

Статическая верстка

Модуль 1 / 5

Технологии

HTML



CSS



JS



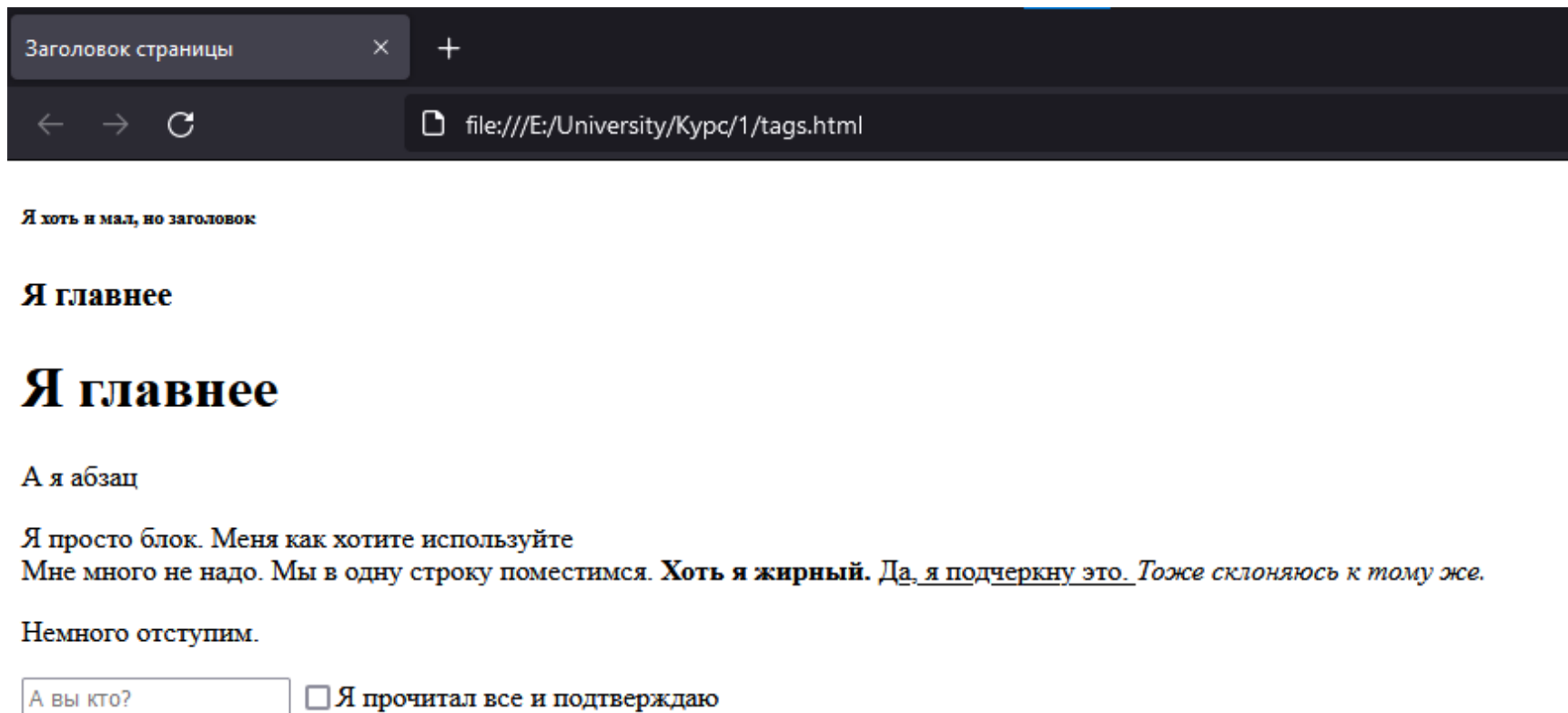
Структура HTML документа

- **HTML** (от англ. HyperText Markup Language — «язык гипертекстовой разметки») — стандартизированный язык разметки **документов** для просмотра веб-страниц в браузере.
- Документы имеют формат .html, открываются через любой браузер. Писать их можно хоть в Блокноте. Мы цивилизованные, будем использовать Sublime Text, вот: <https://www.sublimetext.com/3>
- Сейчас посмотрим, что примерно из себя они представляют.

