[](https://commons.wikimedia.org/wiki/File:HTML5_logo_and_wordmark.svg?uselang=ru)

**HyperText Markup Language**

[HTML 1](#_Toc503967643)

[1 Что такое HTML? 1](#_Toc503967644)

[2 Где создать HTML? 2](#_Toc503967645)

[3 С чего начать? 3](#_Toc503967646)

[4 Заголовки 5](#_Toc503967647)

[5 Форматирование текста 7](#_Toc503967648)

[6 Специальные символы 9](#_Toc503967649)

[7 Списки 10](#_Toc503967650)

[8 Картинки 12](#_Toc503967651)

[9 Ссылки 13](#_Toc503967652)

[10 Таблицы 14](#_Toc503967653)

[Ресурсы 17](#_Toc503967654)

# HTML

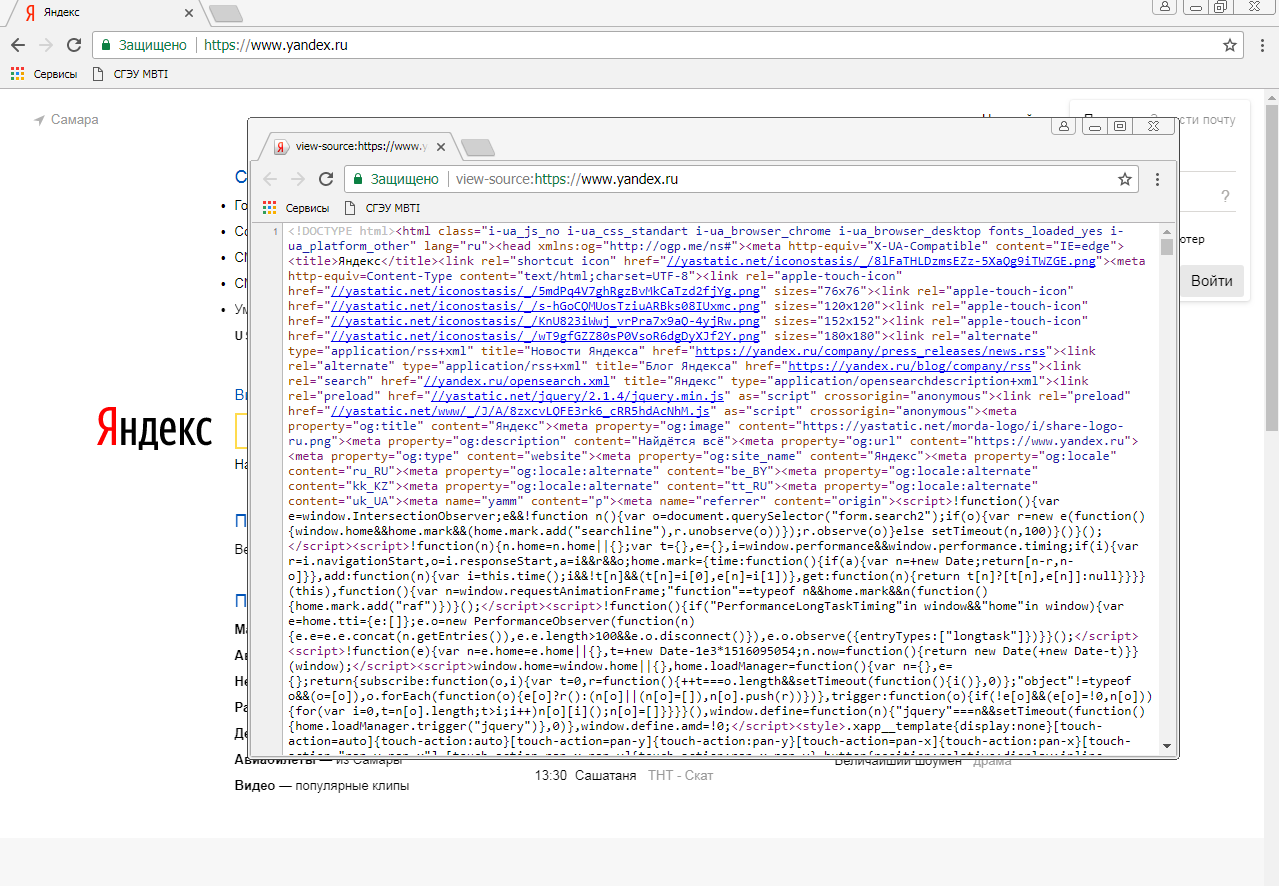
## Что такое HTML?

