемент. Ее можно, например, выравнивать независимо от остального текста.

- Можно использовать другой код:
- <P style="border: 3px solid green">Текст абзаца 1
- <TABLE> . . . </table>
- <P> Текст
- Исчез конечный тег первого абзаца. Теперь таблица является частью первого абзаца, и зеленая рамка будет охватывать таблицу и текст. абзаца 2</p>

- Браузеры построены таким образом, что они ≪стараются≫ не реагировать на ошибки разметки гипертекста. Если страница может быть отображена, то она выводится на экран без каких-либо предупреждающих сообщений.
- Программа интерпретирует ошибочно расставленные теги определенным образом и формирует изображение, следуя логике, заложенной в нее разработчиками. При этом вид страницы может и не соответствовать замыслу автора. И только в случае очень серьезных ошибок или явных противоречий браузер выводит сообщение о невозможности отобразить страницу. Косвенным признаком ошибки разметки может служить появление на странице фрагментов кода HTML.

- Полезная информация одного элемента *должна* находиться **или между начальным и конечным тегами** данного элемента, или **между начальным тегом** данного **и начальным тегом** следующего элемента.
- Любой произвольный текст, введенный на страницу, воспринимается браузером как подлежащий выводу на экран и, следовательно, форматированию в соответствии с окружающими этот текст элементами. При этом не учитывается разбиение текста на строки, полученное в текстовом редакторе.

- Очень важным правилом, которое не имеет исключений, является размещение атрибутов элемента *внутри начального тега*.

# Первый HTML-документ

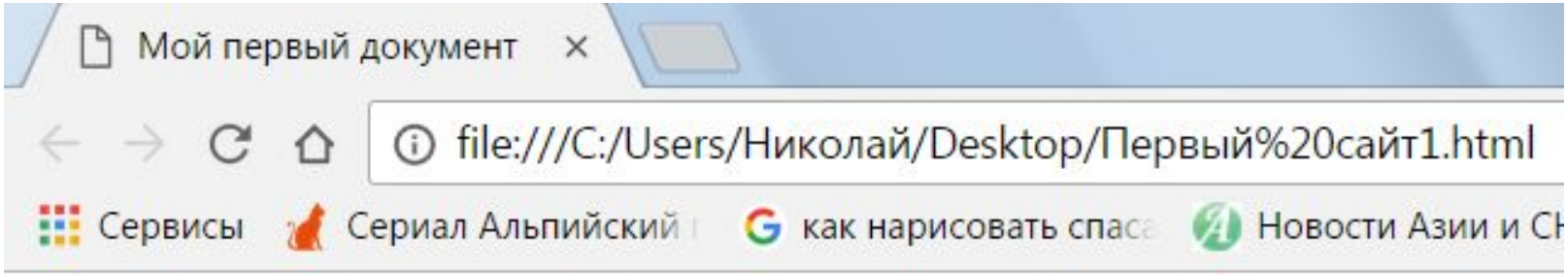
```
<HTML>
<HEAD>
<TITLE>Мой первый документ</TITLE>
</HEAD>
<BODY BGCOLOR="#FFFFFF" TOPMARGIN="30"
TEXT="black" LINK="#OOFFOO"
ALINK="#OOFFOO" VLINK="blue">
Это мой первый HTML-документ!
</BODY>
</HTML>
```

- Код можно написать в блокноте.
- Теперь сохраните код под любым именем, не забыв присвоить файлу расширение html или htm. Откройте файл в браузере.

Увидите там текст:

- Это мой первый HTML-документ!

# Вот что получилось:




Это мой первый HTML-документ

# Задание: Изменим цвета фона и шрифта:

- <HTML>
- <HEAD>
- <TITLE>Мой первый документ</TITLE>
- </HEAD>
- <BODY **BGCOLOR="#00FFFF"** TOPMARGIN="30" **TEXT="blue"** LINK="#OOFFOO"
- ALINK="#OOFFOO" VLINK="blue">
- Это мой первый HTML-документ
- </BODY>
- </HTML>

# Задание: Изменим цвета еще раз:

- <HTML>
- <HEAD>
- <TITLE>Мой первый документ</TITLE>
- </HEAD>
- <BODY **BGCOLOR="#0000FF"** TOPMARGIN="30" **TEXT="red"** LINK="#00FF00"
- ALINK="#OOFFOO" VLINK="blue">
- Это мой первый HTML-документ
- </BODY>
- </HTML>