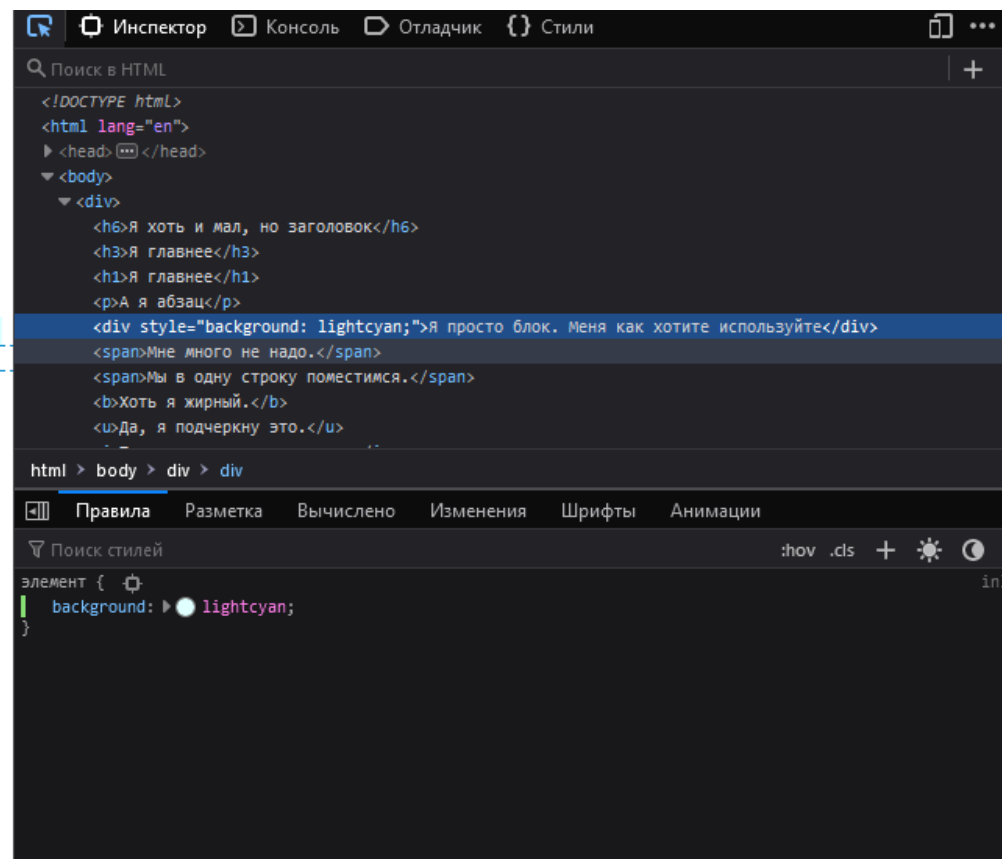
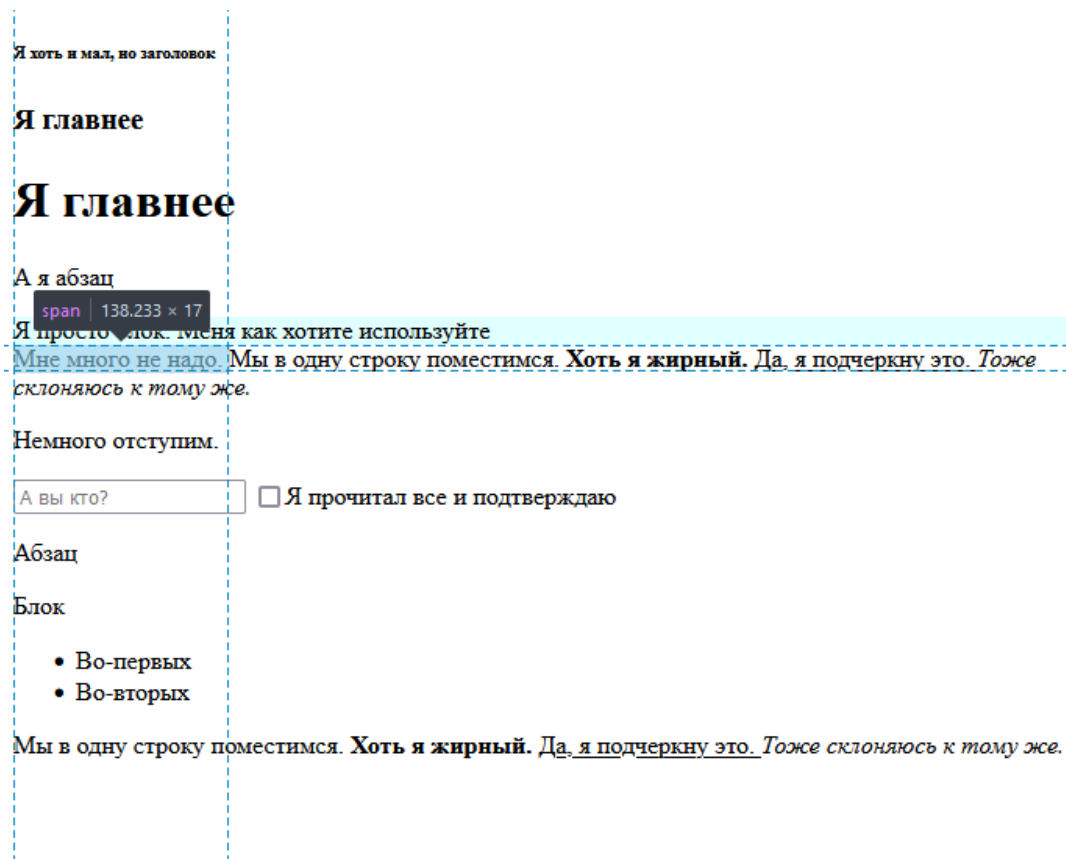
Это в редакторе

```
tags.html x
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Заголовок страницы</title>
8 </head>
9 <body>
10   <div>
11     <h6>Я хоть и мал, но заголовок</h6>
12     <h3>Я главнее</h3>
13     <h1>Я главнее</h1>
14     <p>А я абзац</p>
15     <div>Я просто блок. Меня как хотите используйте</div>
16     <span>Мне много не надо. </span>
17     <span>Мы в одну строку поместимся. </span>
18     <b>Хоть я жирный. </b>
19     <u>Да, я подчеркну это. </u>
20     <i>Тоже склоняюсь к тому же. </i>
21     <br>
22     <p>Немного отступим.</p>
23     <input type="text" placeholder="А вы кто?">
24     <label>
25       <input type="checkbox">Я прочитал все и подтверждаю
26     </label>
27   </div>
28 </body>
29 </html>
```