*HyperText Markup Language* – *язык гипертекстовой разметки* документов. Более точного определения подобрать нельзя. Гипертекст – это такой текст, который содержит гиперссылки (т.е. ссылки, на которые можно нажать и перейти куда-то) на другой гипертекст или ресурсы (рисунки, видео, файлы и т.п.).

HTML – это описание web-страницы. Он показывает, где будет заголовок, где обычный текст, где список, где рисунок, где гиперссылка.

Откройте любую web-страницу, вызовите всплывающее меню (правый клик по странице) и выберите пункт «Просмотр кода страницы» (в разных браузерах называется немного по-разному: «Просмотр HTML-кода», «Исходный код» и т.п.). Или просто нажмите Ctrl+U.

Вот, например, как выглядит исходный код страницы Яндекса (он, конечно, довольно большой и сложный):



В HTML используются теги (читается «тэг») – текст в угловых скобках <…> - он и показывает, где что находится.

## **Где создать HTML?**

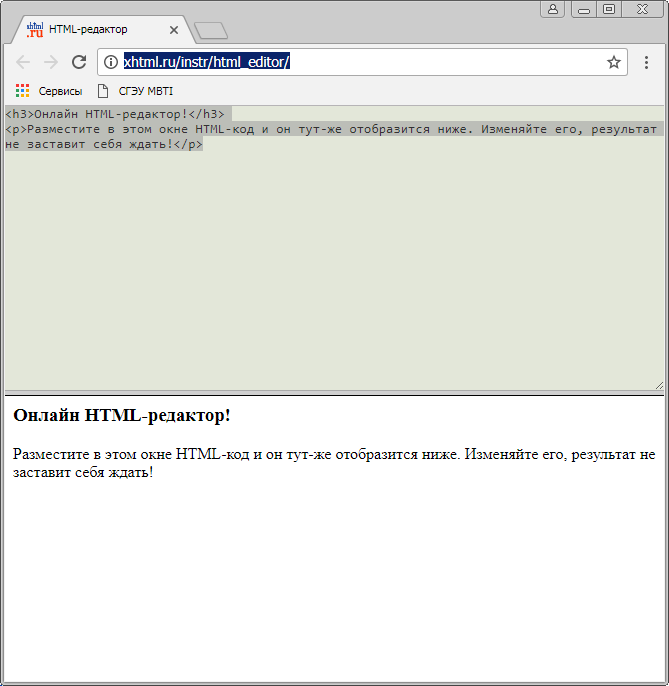
Самый простой вариант – в обычном **Блокноте**. Но не самый удобный. Лучше использовать более продвинутые **редакторы**, которые подсвечивают синтаксис, умеют дописывать начатое (автозавершение) и содержат другие полезные вещи.

Популярные бесплатные редакторы (они нам еще для JavaScript пригодятся):

* **Notepad++:** https://notepad-plus-plus.org/download/
* **Atom:** <https://atom.io/>
* **Komodo Edit:** <https://www.activestate.com/komodo-ide/downloads/edit>

Выбирайте любой, лучше перепробовать все и найти наиболее удобный.

Можно использовать **онлайн-редактор HTML**, но сохранить результат в большинстве случаев не получится. Да и видеть свою страничку вы будете только в пол-окна. Пример: <http://xhtml.ru/instr/html_editor/>



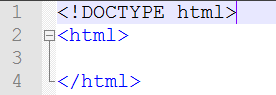
Примеры ниже созданы в Notepad++.

## С чего начать?

Итак, мы обзавелись редактором кода. Что в нем писать? Давайте по-порядку.

