



# Лекция 1

Знакомимся с HTML

## **Цель на сегодня:**

- Научиться создавать элементарные веб-страницы.

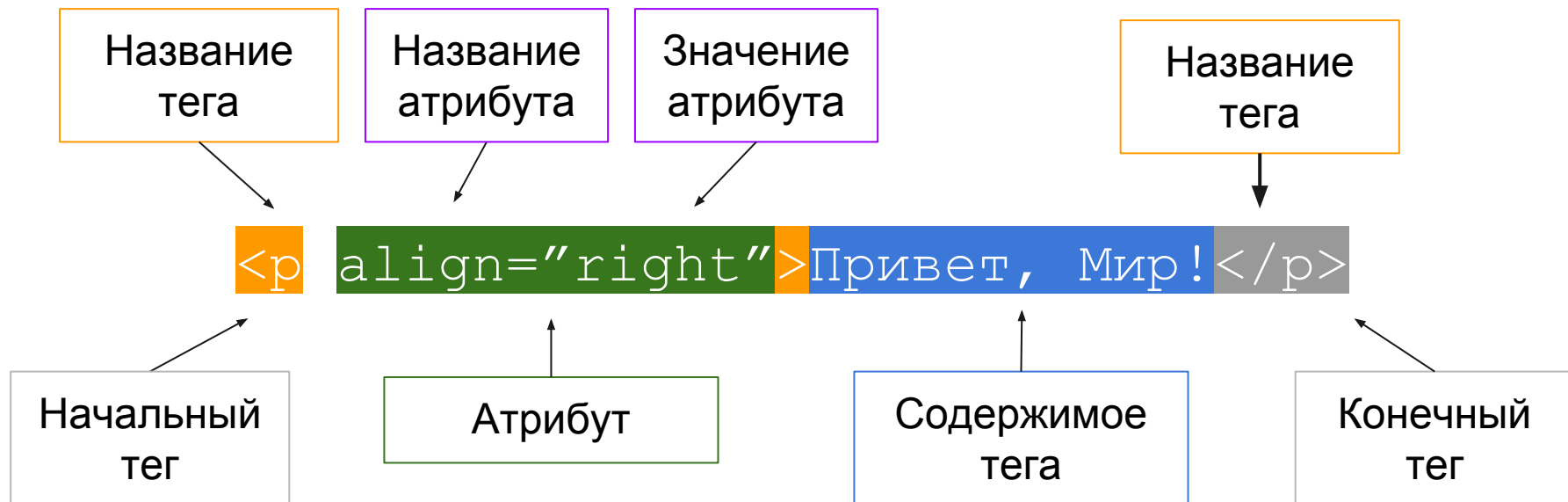
## Самостоятельная работа

В локальном репозитории находим файл **README.md** и на девятой строке пишем такой код:

```
* [Профиль на htmlacademy:] (ссылка-на-профиль) .
```


Сохраняем, пушим в личный репозиторий и создаем pull request.

Файлы в формате **\*.md** – это **Markdown** – популярный и простой язык разметки




## Не все теги закрываются

Некоторые теги (<area>, <base>, **<br>**, <col>, <command>, <embed>, <hr>, **<img>**, **<input>**, <keygen>, **<link>**, **<meta>**, <param>, <source>, <track>, <wbr>) имеют только открывающий (начальный) тег, но не имеют закрывающего (конечного) и содержимого. И это нормально.



```
<!DOCTYPE html>
<html lang="ru">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
</head>
<body>
  <h1>Тут находится контент страницы</h1>
</body>
</html>
```

Создать базовый скелет  
страницы можно написав знак  
“!” и нажав кнопку **Tab**



```
<!-- Говорим браузеру, что используем HTML5 -->
<!DOCTYPE html>
<!-- Сообщаем язык страницы -->
<html lang="ru">
<head>
    <!-- Указываем кодировку страницы -->
    <meta charset="UTF-8">
    <!-- Задаем название вкладки -->
    <title>Занятие #1</title>
</head>
<body>
    <!-- А вот здесь пишем весь основной код -->
</body>
</html>
```

## Комментарии <!-- -->

При помощи конструкции <!-- --> можно указать комментарий. Текст комментария не отображается на странице. Внутри разрешается добавлять другие теги.

Комментарии часто используются разработчиками для описания фрагмента кода для других разработчиков, грязных исправлений, временного отключения кода.

Создать комментарий можно  
сокращением клавиш **Ctrl+/**



## <!DOCTYPE html>

Тег необходим, чтобы браузер понимал, как следует интерпретировать текущую веб-страницу, поскольку есть несколько версий HTML. Кроме того, существует еще XHTML и XML, которые тоже интерпретируются браузером.