А это в браузере



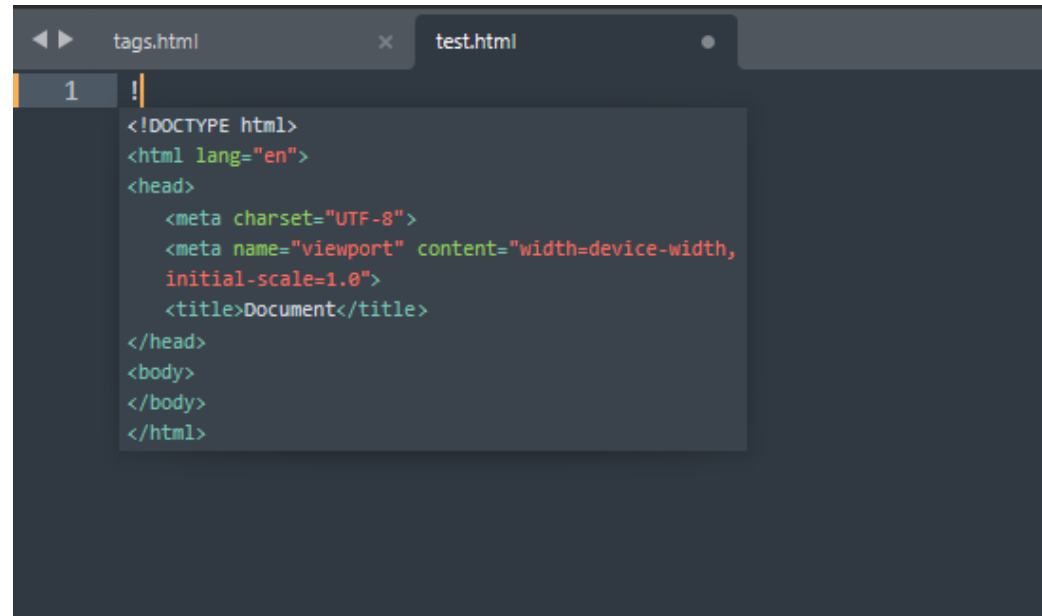
Инструменты разработчика в браузере могут быть полезны.
Вызов F12, либо ПКМ по странице – Просмотреть код.