Для начала надо показать, что мы собираемся работать именно с HTML. Для этого

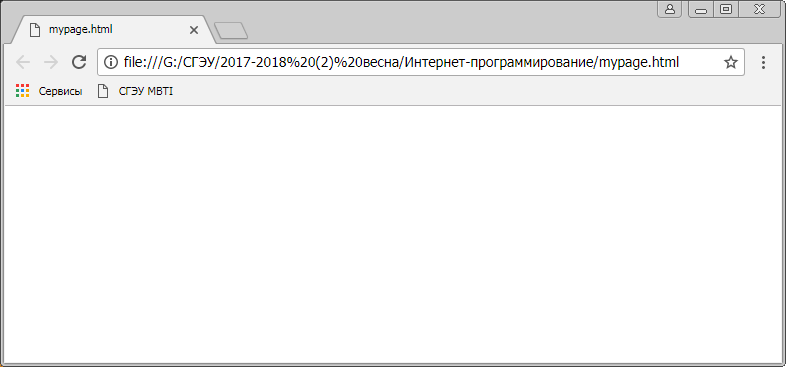
1. в начале документа пишется: <!DOCTYPE html>
2. весь текст документа обрамляется в пару тегов <HTML> </HTML>



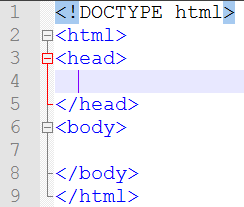
HTML не чувствителен к регистру, поэтому без разницы, какие буквы использовать – заглавные или маленькие. Но лучше для себя сразу определиться и писать везде одинаково, чтобы не было каши. Большинство тегов живут **парами**: открывающий тег <html> и его брат-близнец, только с косой чертой в начале </html>.

Сохраните полученный файл с расширением **.html**, например, "mypage.html". Следите, чтобы это было именно "mypage.html", а не "mypage.html.txt"! Если у вас до этого не сработала подсветка синтаксиса, то после сохранения в правильном формате должна появиться.

Пока что наша страница абсолютно пуста, но при этом корректна – браузер ее откроет без ругани и покажет чистый белый фон.

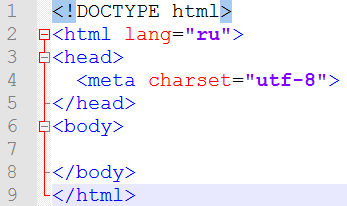


Теперь надо разделить html-документ на две части: голову (head) и тело (body). Разумеется, тоже с помощью пар тегов. Голова должна быть выше тела, согласны?



Голова head содержит **метаданные**, «мысли» о web-странице, которые напрямую на ней часто не отображаются, но очень важны. Внутри body пишется все, что нужно показать на странице.

Например, укажем, что наша страничка на русском языке в кодировке Unicode:



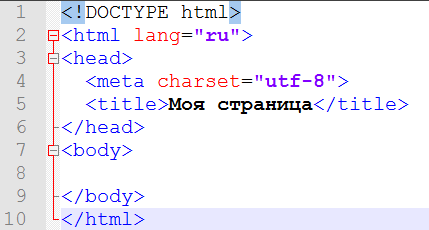
Здесь мы добавили к тегам html и meta **атрибуты** lang и charset. Notepad++ подсвечивает атрибуты красным. Кстати, meta – непарный тег, у него нет закрывающего близнеца.

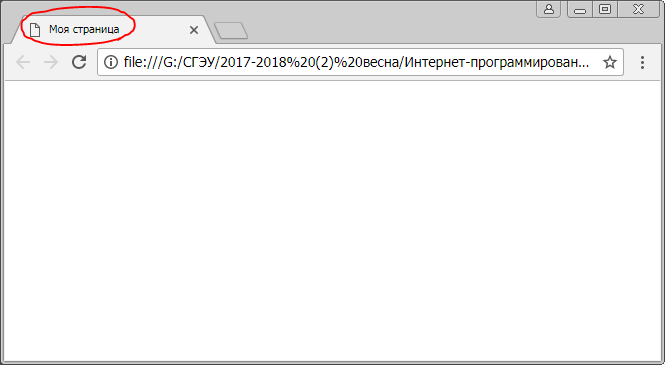
## Заголовки

Давайте наконец что-то покажем на странице. Начать логичнее всего с **заголовка**, например, «Моя страница».

Заголовок по-английски **header** (заголовок главы, параграфа, раздела) или **title** (название).

