

Практическое занятие №1 Основы HTML

Задание №1

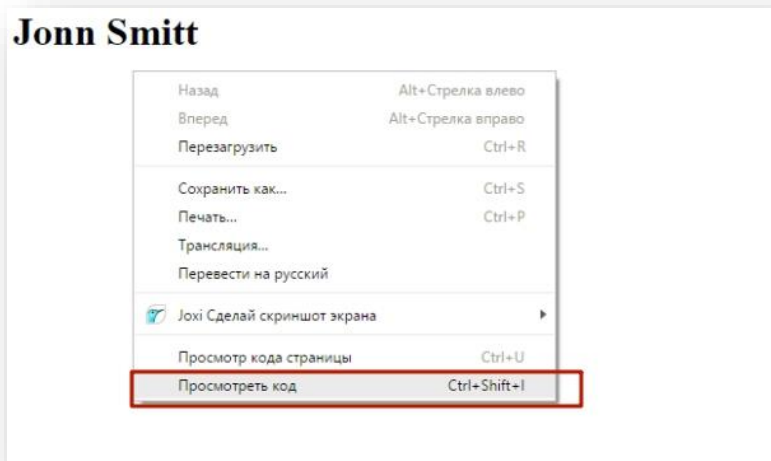
Проверьте запущен ли у Вас Open Server. Папка с материалами урока (html1.loc) должна быть размещена в директории /domains сервера.

Для этого Вам нужно:

1. Перенести архив в директорию ...\\OpenServer\\domains. Распаковать архив в неё.
2. Запустить программу Open Server x64.exe
3. И запустить сам сервер (флажок должен стать зеленым).
4. Если сервер был запущен с самого начала, то его нужно перезапустить.
5. Зайти в браузер Chrome
6. В новой вкладке , перейти по адресу <http://html1.loc/>
7. Убедиться, что всё работает.
8. Нажмите на кнопку "Начнем".
9. Далее, кликните на надпись: "Задание №2".
10. Перед вами будет пустая страница, на которой Вы будете видеть результат своих действий.

Задание №2

1. В папке Open Server найдите файл задания
\\%PATH_TO_OPEN_SERVER%\\OpenServer\\domains\\html1.loc\\tasks\\task-2\\index-2.html.
Аналогичным образом расположены файлы других заданий.
2. Откройте этот файл в текстовом редакторе
3. Между тегами заголовка <h1> и </h1> напишете своё имя.
4. Обновите страницу браузера с заданием, для того что бы увидеть изменения. Обычно на панели вкладок браузера есть кнопка "Обновить страницу". Для удобства можно нажать клавишу "F5" она обновляет страницу в любом браузере.
5. С помощью контекстного меню откройте и просмотрите на код страницы, который получил браузер, он должен совпасть с кодом в Вашем текстовом редакторе.



Задание №3

1. Находим и открываем файл ... html1.loc\\tasks\\task-3\\index-3.html

2. В первой строке размещаем декларацию HTML

```
<!DOCTYPE html>
```

3. Создайте на странице <html> элемент. Для этого после декларации добавьте на страницу открывающий тег <html> и закрывающий тег </html>

```
<!DOCTYPE html>
<html>

</html>
```

4. Создайте ВНУТРИ элемента <html> элемент <head>. Элемент <head> содержит информацию которая не отображается непосредственно на странице.
5. Разместите внутри элемента <head> элемент <title> с текстом “Моя первая HTML страница”. Элемент <title> указывает заголовок вкладки браузера.

```
<!DOCTYPE html>
<html>
  <head>
    <title>My Coding Journal</title>
  </head>
</html>
```

6. Перезагрузите вкладку браузера (F5) что бы увидеть Ваш title.



7. Добавьте элемент <body> после элемента <head>. Элемент <body> отвечает за содержимое самой страницы.
8. Разместите элемент заголовка <h1> в <body> с любым произвольным текстом

```
<h1>Hello World!</h1>
```

9. На главной странице урока(<http://html1.loc/task-list/>), перейдите на Задание №3. Или, если вы уже на странице задания, перезагрузите вкладку браузера (F5) что бы увидеть Ваш title.

Мы получили каркас рабочий HTML страницы!

Задание №4

В HTML определены шесть тегов заголовков от <h1> до <h6>. Заголовок <h1> имеет наибольший вес и его принято использовать один раз на странице. Задача –воссоздать каркас HTML документа из Задания №2 и разместить заголовки от <h1> до <h6> с Вашим именем.

Задание выполнять в файле ...html1.loc\tasks\task-4\index-4.html

Задание №5

Тег <p> определяет параграф в HTML документе. Разбейте следующий текст на параграфы и заголовки на HTML странице.

HTML

HTML - декларативный язык программирования. Используемый для разметки документов в сети интернет. HTML “рендерится” (отображается, исполняется) в браузерах. Таким образом, может отображаться на всех устройствах (мобильные, планшеты, стационарные компьютеры и т.п) где есть установленный браузер.