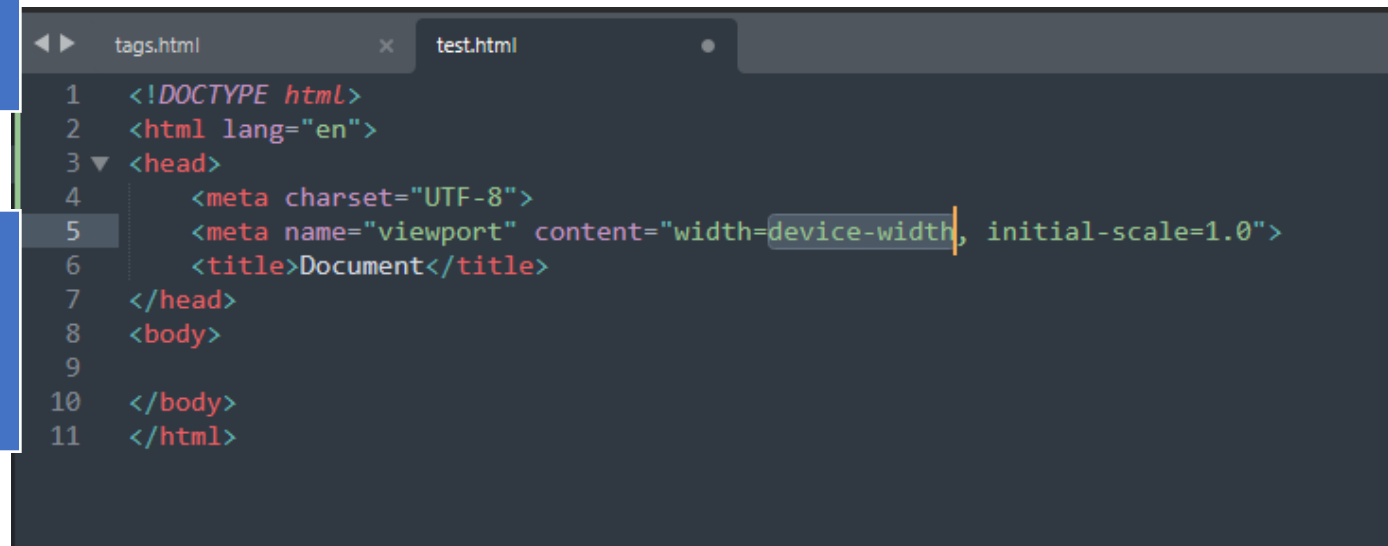
Совет: используем плагин Emmet (в VSCode, например он из коробки) для быстрой верстки.

Например, если ввести «!» и нажать Tab, получите готовую структуру HTML.

Подробнее, как всегда, обратитесь, в Google.