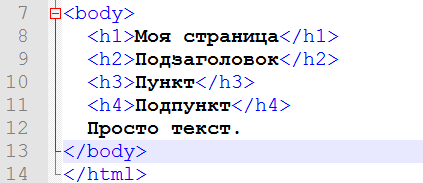
Тег title пишется внутри head и отвечает за название, которое пишется на вкладке браузера и которое показывает поисковик в результатах поиска.

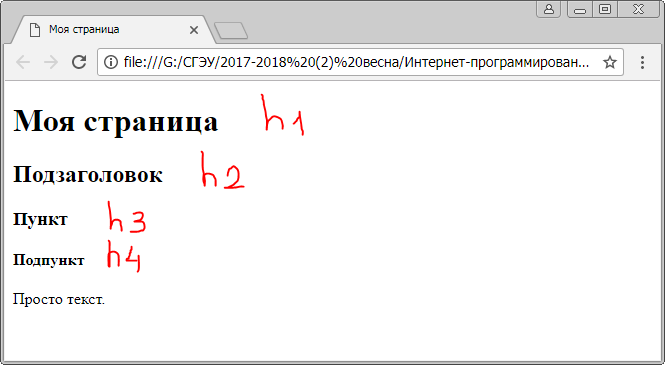




Т.е. заголовком будет то, что мы напишем между парами тегов <title> </title>.

Заголовки **header** – многоуровневые (главный заголовок – 1ый уровень, подзаголовок – 2ой уровень, и т.д.). Теги заголовков на странице, соответственно, обозначаются h1, h2, h3 и т.д. Они парные.





Заголовки по умолчанию показываются более крупным полужирным шрифтом. Перенастроить их оформление можно через CSS (см. ниже).

Обратите внимание, текст безо всяких тегов вывелся «как есть» стандартным шрифтом. Поэтому даже если «наваять» с тегами – хоть что-то на странице да отобразится.

## Форматирование текста

Обычно текст на странице состоит не из одной строчки. Однако символы перехода на новую строку, используемые для разбиения на абзацы обычного текста (Enter), в коде HTML воспринимаются как обычные пробелы. Поэтому для создания *абзацев* в HTML-страницах используется парный тэг <p> (paragraph).

У тэга <p> есть параметр align, который позволяет выровнять абзац по левому, правому краю и по центру:

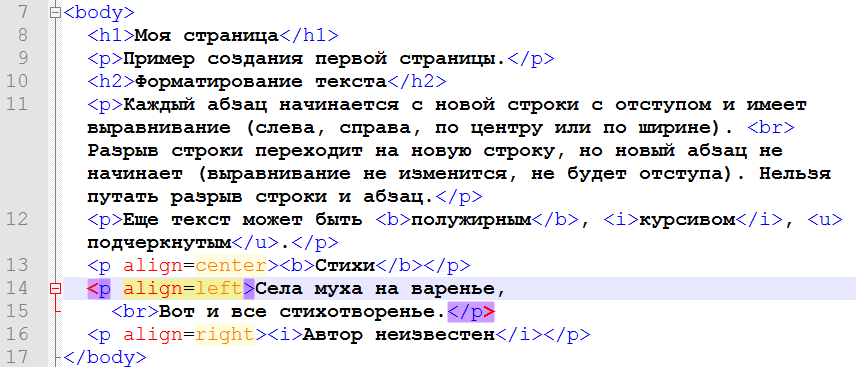
* <p align=center>текст</p> – выравнивает текст по центру;
* <p align=left>текст</p> – выравнивает по левому краю;
* <p align=right>текст</p> –выравнивает по правому краю;
* <p align=justify>текст</p> – выравнивает текст по ширине.

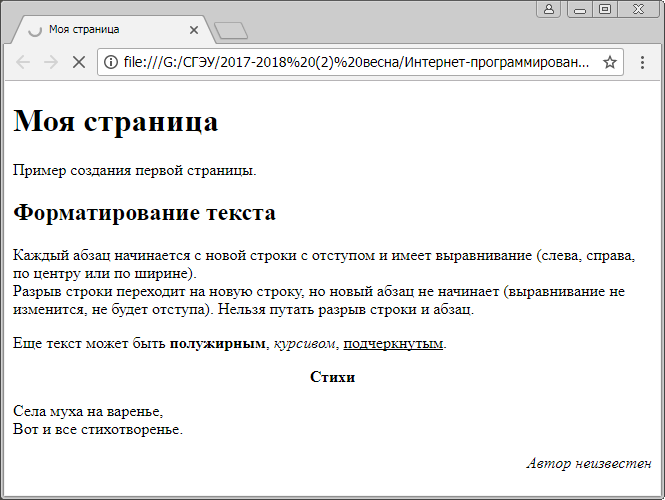
По умолчанию (если не указывать align) абзац выравнивается по левому краю.

