**15.11.2021 Учебная группа 2СТМ, 4-я пара**

**Преподаватель Иванова Наталия Викторовна**

**ОП.10 Информационные технологии в профессиональной деятельности**

**Тема**: «Форматирование web-страниц»

**Цель занятия:**

*учебные***:** познакомить со способами создания, размещения и продвижения сайтов в Сети; реализовать способности учеников в ходе проектирования и конструирования сайтов; сформировать компетенции информационно грамотного человека в современном обществе;

*развивающие:* развить умения устанавливать причинно-следственные связи, выделять главное, обобщать имеющиеся факты, логически и абстрактно мыслить;

*воспитательные:* формировать устойчивый интерес к предмету

**Задачи занятия:** познакомить с видами веб-сайтов, их функциональными, структурными и технологическими особенностями;создать представление о языке HTML, научить использовать эти технологии для создания веб-страниц; сформировать навыки проектирования веб-сайта; научить пользоваться различными программными решениями для автоматизации своей работы; дать представление о теоретических основах веб-дизайна и реальном процессе создания веб-сайта; сформировать навыки коллективной работы при разработке веб-проекта.

**Мотивация:** для развития собственного бизнеса необходимо качественно себя представить в интернете, этот урок вам поможет понять как создаются страницы сайта, как их редактировать и при этом не надо много платить денег за разработку сайта.

**Задание студентам:**

1. Посмотреть видеоурок<https://youtu.be/_J6hMLsscOo>  **редактор кода** Atom **(**[https://atom.io](https://www.youtube.com/redirect?event=video_description&redir_token=QUFFLUhqbHJWWlZjZENrY3RqSG1lc3A0VEwxdThKZ3dPQXxBQ3Jtc0ttSHJhQ0t4UWZMRmdkazduUzFHdG5pV1BSdVRXYW80OGFQU08zUkhOZXE5R051VVk4QlRyUW5FeEZGLWlua3dtdWI0LXN4Mko1V2VnUlhqRHhJbHJYMkxVN3N3eVh6RzVtbWo2SFVISEdreDZpRDNpbw&q=https%3A%2F%2Fatom.io)**). Этот редактор необходим, чтобы повторить все коды из видео.**
2. Записать в тетрадь основные понятия.
3. Ответить на контрольные вопросы.

Фотографию с выполненным заданием прислать на электронный адрес **atata17@yandex.ru** в срок **до 08.00 16.11.2021** **г.**

**Основная литература:** Михеева Е.В. «Информационные технологии в профессиональной деятельности» : учеб. пособие для студ. сред. проф. образования / Е.В. Михеева. - 7-е изд., стер. - М. : Издательский центр «Академия», 2008. - 384 с.

**Дополнительные источники:** Дыбкова Л.М. «Информатика и компьютерная техника» - М. : Издательство А.С.К., 2003-512 с.: ил.

**Лекция №22**

**План занятия:**

* + - 1. Основы HTML.
			2. Обязательные метки
			3. Структура простейшей страницы.

**Вопрос 1. Основы HTML**

**Что такое HTML?**

Термин HTML (HyperText Markup Language) означает "язык маркировки гипертекстов". Первую версию HTML разработал сотрудник Европейской лаборатории физики элементарных частиц Тим Бернерс-Ли.

**Что необходимо для освоения HTML?**

1. Любой **браузер**, т.е., программа, пригодная для просмотра HTML-файлов.
2. Любой редактор текстовых файлов, поддерживающий русский язык в выбранной Вами кодировке. Если на Вашем компьютере установлен *Windows*, вполне подойдет *Блокнот*.

В настоящее время широко используются два типа редакторов HTML:

1. Редакторы типа "что видишь, то и получишь" (*Браузере Navigator Gold*, *Microsoft Front Page*). Пользователь не видит "внутренностей" документа, с которым он работает точно так же, как при работе с текстовым процессором типа *Microsoft Word* или *Word Perfect*. Кстати говоря, существует довольно много конвертеров, способных преобразовывать документы, созданные в *Microsoft Word* или *Word Perfect*, в HTML-документы.
2. Редакторы HTML-текстов (*HotDog*, *Ken Nesbitt Web Editor* и многие другие). В процессе работы пользователь видит внутреннее содержание HTML-файла и может изменять его либо вручную, либо вызывая команды меню для вставки определенных элементов HTML. Работа с таким редактором очень похожа на работу с интерактивной средой программирования типа *Microsoft Visual Basic* или *Borland Delphi*.