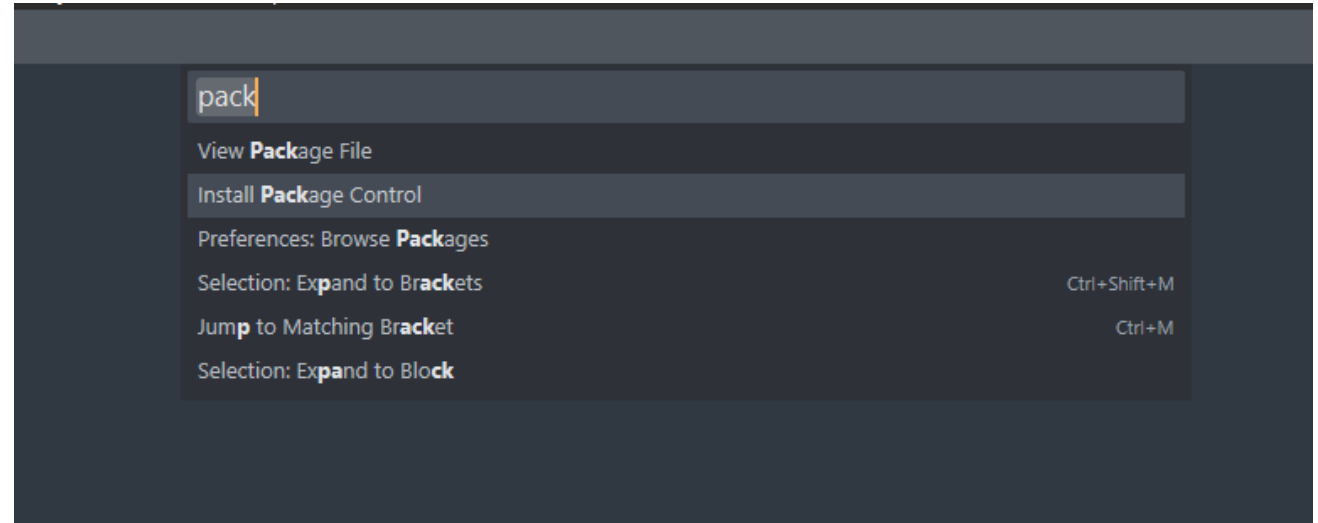
```
1  !  
  <!DOCTYPE html>  
  <html lang="en">  
  <head>  
    <meta charset="UTF-8">  
    <meta name="viewport" content="width=device-width, initial-scale=1.0">  
    <title>Document</title>  
  </head>  
  <body>  
  </body>  
</html>
```



```
1  <!DOCTYPE html>  
2  <html lang="en">  
3  <head>  
4    <meta charset="UTF-8">  
5    <meta name="viewport" content="width=device-width, initial-scale=1.0">  
6    <title>Document</title>  
7  </head>  
8  <body>  
9  
10 </body>  
11 </html>
```

Установка Emmet в Sublime Text

- Ctrl+Shift+P – Install Package Control
- Emmet будет сразу установлен. Если чуда не случилось, снова Ctrl+Shift+P – Install Packages, ищем Emmet и ставим.



Теги

- Простыми словами, теги – это составные части тела документа, которые читает и разбирает браузер при отображении. Они могут вкладываться друг в друга и образуют дерево элементов.
- Теги выполняют разные функции. Есть теги, являющиеся просто контейнерами для текста и другого содержимого, например, абзацы. Есть теги изображений, таблиц, видео, ссылок, кнопок и так далее. Отдельная история с формами: её элементы – специальные для ввода значений (текстовые поля, выпадающие списки, галочки и многое другое. Все обязательно подробно рассмотрим.
- Тегов довольно много, но часто используются не все. Разберем, какие стоит знать для быстрого старта, основные атрибуты и особенности.

```
<form action="#" method="GET" class="example">  
  <h1 class="heading">Начнем</h1>  
</form>
```

```
<!-- Парные теги -->
<section>
  <table>
    <tr>
      <td>Раз ячейка</td>
      <td>Два ячейка</td>
    </tr>
  </table>
</section>

<!-- Одиночные теги -->
<input type="text">

```

- Парные имеют закрывающий тег, в них может быть вложено какое-то содержимое. Это абзацы, блоки, таблицы и другие.
- Одиночные не имеют закрывающего тега, только атрибуты. Например, картинка.