Для того, чтобы осуществить перенос текста на новую строку без создания нового абзаца, можно использовать непарный тэг <br> (break, *разрыв строки*).

Для оформления текста используют парные теги:

* <I> (italic) – *курсивное* начертание;
* <B> (bold) – **жирное** начертание;
* <U> (underlined) – подчеркивание.





## Специальные символы

Поскольку символы < и > используются для тэгов, их не получится отобразить в тексте страницы. Существуют и другие зарезервированные символы, а также символы, которых нет на клавиатуре, например копирайт ©, длинное тире —, плюс-минус ±.

Для них используются специальные последовательности (escape-последовательности), начинающиеся с символа & (амперсанд) и заканчивающиеся точкой с запятой (;).

Основные спецсимволы:

&lt;**–** знак меньше <;

&gt; – знак больше >;

&amp; – амперсанд &;

&nbsp; – неразрывный пробел (по нему нельзя переносить текст на другую строку, например Фамилия&nbsp;И.О.);

&shy; – мягкий перенос (отображается, только когда слово переносится);

&quot;– двойные кавычки “;

&plusmn; – плюс-минус ±;

&copy; – знак копирайта ©;

&trade; **–** товарный знак ™;

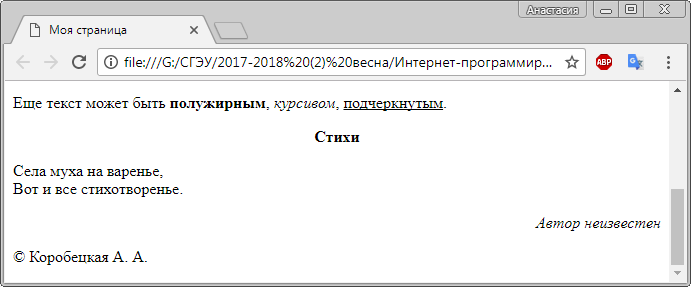
&deg; **–** знак градуса °.

Например, текст &lt;тег&gt; &amp;текст отобразится в браузере как <тег>&текст.

В принципе, любой символ можно представить в виде escape-последовательности. В Интернте можно найти таблицы, содержащие коды escape-последовательностей для всех символов.

Добавим копирайт в нижнюю часть страницы (укажите свою фамилию и группу).





## Списки

Списки помогают структурировать текст и облегчают восприятие информации.

Пункты списков оформляются с помощью парного тега <li>, причем каждый элемент списка заключается в данный тег.

Пример:

<li>строка 1</li>

<li>строка 2</li>

Для того, чтобы создать маркированный список, можно воспользоваться парным тегом <ul> (unordered list):

<**u**l>

<li>строка 1</li>

<li>строка 2</li>

</**u**l>

Нумерованные списки создаются с помощью тега <ol> (ordered list).