Раньше доктайп выглядел бы так:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"  
"http://www.w3.org/TR/html4/loose.dtd">
```

## Самый главный тег <html>

В этом теге находится все содержимое сайта.

Никакие теги не должны выходить за границы <html></html>, так как это приведет к ухудшению SEO-позиции.

## Заголовок сайта – тег <head>

### Какие теги может содержать <head>?

<code>&lt;title&gt;</code>	Отображает название страницы на вкладке
<code>&lt;meta&gt;</code>	Содержит техническую информацию для браузера и поисковика
<code>&lt;link&gt;</code>	Подключает CSS, RSS и favicon
<code>&lt;script&gt;</code>	Подключает JavaScript-файлы

## Тело документа <body>

Этот тег является контейнером для всех остальных тегов страницы.

Он всегда открывается после <head> и содержит весь контент страницы, который увидит пользователь.

## Тип элементов

### Строчные

- Занимают ширины не больше, чем необходимо;
- Являются непосредственной частью другого элемента;
- Привязаны к блоку.

**(a, span, input)**

### Блочные

- Занимают всю доступную ширину окна;
- Начинаются с новой строки;
- Самодостаточны.

**(div, h1, p, form, ul)**

`<h1>Заголовки <h1> - <h6></h1>`

Заголовки разного уровня соответствуют важности контента после них.

Так, тег **<h1>** представляет собой наиболее важный заголовок (первого уровня), а тег **<h6>** служит для обозначения заголовка шестого уровня и является наименее значительным.

# <h1>Сравнение с рефератом</h1>

Хоть заголовки от **h1** до **h6** и отличаются своими размерами, не стоит использовать их бездумно, смотря только на размер. Заголовок каждого уровня представляет собой **вложенность**, которую легко понять на примере плана реферата или курсовой работы.

# <h1> Содержание (план) </h1>

## <h2> Введение </h2>

### <h3> 1. Сущность инфляции </h3>

#### <h4> 1.1. Понятие и классификация инфляции </h4>

#### <h4> 1.2. Измерение инфляции </h4>

### <h3> 2. Инфляционные процессы в России </h3>

#### <h4> 2.1. Этапы развития инфляции </h4>

#### <h4> 2.2. Особенности развития инфляции в Республике Татарстан </h4>

### <h3> 3. Антиинфляционная политика в Российской Федерации </h3>

#### <h4> 3.1. Социально – экономические последствия инфляции </h4>

#### <h4> 3.2. Антиинфляционные меры в РФ и РТ </h4>

## <h2> Заключение </h2>

## <h2> Список литературы </h2>



## Абзац <p>

<p>Используется для хранения абзаца текста. Каждый новый абзац начинается с новой строки.</p>

<p>Между абзацами сохраняется расстояние, указанное в стандартных стилях браузера.</p>

## Строка <span>

Используется для хранения текста, но, в отличие от абзаца, **является строчным** элементом.

Обычно используется для выделения фрагментов в абзаце для последующей их стилизации.

## Гиперссылка <a>

С этих элементов и начинается **гипертекст** – система из текстовых страниц, имеющих перекрестные ссылки.

Позволяет ссылаться с одного документа на другой без использования адресной строки. В зависимости от присутствия атрибутов ***name*** или ***href*** тег <a> устанавливает якорь или ссылку.

Атрибут ***href*** указывает путь, по которому пользователь переходит при нажатии на ссылку. Например:

```
<a href="https://google.com.ua">Google it!</a>
```

## Гиперссылка <a>

Атрибут **href** не может быть пустым – это вызовет ошибку при валидации. Однако, на этапе разработки, мы обычно не знаем настоящие пути, да и в большинстве случаев проект не ограничивается простой статикой, поэтому стоит использовать один из приемов:

**Вариант 1:** использовать заглушку-октоторп ("#");

**Вариант 2:** не указывать атрибут **href** (изменяет поведение ссылки, что чаще всего нежелательно).

## Гиперссылка <a>

Путь к ресурсу может быть абсолютным или относительным.

### Абсолютные пути:

- `https://example.com/hello.html`
- `/hello.html`

### Относительные пути:

- `./pages/hello.html`
- `hello.html`

`<a href =  
http://example.com/page>`  
(absolute)

VS.

`<a href =  
./page>`  
(relative)

## Изображение <img>

Тег предназначен для отображения на странице изображений в одном из графических форматов.

Путь к изображению задаётся через атрибут **src**.

Атрибут **alt** важен для поисковиков и отображает заданный текст тогда, когда картинка не может загрузиться.

Всегда указывай подробный  
**alt** для изображений

## Горизонтальный разделитель <hr>

Создает горизонтальную линию. Не нуждается в закрывающем теге.

## Перевод строки <br>

Устанавливает перевод строки в том месте, где этот тег находится. **Используется в исключительных случаях:** для создания переносов в адресах, стихах, там где нужно добиться точного переноса. Закрывающий тег не нужен.

## Обрати внимание на визуализацию вложенности!

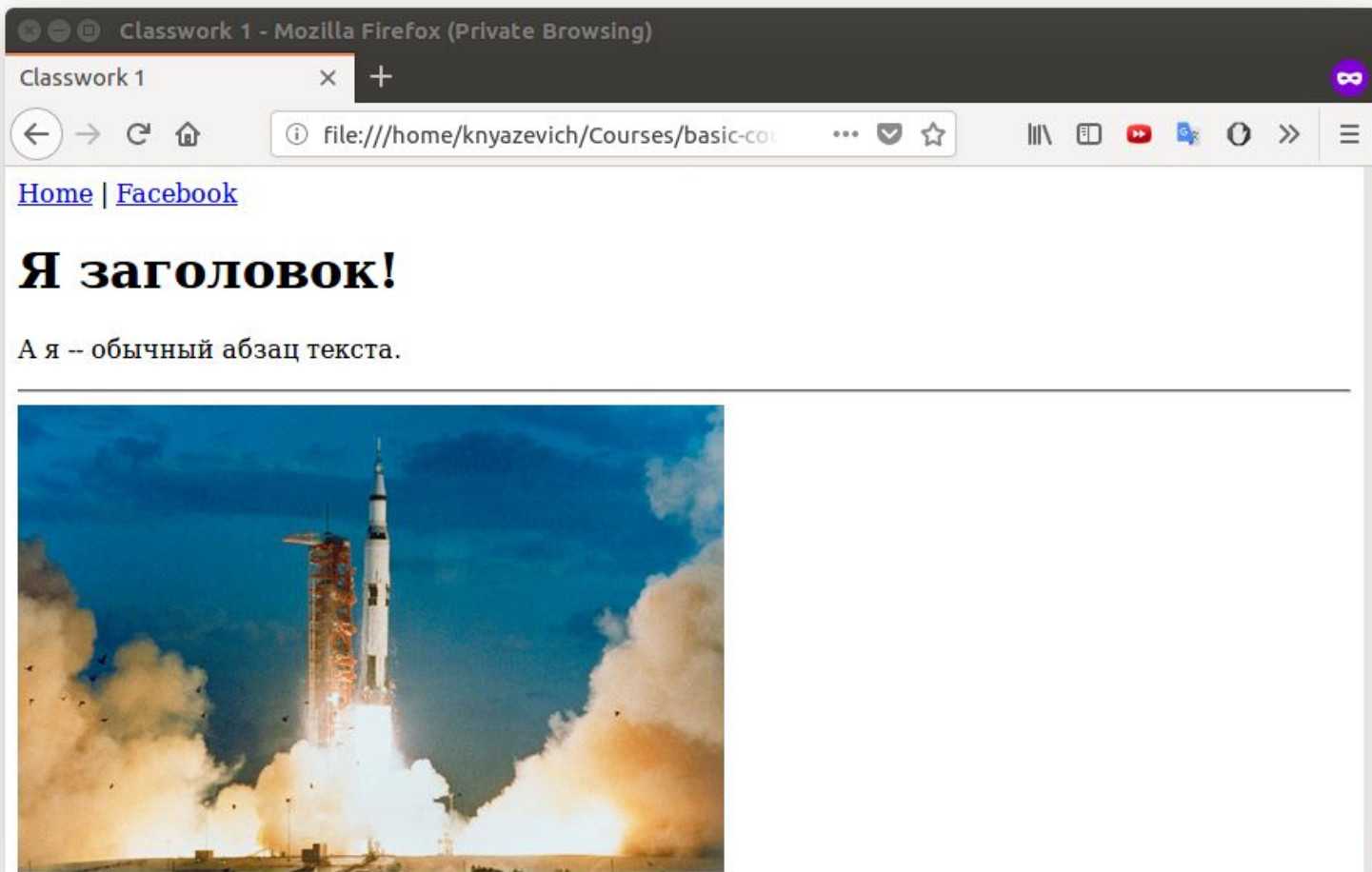
```
<body>
  <div class="news">
    <div class="news-item">
      <h1 class="news-title">Моя первая статья</h1>
      <p class="news desc">Далеко-далеко за словесными горами в стране.</p>
      <a href="/news-01.html" class="news-link">Читать далее</a>
    </div>
    <div class="news-item">
      <h1 class="news-title">Моя вторая статья</h1>
      <p class="news desc">Далеко-далеко за словесными горами в стране.</p>
      <a href="/news-02.html" class="news-link">Читать далее</a>
    </div>
  </div>
</body>
```



## Самостоятельная работа

Создаем страницу **index.html**, которая содержит всю необходимую базовую разметку и соответствует изображению на следующем слайде.

- Ссылка **Home** никуда не ведет, она заглушечная. **Facebook** ведет на facebook.com.
- Картинка кликабельна. При нажатии откроется полная версия.



## Разобранные требования

- В файле должен присутствовать DOCTYPE, favicon, стили и шрифты подключаются через `<link>`, присутствуют все базовые элементы.
- Выполнена HTML-разметка всех страниц и всех элементов на страницах.
- Страница проходит проверку валидатором (<http://validator.w3.org> или плагин W3C Validator в Sublime Text 3).
- Код должен выглядеть аккуратно и понятно для стороннего человека.  
Предпочтителен Google HTML Styleguide (<https://google.github.io/styleguide/htmlcssguide.html#HTML>).