**Как устроен HTML-документ?**

HTML-документ — это просто текстовый файл с расширением\*.html (Unix-системы могут содержать файлы с расширением\*.htmll). Вот самый простой HTML-документ:

<html>

 <head>

 <title>

 Пример 1

 </title>

 </head>

 <body>

 <H1>

 Привет!

 </H1>

 <P>

 Это простейший пример HTML-документа.

 </P>

 <P>

 Этот \*.html-файл может быть

 одновременно открыт и в Блокноте, и в Браузере.

 Сохранив изменения в Блокноте, просто нажмите

 кнопку Reload ('перезагрузить') в Браузере,

 чтобы увидеть эти изменения реализованными

 в HTML-документе.

 </P>

 </body>

</html>

Для удобства чтения введены дополнительные отступы, однако в HTML это совсем не обязательно. Более того, браузеры просто игнорируют символы конца строки и множественные пробелы в HTML-файлах. Поэтому наш пример вполне мог бы выглядеть и вот так:

<html>

<head>

<title>Пример 1</title>

</head>

<body>

<H1>Привет!</H1>

<P>Это простейший пример HTML-документа.</P>

<P>Этот \*.html-файл может быть одновременно открыт и в Блокноте, и в Браузере.

Сохранив изменения в Блокноте, просто нажмите кнопку

 Reload ('перезагрузить') в Браузере, чтобы увидеть

 эти изменения реализованными в HTML-документе.</P>

</body>

</html>

Как видно из примера, вся информация о форматировании документа сосредоточена в его фрагментах, заключенных между знаками "<" и ">". Такой фрагмент (например, <html>) называется меткой (по-английски — tag, читается "тэг").

Большинство HTML-меток — парные, то есть на каждую открывающую метку вида <tag> есть закрывающая метка вида </tag> с тем же именем, но с добавлением "/".

Метки можно вводить как большими, так и маленькими буквами. Например, метки <body>, <BODY> и <Body> будут восприняты браузером одинаково.

Многие метки, помимо имени, могут содержать **атрибуты** — элементы, дающие дополнительную информацию о том, как браузер должен обработать текущую метку. В нашем простейшем документе, однако, нет ни одного атрибута. Но мы обязательно встретимся с атрибутами уже в следующем разделе.

**Вопрос 2. ОБЯЗАТЕЛЬНЫЕ МЕТКИ**

**<html> ... </html>**

Метка <html> должна открывать HTML-документ. Аналогично, метка </html> должна завершать HTML-документ.

**<head> ... </head>**

Эта пара меток указывает на начало и конец заголовка документа. Помимо наименования документа (см. описание метки <title> ниже), в этот раздел может включаться множество служебной информации, о которой мы обязательно поговорим чуть позже.

**<title> ... </title>**

Все, что находится между метками <title> и </title>, толкуется браузером как название документа. *Браузера*, например, показывает название текущего документа в заголовке окна и печатает его в левом верхнем углу каждой страницы при выводе на принтер. Рекомендуется название не длиннее 64 символов.

**<body> ... </body>**

Эта пара меток указывает на начало и конец тела HTML-документа, каковое тело, собственно, и определяет содержание документа.

**<H1> ... </H1> — <H6> ... </H6>**

Метки вида <Hi> (где i — цифра от 1 до 6) описывают заголовки шести различных уровней. Заголовок первого уровня — самый крупный, шестого уровня, естественно — самый мелкий.

**<P> ... </P>**

Такая пара меток описывает абзац. Все, что заключено между <P> и </P>, воспринимается как один абзац.

Метки <Hi> и <P> могут содержать дополнительный атрибут ALIGN (читается "элайн", от английского "выравнивать"), например:

<H1 ALIGN=CENTER> Выравнивание заголовка по центру</H1>

или

<P ALIGN=RIGHT>Образец абзаца с выравниванием по правому краю</P>

## Вопрос 3. СТРУКТУРА ПРОСТЕЙШЕЙ СТРАНИЦЫ

Страница сайта - это обычный текстовый файл с расширением **.html**. Внутри этого файла и хранится текст HTML страницы вместе с тегами. Этот файл обязательно должен иметь следующие теги: тег **<html>**, который должен содержать в себе текст всего сайта (все, что написано вне этого тега, браузером будет проигнорировано), а внутри него должны быть еще два тега: тег **<head>** для служебного содержимого страницы и тег **<body>** - для основного текста, который и виден на экране браузера.

В служебное содержимое, которое располагается внутри тега **<head>**, входит много различных вещей, но пока нам нужны только две из них. Это тег **<title>**, задающий название страницы, которое будет видно во вкладке браузера, и тег **<meta>**, который задает кодировку страницы (она ставится в атрибуте **charset** и обычно имеет значение **utf-8**, подробнее об этом в видео, которое будет ниже на пару абзацев).