<**o**l>

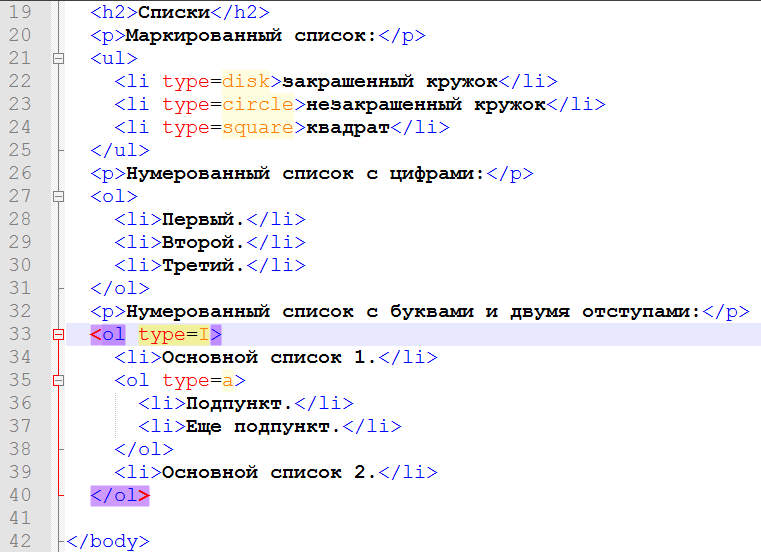
<li>строка 1</li>

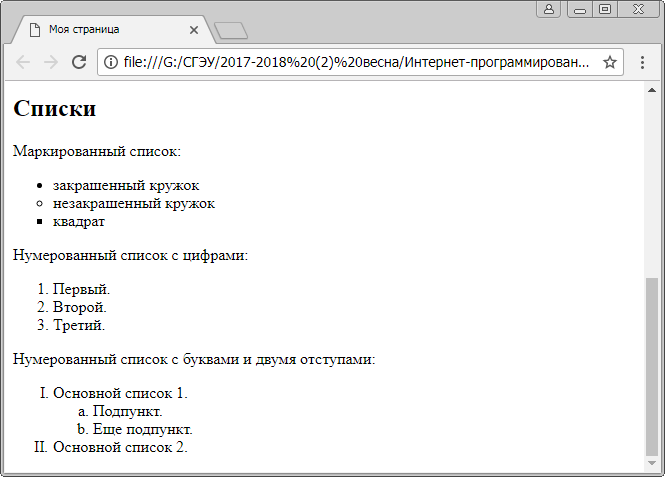
<li>строка 2</li>

</**o**l>

При этом количество тегов <ul> или <ol> соответствует числу отступов вправо для каждого пункта списков.

Параметр type, который может быть задан для любого из тэгов <ul>, <ol> и <li>, отвечает за стиль маркировки или нумерации списка и может принимать значения disk, circle, square для маркированного списка и 1, a, A, i, I для нумерованного списка.





## Картинки

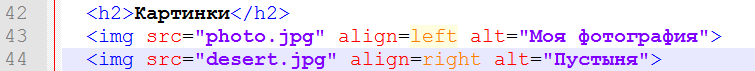
Для вставки картинки в документ используется тег <img src= ”название картинки”>. В качестве названия картинки можно использовать:

* название файла, если картинка и файл лежат в одной папке, например picture.gif;
* относительный путь к файлу, если картинка лежит в подпапке (например, “files/my\_pic.bmp”);
* полный путь к файлу, если картинка находится в другом файле или на другом сайте (например, “C:/My documents/1.jpg” или “http://mysite.ru/my\_pic.png”).

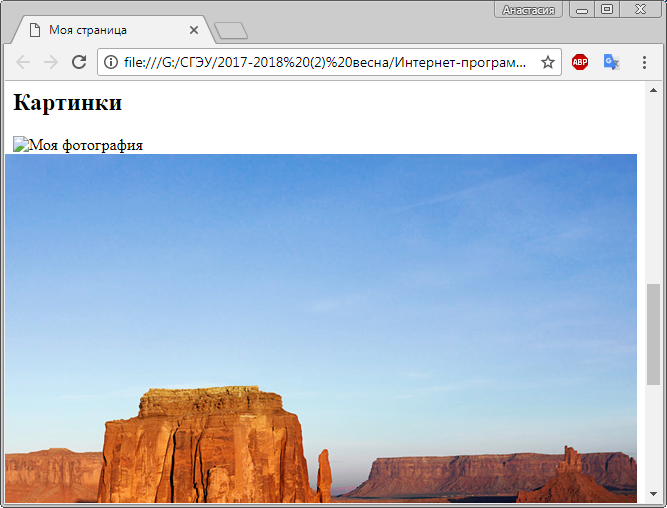
Желательно размещать все ресурсы (картинки и другие файлы) сайта в одной папке и использовать относительные пути.