ТЕГ - элемент разметки, может применяться с настройками (атрибутами) или без них. Текст или другой контент “внутри” тега отображается в зависимости от типа, и названия тега, а также в зависимости от того какие атрибуты “стоят” у тега.

Основные теги

`html` - тег-контейнер для “головы” и “тела” документа. Больше никаких элементов внутри него лежать не должно.

`head` - тег, визуально не отображаемый (не рендерится) браузером. В этот тег вносятся настройки документа, метатеги, теги для распознавания seo, заголовок страницы, подключение стилей и т.п.

`body` - тег определяющий “тело” документа. Именно в этот тег помещаются вся остальная разметка и подключения скриптов `JavaScript`.

Задание №6

С помощью тегов `` и `` создайте маркированный список из названий пяти стран.

```
<ul>
<li>Limes</li>
<li>Tortillas</li>
<li>Chicken</li>
</ul>
```

С помощью тегов `` и `` создайте нумерованный список из 5 различных городов.

Задание №7

Создайте список ссылок, которые ведут на внешние ресурсы

- <https://ru.wikipedia.org/>
- <http://it-news.club/>
- <http://htmlbook.ru/>

```
5
6 <a href="https://ru.wikipedia.org" >Ссылка на википедию</a>
7
```

Добавьте всем ссылкам атрибут `“target”` со значением `“_blank”` и понаблюдайте за изменившимся поведением ссылки.

2. Создайте список из 5 ссылок на страницы предыдущих заданий. При этом, Вы можете опустить домен в адресе ссылки.

```
5
6 <a href="/task-6/" >Предыдущее задание</a>
7
```

Уберите слеш в начале и проверьте что произойдет.

Задание №8

Создания «якоря» ссылки. Ссылки позволяют нам делать не только переходы между сайтами и между страницами одного сайта, но и переходы между блоками одной и той же страницы.

Создайте следующую ссылку в самом верху страницы

```
4 <a href="#bottom" >Вниз</a>
```

После ссылки навестайте следующий фрагмент как на рисунке ниже. В абзацах троеточие замените большими кусками текста с сайта <http://www.lipsum.com/>

```
1  
2 <h1 id="top"></h2>  
3 <p>...</p>  
4 <p>...</p>  
5 <p>...</p>  
6 <p id="bottom"></p>  
7
```

Создайте следующую ссылку в самом низу страницы

```
3  
4 <a href="#top" >Вверх</a>  
5
```

При необходимости уменьшите ширину окна браузера что бы появилась полоса прокрутки. Протестируйте ссылки, какое поведение Вы наблюдаете?

Задание №9

Передача файла по ссылке. С помощью ссылок мы можем передавать файлы из файловой системы сервера. Откройте файл задания и создайте там ссылку на файл, как показано ниже.

```
<a href="/tasks/task-9/CustomerLaw.pdf" >Ссылка на файл</a>
```

Исследуйте поведение этой ссылки.

Задание №10

Откройте папку задания /tasks/task-10/ и навестайте в файле задания index-10.html текст, представленный в текстовом файле «Задание».

Задание №11

Тег `` позволяет добавлять изображения на страницу. Этот элемент особенный потому что он не имеет закрывающего тега. Обязательный атрибут «src» указывает источник изображения.

Вставьте JPEG изображение на страницу как на картинке ниже.

```

```

Следующим распространённым форматом изображений в Web является PNG. Вставьте на страницу PNG изображение как показано ниже.

```

```

Формат GIF используется для вставки анимированных изображений. Вставьте GIF изображение.

```

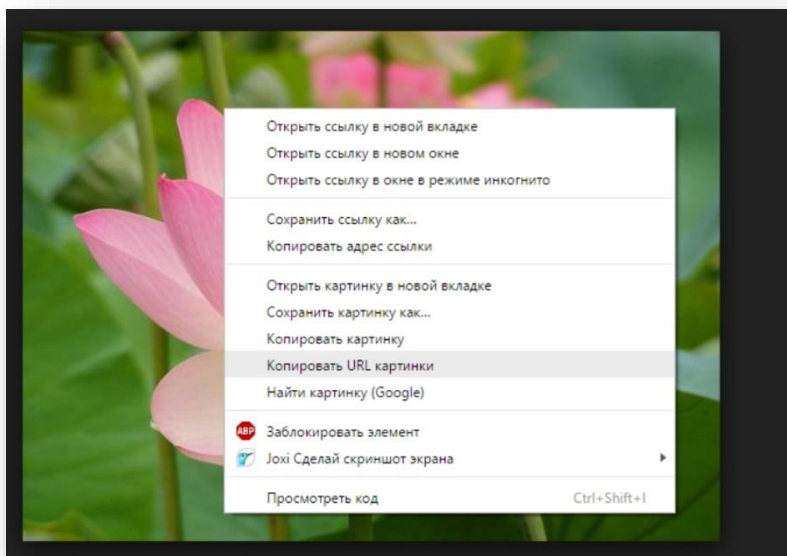
```

Задание №12

В предыдущем уроке мы использовали изображения, которые находятся на одном домене (сайте) непосредственно у нас в папке урока. У нас есть возможность использовать изображения с удаленных серверов, при этом нужно указывать полный путь к изображению.