- Блочные – по умолчанию занимают всю ширину, строчные – автоматически выстраиваются в одну строчку. Картинка, текстовое поле, span, b, u, i – строчные теги.
- Конечно, впоследствии можно будет как угодно стилизовать элементы, но сейчас смотрим на их обычное поведение.

```
<!-- Блочные теги -->
<p>Абзац</p>
<div>Блок</div>
<form action="#"></form>
<ul>
  <li>Во-первых</li>
  <li>Во-вторых</li>
</ul>

<!-- Строчные теги -->
<span>Мы в одну строку поместимся. </span>
<b>Хоть я жирный. </b>
<u>Да, я подчеркну это. </u>
<i>Тоже склоняюсь к тому же. </i>
```

Абзац

div | 691 × 18

Блок

- Во-первых
- Во-вторых

Мы в одну строку поместимся. **Хоть я жирный.** Да, я подчеркну это. *Тоже склоняюсь к тому же.*

- `<doctype>` - это просто есть, не трогаем, и оно нас не трогает. Это просто строчка, которая говорит: «Браузер, будь ты хром или лиса, даже прошу, если explorer, вот все, что дальше – веб-страница, прочитай её как надо». Это все равно, что формат файла.
- Все, что находится внутри `<body></body>`, будет показано на странице браузера.
- Но внутри тега `<html>` еще есть секция `<head>`.
- Вот там разная нужная служебная информация, которая явно не видна. Здесь устанавливается кодировка, подключаются стили, прописывается, как будет адаптироваться страница. SEO-штуки, кстати, тоже здесь будут.
- Title – это то, как будет называться вкладка в браузере
- Но насчет этого можно не особо заморачиваться, так как при создании шаблонной страницы через Emmet минимально необходимое уже будет.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Заголовок страницы</title>
</head>
```

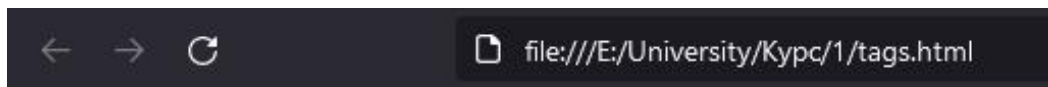
- Это примеры распространенных блочных тегов. На самом деле большинство из них работают как обычный `div`, но названия придумали для удобства и семантической верстки.
- Абзац ***p*** по умолчанию имеет небольшой отступ снизу. Заголовки идут по возрастанию от ***h6*** до ***h1***.

```
<header>
  <nav></nav>
</header>
<aside></aside>
<section>
  <article>
    <h1></h1>
    <div></div>
    <p></p>
  </article>
</section>
<footer></footer>
```

- Ссылки – кликабельные элементы страницы, ведущие на определенный ресурс. Это может быть как абсолютная ссылка на полный URL, так и относительная на какой-то файл. Советую почитать про абсолютные ссылки.
- Ссылка с таким же успехом может вести на любой файл. Браузер либо откроет его, если умеет (точно умеет картинки, pdf, аудио, видео, текстовые файлы). Если не умеет, то файл автоматически скачивается.
- А вот если указать *атрибут **download***, то скачается в любом случае. При этом, если атрибуту еще и передать значение, то скачается с этим именем.

```
<a href="https://tou.edu.kz">Ссылка на сайт</a>  
<a href="https://tou.edu.kz" target="_blank">Ссылка откроется в другой вкладке</a>  
<a href="../lessons/example.html">Относительная ссылка</a>  
<a href="images/image.jpg">Ссылка на картинку</a>  
<a href="images/image.jpg" download>Скачает картинку</a>  
<a href="images/image.jpg" download="скачанное.jpg">Скачает картинку с именем</a>
```

- Тег изображения имеет атрибут `src`. В него передается путь до изображения. Так же может быть как относительным, так и абсолютным (например, ссылаться на сторонний ресурс). Может вообще принимать base64 строку. Атрибут `alt` - это текст, который покажется в браузере, если изображение не удастся загрузить. Полезен для SEO.
- Можно еще почитать про `srcset`, нужно, чтобы указать несколько источников для изображения. Например, если нужно загружать разные картинки для разных разрешений экрана, или если формат не поддерживается.