Тег <img> не имеет закрывающего тега и обладает следующими основными атрибутами:

* align отвечает за расположение картинки и может принимать значения left (слева), right (справа), center (по центру), bottom (текст располагается ниже картинки), middle (текст располагается посередине картинки), top (текст располагается над картинкой);
* alt – краткое описание картинки, отображается при наведении курсора на картинку, либо вместо нее, пока идет загрузка;
* width – ширина картинки (в пикселях);
* height – высота самой картинки (в пикселях);
* border – толщина рамки вокруг картинки (в пикселях).



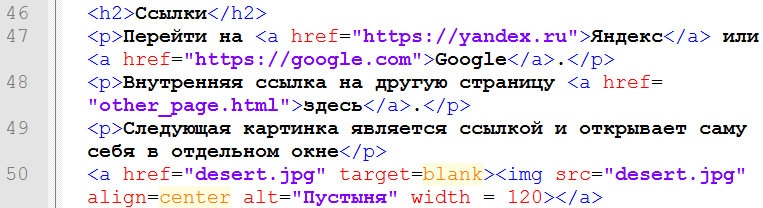
Если файл изображения отсутствует, то вместо него отобразится подсказка alt.

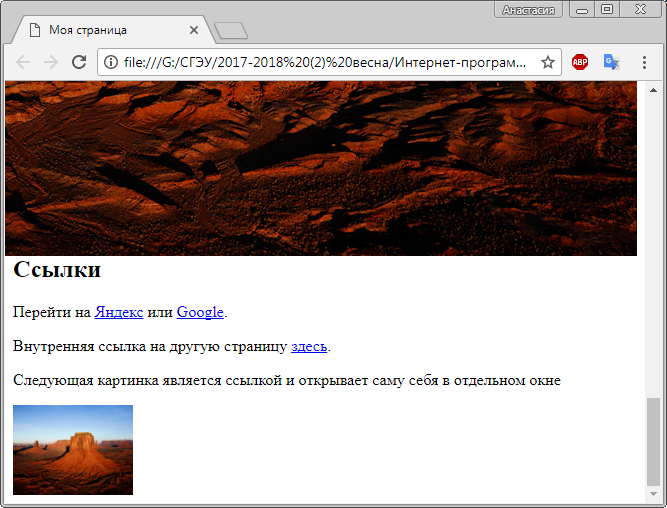


## Гиперссылки

Гиперссылкой может быть как слово или фраза, так и объект (например, картинка).

Ссылкой является любой фрагмент html внутри парного тега <a>. Куда ведет ссылка, указывается в атрибуте href. Атрибут target отвечает за то, в каком окне откроется страница. Если он принимает значение blank, то ссылка откроется в новом окне или вкладке браузера.





Самостоятельно создайте страницу other\_page.html (с любым содержимым), на которую будет вести ссылка.

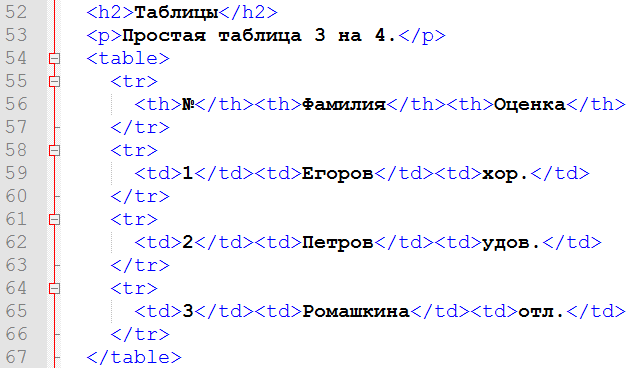
## Таблицы

Таблицы создаются с помощью следующих тегов:

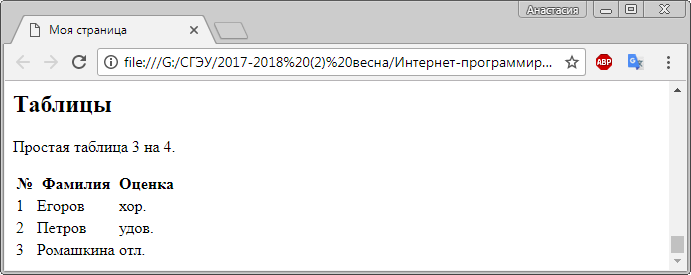
* <table> – парный тег, определяющий начало и конец таблицы;
* <tr> (table row) – парный тег, задающий строку таблицы;
* <td> – парный тег, определяющий ячейку внутри строки;
* <th> (table header) – задается для ячеек первой строки таблицы (с заголовками стобцов).