Вставляем изображение с удалённого сервера на страницу, используя URL
http://www.images.lesyadraw.ru/2014/12/kak_narisovat_krasiviy_cvetok.jpg

В сети находим еще три произвольных изображения и вставляем их на страницу. Что бы узнать URL изображения на удалённом сайте используйте контекстное меню браузера Google Chrome.



Задание №13

Атрибут `alt` содержит описание изображения. Добавьте к изображению атрибут `alt` с текстом: «Изображение автомобиля». Наблюдайте за изменением поведения на странице.

Атрибут `alt` является обязательным для картинок. Это не значит, что без него картинка не будет отображаться, но код будет считаться невалидным, что в некоторых случаях может привести к неприятным последствиям (например, сайт с валидным кодом будет лучше восприниматься поисковиками). Обязательные атрибуты встречаются довольно редко, и, если вы не уверены есть ли они у тега, который вы хотите использовать – проверьте документацию.

Задание №14

Ссылка может быть не только текстом, но и в виде изображения создадим одну такую ссылку как показано ниже.

```
<a href="https://www.google.com.ua/" >  
    
</a>
```

Задайте ссылке атрибут **target** так чтобы ссылка открывалась в новом окне.

Задание №15

Еще одним примером одиночного тега служит тег `
`. Этот тег разрывает строку в том месте где он находится. С помощью тега `
` правильно отформатируйте текст стиха на странице задания.

Задание №16

Комментарии используются для того что бы оставлять подсказки и пометки, а также что бы временно скрыть нежелательные элементы страницы.

Раскомментируйте первый абзац и прокомментируйте второй абзац.

Задание №17

Тег `<sup>` позволяет нам отобразить текст в виде верхнего индекса. Наверстайте формулу, показанную на картинке ниже.

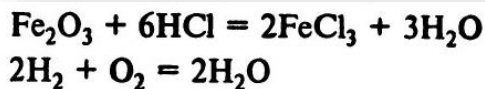
Великая теорема Ферма

$$X^n + Y^n = Z^n$$

где n - целое число > 2

Задание №18

Тег `<sub>` используется для отображения текста в виде нижнего индекса. Наверстайте химические формулы, показанные на картинке ниже.



Задание №19

Мнемоники позволяют нам вставлять на страницы специальный символы. Например, знак охраны авторского права © (©). Список мнемоников можно посмотреть [по ссылке](#).

Используя мнемоники навестайте следующий текст.

© All rights reserved

Параграф § 3

Тег `<HTML>` основа всего документа

Домашнее задание

Просмотреть теги HTML на внешнем ресурсе <https://webref.ru/html>. Обратите внимание что многие теги уже не используются и будут удалены.

Также очень полезными являются ресурсы <http://htmlbook.ru/> (он также содержит информацию по CSS) и <https://caniuse.com/>, который подскажет какие браузеры поддерживают интересующие вас теги (или CSS свойства) и с какой версии.

Домашнее задание №2

Самостоятельно изучить (и попробовать в коде) следующие семантические теги.

Обозначить ошибку в тексте можно тегом *del* (по умолчанию он будет перечеркнут), а вставленный текст — тегом *ins* (обозначается подчеркиванием). Несмотря на то, что это строчные теги (не разрывают строки), внутри них допускается размещение блочных элементов.

```
<del><p>Неверная информация</p></del>
<ins><p>Новая информация</p></ins>
```

Есть также:

```
<dfn>термин, впервые определяемый в документе</dfn>
<code>программный код или его фрагмент</code>
<samp>результат вывода компьютерной программы или скрипта</samp>
<kbd>название клавиш или набираемого на клавиатуре текста</kbd>
<var>переменная (математическая или компьютерной программы)</var>
<cite>цитата или сноска на другой материал</cite>
```

Контактную информацию предлагается размещать в блочном элементе *address*, содержимое которого также отображается наклонным начертанием.

Короткие цитаты можно выделить строчным тегом *q* (текст отобразится в двойных кавычках). Для длинных предназначен блочный элемент *blockquote*, который отображается с некоторым отступом со всех сторон. У тегов *q* и *blockquote* есть необязательный атрибут *cite*, значением которого обычно указывают ссылку (URL) на цитируемый источник. Большинство браузеров этот атрибут игнорируют, однако поисковые системы могут учитывать его при анализе страницы.

Для аббревиатур и акронимов предназначен тег *abbr*. Обычно его снабжают всплывающей подсказкой с помощью глобального атрибута *title*, в котором указывают расшифрованное значение.

Тег *mark* предназначен для выделения фрагментов текста в контексте действий пользователя, даже если они не выделяются в оригинале. Например, им можно обозначать слова, соответствующие поисковому запросу или ошибки во вводимых пользователем данных. Оформление содержимого элементов определяется в таблицах стилей.

Некоторые из перечисленных тегов на сегодняшний день рекомендуется заменять CSS свойствами (например, теги *del* и *ins*), однако есть ситуации, когда лучше применять теги.