Кроме того, перед тегом **<html>** обычно пишется конструкция **doctype**, которая указывает версию языка HTML, на которой сделан сайт. Актуальная сейчас версия языка имеет номер пять и доктайп для нее должен выглядеть так - **<!DOCTYPE html>**.

Итак, давайте посмотрим на основную структуру страницы (чтобы запустить это пример в браузере - скопируйте его в текстовый файл с расширением **.html** и откройте в браузере):

<!DOCTYPE html>

<html>

 <head>

 <meta charset="utf-8">

 <title>Это заголовок тайтл</title>

 </head>

 <body>

 Это основное содержимое страницы.

 </body>

</html>

Теперь перейдем к изучению полезных тегов, которые следует использовать внутри тега **<body>**. Это будут теги для абзацев, заголовков, списков, ссылок и других полезных вещей. Итак, приступим.

## АБЗАЦЫ

Одним из основных элементов страницы являются **абзацы**. Их можно сравнить с абзацами в книге - каждый абзац начинается с новой строки и имеет так называемую красную строку (это когда первая строка текста абзаца немного отступает вправо). По умолчанию красной строки нет, но ее легко сделать (об этом чуть позже).

Абзац создается с помощью тега **<p>** таким образом:

<!DOCTYPE html>

<html>

 <head>

 <meta charset="utf-8">

 <title>Это заголовок тайтл</title>

 </head>

 <body>

 <p>

 Это абзац.

 </p>

 <p>

 Это еще один абзац.

 </p>

 <p>

 И еще один абзац.

 </p>

 </body>

</html>

## ЗАГОЛОВКИ H1, H2, H3, H4, H5, H6

Кроме абзацев важное значение на странице имеют **заголовки**. Их также можно сравнить с заголовками из книги - каждая глава имеет свой заголовок (название этой главы) и разбита на параграфы, которые тоже имеют свои заголовки.

**Заголовки создаются** с помощью тегов **<h1>**, **<h2>**, **<h3>**, **<h4>**, **<h5>**, **<h6>**. Они имеют разную степень важности. В заголовке **h1** следует располагать **название всей HTML страницы**, в **h2** - название **блоков** страницы, в **h3** - название подблоков и так далее.

Все заголовки по умолчанию жирные и имеют разный размер. Пример:

<!DOCTYPE html>

<html>

 <head>

 <meta charset="utf-8">

 <title>Это заголовок тайтл</title>

 </head>

 <body>

 <h1>Заголовок h1</h1>

 <h2>Заголовок h2</h2>

 <h3>Заголовок h3</h3>

 <h4>Заголовок h4</h4>

 <h5>Заголовок h5</h5>

 <h6>Заголовок h6</h6>

 <p>Это первый абзац.</p>

 <p>Это второй абзац.</p>

 <p>Это третий абзац.</p>

 </body>

</html>

## ЖИРНЫЙ

Заголовки по умолчанию **жирные**. Однако, можно сделать жирным и обычный текст - достаточно взять его в тег **<b>**. Пример:

<!DOCTYPE html>

<html>

 <head>

 <meta charset="utf-8">

 <title>Это заголовок тайтл</title>

 </head>

 <body>

 <p>

 Это обычный текст, а это <b>жирный</b> текст.

 </p>

 </body>

</html>

Обратите внимание на то, что тег **<b>** следует использовать внутри какого-либо другого тега, например абзаца. В этом случае абзацы создают общую структуру страницы (абзацы и заголовки), а тег **b** делает жирным отдельные кусочки текста.

## КУРСИВ

Кроме жирного можно сделать также и *курсив* с помощью тега **<i>**:

<!DOCTYPE html>

<html>

 <head>

 <meta charset="utf-8">

 <title>Это заголовок тайтл</title>

 </head>

 <body>

 <p>

 Это обычный текст, а это <i>курсивный</i> текст.

 </p>

 </body>

</html>

## СПИСКИ

Наряду с абзацами и заголовками существует еще один важный элемент страницы - это **списки**. Такие элементы наверняка знакомы всем пользователям интернета. Они представляют собой перечисление чего-либо (список) по пунктам. Возле каждого пункта списка обычно стоит закрашенный кружок (он называется **маркер** списка).