Например, создадим простую таблицу 3 на 4:

|  |  |  |
| --- | --- | --- |
| № | Фамилия | Оценка |
| 1 | Егоров | хор. |
| 2 | Петров | удов. |
| 3 | Ромашкина | отл. |

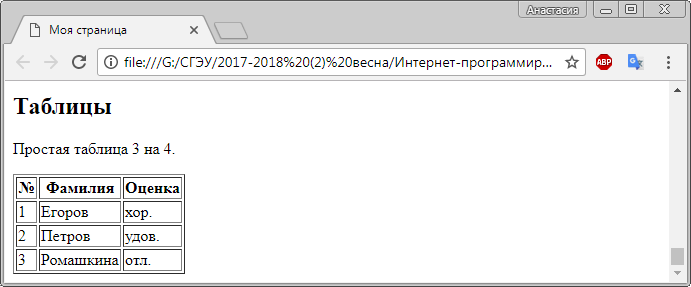


По умолчанию границы ячеек не отображаются.



Чтобы отобразить границы, нужно добавить тегу <table> атрибут border.

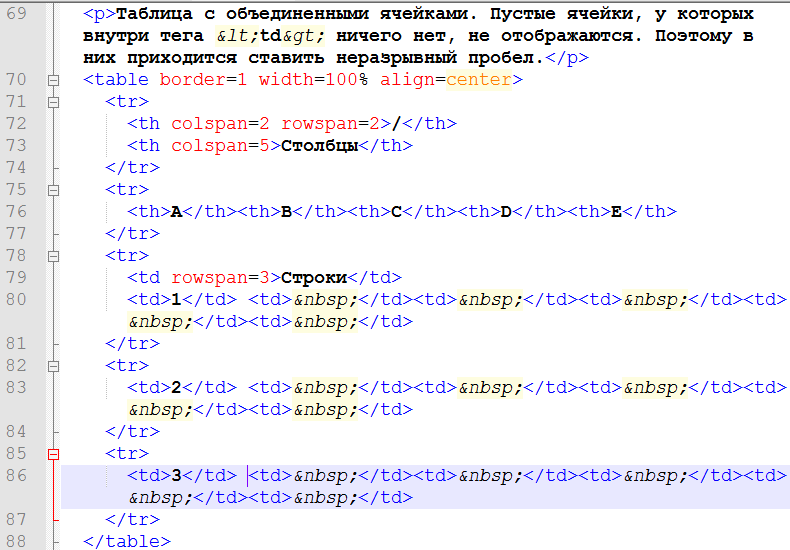


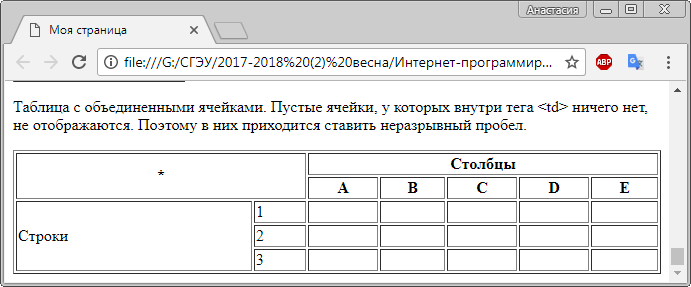


Для настройки ширины таблицы (всей или отдельных строк и столбцов) используются атрибуты height (высота) и width (ширина).

Чтобы создать объединенные ячейки, нужно использовать атрибуты colspan (сколько столбцов займет объединенная ячейка) и rowspan (cколько строк займет объединенная ячейка).

Рассмотрим более сложный пример.





# Ресурсы

<http://htmlbook.ru/>

<https://codebra.ru/>

<https://www.youtube.com/playlist?list=PL0lO_mIqDDFUpe6yMyXAlcrfT6AO0KW1a>

<https://www.youtube.com/watch?v=8MEJdXJvCUY&index=2&list=PLVfMKQXDAhGWUc3NEYT2De304hn290FdI>