```
FOLDERS
▼ 1
  image.jpg
  <> tags.html
  <> test.html

tags.html
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta http-equiv="X-UA-Compatible" content=">
6      <meta name="viewport" content="width=device-width, initial-scale=1">
7      <title>Заголовок страницы</title>
8  </head>
9  <body>
10     <div class="image-wrapper">
11         
12     </div>
13 </body>
14 </html>
```

- Уже интереснее с тегами форм. На скриншоте то, как выглядят некоторые элементы в браузере. Думаю, большинство знакомы.

Ваше имя

Ваш email

Пароль

Зарплата

Дата рождения

Во сколько спать?

Любимый цвет

Как настроение

Ваш пол ☐ М ☐ Ж

Город

О себе

☐ Я принимаю и уважаю себя

- Основной элемент – тег **form**. Все элементы помещаются внутрь него. Рассмотрим атрибуты.
- Action – адрес, куда будет отправлена форма. Обычно это какой-то скрипт. В примере форма отправит в поиск Google.
- Method – GET или POST. Вообще еще есть HTTP-методы HEAD, PUT, PATCH, DELETE, но к ним вернемся в модуле про серверную разработку. GET – при отправке данные формы увидим в строке поиска браузера. Чаще всего используется для поиска. Пример использования метода POST – отправка данных на сохранение, загрузка файлов и т.д.
- Если атрибут method не указан, то по умолчанию GET.

```
<form action="https://www.google.com/search" method="GET">  
  <input type="text" name="q">  
  <!-- И дальше элементы формы... -->  
</form>
```

- Теги элементов форм:
- Type – у тегов *input* указывает, что это за тип ввода (например, text, email, password, search, color, checkbox, radio, range)
- Name – имя, обязательное поле, чтобы сервер понял, что это за данные.
- Placeholder – у *input type email, text, search, password* текст-подложка, подсказка
- Id – для связи фокуса с элементом label
- Autocomplete – при установке в false отключает автозаполнение. Также этот атрибут можно прописать для всей формы
- Step – опционально шаг
- Required – обязательное поле, если не заполнено, форма не отправится (если не задано novalidate)
- Disabled – отключенное. При это данные с него не отправятся на сервер
- Readonly – недоступно для редактирования, но данные передадутся

```
<p>
  <label for="name">Ваше имя</label>
  <input type="text" name="name" id="name" placeholder="Ваше имя" required>
</p>
<p>
  <label for="email">Ваш email</label>
  <input type="email" name="email" id="email" placeholder="example@gmail.com" required>
</p>
<p>
  <label for="password">Пароль</label>
  <input type="password" name="password" id="password" placeholder="Пароль">
</p>
<p>
  <label for="salary">Зарплата</label>
  <input type="number" name="salary" id="salary" step="10" value="200000">
</p>
<p>
  <label for="born">Дата рождения</label>
  <input type="date" name="born" id="born">
</p>
<p>
  <label for="time">Во сколько спать?</label>
  <input type="time" name="time" id="time">
</p>
<p>
  <label for="color">Любимый цвет</label>
  <input type="color" name="color" id="color">
</p>
<p>
  <label for="mind">Как настроение</label>
  <input type="range" name="mind" id="mind">
</p>
```

- Здесь также представлен тег *select* для выпадающего списка, многострочное поле ввода *textarea* и кнопки. Атрибут *type* тега *button* влияет на ее функцию: *submit* – отправить форму, *reset* – сбросить, *button* – просто кнопка. По умолчанию, если не указано явно, то “*submit*”.
- Также форма отправляется, если нажать Enter, пока фокус в одном из элементов.

```
<p>  
  <label>Ваш пол</label>  
  <label><input type="radio" name="gender" value="m">М</label>  
  <label><input type="radio" name="gender" value="f">Ж</label>  
</p>  
<p>  
  <label for="city">Город</label>  
  <select name="city" id="city">  
    <option value="">Не выбрано</option>  
    <option>Павлодар</option>  
    <option>Нур-Султан</option>  
    <option>Семей</option>  
  </select>  
</p>  
<p>  
  <label for="about">О себе</label>  
  <br>  
  <textarea name="about" id="about" rows="10" placeholder="Расскажи, а?"></textarea>  
</p>  
<p>  
  <label><input type="checkbox" name="agree" value="1">Я принимаю и уважаю себя</label>  
</p>  
<p>  
  <button type="submit">Отправить</button>  
  <button type="reset">Сбросить</button>  
</p>
```

Документация по HTML

<https://html5book.ru/html-tags/>

<http://htmlbook.ru/html>

<https://developer.mozilla.org/ru/docs/Learn/HTML>