Списки создаются с помощью тега **<ul>**, внутри которого обязательно должны идти теги **<li>**. Тег **ul** задает сам список, а в теги **li** следует ложить пункты списка (то есть одному li соответствует один маркер списка). Имейте ввиду, что отдельно теги li использовать нельзя. Пример:

<!DOCTYPE html>

<html>

 <head>

 <meta charset="utf-8">

 <title>Это заголовок тайтл</title>

 </head>

 <body>

 <ul>

 <li>Первый пункт списка.</li>

 <li>Второй пункт списка.</li>

 <li>Третий пункт списка.</li>

 </ul>

 </body>

</html>

Списки, созданные через тег **ul**, называются **неупорядоченными** списками. Такое название им дано потому, что существуют еще и **упорядоченные** списки, у которых вместо маркеров-точек будут маркеры-цифры. Такие списки вместо тега ul имеют тег **<ol>**, а пункты таких списков также создаются через теги **li**.

<!DOCTYPE html>

<html>

 <head>

 <meta charset="utf-8">

 <title>Это заголовок тайтл</title>

 </head>

 <body>

 <ol>

 <li>Первый пункт списка.</li>

 <li>Второй пункт списка.</li>

 <li>Третий пункт списка.</li>

 </ol>

 </body>

</html>

Удобство упорядоченных списков в том, что можно вставить новый пункт списка в любое место - и нумерация сама перестроится (то есть не приходится за ней следить в случае каких-либо изменений, как было бы в том случае, если бы ее расстановка происходила в ручную).

## ССЫЛКИ

**Ссылки** являются теми элементами, которые делают из интернета интернет. Нажимая на ссылки, мы можем переходить с одной страницы сайта на другу. Если бы их не было - интернет был бы просто набором страниц, никак не связанных друг с другом.

Ссылка создается с помощью тега **<a>**, при этом у нее должен быть обязательный атрибут **href**, в котором хранится адрес той страницы, на которую ведет ссылка. Пример:

<!DOCTYPE html>

<html>

 <head>

 <meta charset="utf-8">

 <title>Это заголовок тайтл</title>

 </head>

 <body>

 <a href="http://phphtml.net">Ссылка на сайт phphtml.net.</a>

 </body>

</html>

Ссылки бывают **абсолютные** и **относительные**, кроме того, они могут вести как на ваш сайт, так и на чужой.

**Относительный адрес** означает, что указание пути на нужный файл или страницу вашего сайта начинается относительно страницы, на которой расположена ссылка, либо относительно корневого каталога сайта. Рассмотрим компоненты (части), из которых может состоять относительный путь:

|  |  |  |
| --- | --- | --- |
| **Части пути** | **Описание** | **Примеры значений** |
| **имя\_файла** | Если в качестве значения атрибута указать только имя файла это значит, что нужный файл находится в той же папке, где и страница со ссылкой. | "page.html" |
| **каталог/** | Если файл, к которому нужно указать путь, расположен в дочернем каталоге относительно файла с ссылкой, это означает, что нам надо спуститься на один уровень вниз (в дочернюю папку текущего каталога), в этом случае путь начинается с указания имени дочернего каталога, после его имени указывается прямой слэш "/", он служит для разделения частей пути, после него указывается имя нужного нам файла.**Примечание:** опуститься можно ровно на столько папок вниз, сколько вы их создали. К примеру, если вы создали папку на 10 уровней ниже корневой, то можете указать путь, который приведет вас вниз на 10 папок. Однако, если у вас так много уровней, это, скорее всего, означает, что организация вашего сайта излишне неудобно составлена. | "каталог/page.html""каталог1/каталог2/page.html" |
| **../** | Если нужно указать, что файл, на который вы ссылаетесь, находится в родительской папке, используйте символы .. (две точки), они означают подняться на один уровень вверх (в родительскую папку текущего каталога). Далее мы указываем прямой слэш "/", чтобы разделить части пути, и пишем имя нашего файла.**Примечание:** символы .. можно использовать сколько угодно раз подряд, используя их, вы поднимаетесь каждый раз на одну папку вверх. Однако, подниматься вверх можно до тех пор, пока не придете в корневую папку своего сайта. Выше этой папки подняться нельзя. | "../page.html""../../page.html""../../../кат1/кат2/page.html" - поднимаемся из текущей папки на три каталога выше и уже из него спускаемся на два уровня ниже к требуемому файлу |
| **/** | Относительный путь не обязательно всегда должен начинаться относительно текущего расположения страницы со ссылкой, он также может начинаться относительно корневого каталога сайта. К примеру, если нужный файл находится в корневом каталоге, путь может начинаться с символа "/", после которого надо всего лишь указать имя нужного файла, который расположен в корневом каталоге.**Примечание:** когда символ "/" указывается первым, это означает начало пути от корневого каталога. | "/page.html""/кат1/кат2/car.png" |

**Абсолютный адрес** обычно применяется для указания пути к файлу, который расположен на другом сетевом ресурсе. Он представляет из себя полный URL-адрес к файлу или странице. Первым делом в адресе указывается используемый протокол, после которого идет название домена (имя сайта).

Например: http://www.пример.ру - так выглядит абсолютный путь к конкретному веб-сайту. http:// - это протокол передачи данных, а www.пример.ру - имя сайта (домен).

Абсолютный адрес можно использовать и на собственном сайте. Однако внутри сайта рекомендуется использовать в качестве значения ссылок относительный путь.

Теперь давайте рассмотрим, что такое **URL**-адрес. Каждая веб-страница в сети Интернет имеет свой собственный уникальный адрес, вот он как раз и называется URL. Аббревиатура **URL** расшифровывается как **U**niform **R**esource **L**ocator (унифицированный адрес ресурса), проще говоря URL - это определитель местонахождения ресурса. Этот способ записи адреса стандартизирован в сети Интернет.

## КАРТИНКИ

Давайте теперь разберемся с тем, как разместить **изображение** на странице вашего сайта. Для этого предназначен тег **<img>**, имеющий обязательный атрибут **src**, в котором хранится путь к файлу картинки.

Пример:

<!DOCTYPE html>

<html>

 <head>

 <meta charset="utf-8">

 <title>Это заголовок тайтл</title>

 </head>

 <body>

 <img src="smile.png">

 </body>

</html>

Обратите внимание на то, что тег **<img>** не требует закрывающего тега.

## ССЫЛКИ В ВИДЕ КАРТИНОК

**Ссылкой** может быть не только текст, но и **картинка** - для этого достаточно тег **<img>**вложить в тег **<a>**, как это сделано в следующем примере. Нажмите на картинку - и вы перейдете по ссылке на сайт phphtml.net (чтобы потом вернуться назад на книжку - нажмите кнопку "назад" в браузере):

<!DOCTYPE html>

<html>

 <head>

 <meta charset="utf-8">

 <title>Это заголовок тайтл</title>

 </head>

 <body>

 <a href="http://phphtml.net"><img src="smile.png"></a>

 </body>

</html>

## РАЗРЫВ СТРОКИ

Вспомните, что будет, если сделать, к примеру, два абзаца рядом - в этом случае текст, который лежит в каждом из них, будет начинаться с **новой** строки.

Пример:

<!DOCTYPE html>

<html>

 <head>

 <meta charset="utf-8">

 <title>Это заголовок тайтл</title>

 </head>

 <body>

 <p>Это первый абзац.</p>

 <p>Это второй абзац.</p>

 </body>

</html>

Бывают, однако, ситуации, когда мы хотели бы, чтобы у нас был один абзац, но некоторый текст в нем начинался с новой строки.

К примеру, необходимо набрать стихотворение, но разбивку строки на отдельные абзацы дело слишком муторное.

Чтобы сделать такое, в том месте, где должен быть разрыв строки, следует написать тег **<br>**. Учтите, что этот тег особенный и не имеет закрывающего.

Пример:

<!DOCTYPE html>

<html>

 <head>

 <meta charset="utf-8">

 <title>Это заголовок тайтл</title>

 </head>

 <body>

 <p>Это первая строка текста, <br> а это вторая.</p>

 </body>

</html>

## КОММЕНТАРИИ HTML

Практически во всех языках программирования есть такое понятие как "комментарии".

**Комментарии** - это такой текст, который будет проигнорирован браузером - на экране он будет не виден, но останется в исходном коде страницы.

Для чего нужны комментарии?

**Во-первых**, для того, чтобы оставлять в исходном коде пометки себе или тому человеку, который будет править ваш код после вас.

**Во-вторых**, с помощью комментариев можно убирать некоторые вещи, которые вы бы сейчас хотели удалить, но потом, может быть, захотите вернуть. Их можно просто временно *закомментировать*, а потом *откомментировать* при необходимости.

Комментарии в HTML оформляются следующим образом: сначала уголок, знак ! и два дефиса - **<!--**, потом текст комментария, а потом два дефиса и уголок - **-->**.

Пример:

<!DOCTYPE html>

<html>

 <head>

 <meta charset="utf-8">

 <title>Это заголовок тайтл</title>

 </head>

 <body>

 <!--

 Это комментарий HTML -

 на странице сайта он не отображается.

 -->Это основное содержимое страницы.

 </body>

</html>

 **Контрольные вопросы**

1. Каково назначение языка разметки HTML?
2. Что понимается под HTML-документом?
3. Что такое теги и элементы HTML?
4. Что такое заголовок и тело документа?
5. Какие категории элементов HTML вам известны?
6. Каково назначение атрибутов и в какой части кода они размещаются?
7. Как создать комментарий?