



ҚАЗАҚСТАН РЕСПУБЛИКАСЫ
БІЛІМ ЖӘНЕ ҒЫЛЫМ МИНИСТРЛІГІ
МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
РЕСПУБЛИКИ КАЗАХСТАН
MINISTRY OF EDUCATION AND SCIENCE
OF THE REPUBLIC OF KAZAKHSTAN



Л. Н. ГУМИЛЕВ АТЫНДАҒЫ
ЕУРАЗИЯ ҰЛТТЫҚ УНИВЕРСИТЕТІ
ЕВРАЗИЙСКИЙ НАЦИОНАЛЬНЫЙ
УНИВЕРСИТЕТ ИМ. Л. Н. ГУМИЛЕВА
GUMILYOV EURASIAN
NATIONAL UNIVERSITY



Студенттер мен жас ғалымдардың
«Ғылым және білім - 2015»
атты X Халықаралық ғылыми конференциясының
БАЯНДАМАЛАР ЖИНАҒЫ

СБОРНИК МАТЕРИАЛОВ
X Международной научной конференции
студентов и молодых ученых
«Наука и образование - 2015»

PROCEEDINGS
of the X International Scientific Conference
for students and young scholars
«Science and education - 2015»

УДК 001:37.0
ББК72+74.04
Ғ 96

Ғ96

«Ғылым және білім – 2015» атты студенттер мен жас ғалымдардың X Халық. ғыл. конф. = X Межд. науч. конф. студентов и молодых ученых «Наука и образование - 2015» = The X International Scientific Conference for students and young scholars «Science and education - 2015». – Астана: <http://www.eni.kz/ru/nauka/nauka-i-obrazovanie-2015/>, 2015. – 7419 стр. қазақша, орысша, ағылшынша.

ISBN 978-9965-31-695-1

Жинаққа студенттердің, магистранттардың, докторанттардың және жас ғалымдардың жаратылыстану-техникалық және гуманитарлық ғылымдардың өзекті мәселелері бойынша баяндамалары енгізілген.

The proceedings are the papers of students, undergraduates, doctoral students and young researchers on topical issues of natural and technical sciences and humanities.

В сборник вошли доклады студентов, магистрантов, докторантов и молодых ученых по актуальным вопросам естественно-технических и гуманитарных наук.

УДК 001:37.0
ББК 72+74.04

ISBN 978-9965-31-695-1

©Л.Н. Гумилев атындағы Еуразия
ұлттық университеті, 2015

3. Traxler John. Current State of Mobile Learning. International Review on Research in Open and Distance Learning (IRRODL) 8, no. 2. 2007. www.irrodl.org/index.php/irrodl/article/view/346/875.
4. Образование в Японии <http://www.orientalica.com/lofiversion/index.php?t203.html>.
5. Mobiledu and Widsets for China. <http://mobiled.uiah.fi/?p=67>
6. И.Н. Голицына Мобильное обучение как новая технология в образовании.

УДК 004.438.5

HTML5, НОВЫЕ ВОЗМОЖНОСТИ СВЯЗАННЫЕ С НИМ СПЕЦИФИКАЦИИ

Өмірзақ Ислам Аманжолұлы

islam.omirzak@gmail.com

Студент 3-го курса кафедры информатики факультета информационных технологий
ЕНУ им. Л.Н.Гумилева, Астана, Казахстан

Научный руководитель - Г.Б. Балгожина, старший преподаватель

HTML5 (англ. HyperText Markup Language, version 5) — язык для структурирования и представления содержимого всемирной паутины. Хотя стандарт был завершён только в 2014 году(предыдущая, четвёртая, версия опубликована в 1997 году), ещё с 2013 года браузерами оперативно осуществлялась поддержка, а разработчиками — использование рабочего стандарта (англ. *HTML Living Standard*).

Цель разработки HTML5 — улучшение уровня поддержки мультимедиа-технологий, сохраняя при этом удобочитаемость кода для человека и простоту анализа для парсеров. Во всемирной паутине долгое время использовались стандарты HTML 4.01, XHTML 1.0, и XHTML 1.1 веб-страницы на практике оказывались свёрстаны с использованием смеси особенностей, представленных различными спецификациями, включая спецификации программных продуктов, например веб-браузеров, а также сложившихся общеупотребительных приёмов. HTML5 был создан, как единый язык разметки, который мог бы сочетать синтаксические нормы HTML и XHTML. Он расширяет, улучшает и рационализирует разметку документов, а также добавляет единое API для сложных веб-приложений [1].

Спецификация HTML5 значительна по объёму — она в три раза больше описания HTML 4.01. Значительная часть спецификации адресована разработчикам браузеров и нацелена на обеспечение совместимости реализаций HTML5, выполненных различными производителями.

У HTML5 есть три основных цели:

1. **установить особенности современных браузеров, в которых возможна интероперабельность;**
2. **впервые определить обработку ошибок;**
3. **доработать язык для упрощения создания веб - приложений.**

HTML5 несет в себе множество изменений разного уровня и разной важности. Принципиально, ключевые изменения можно разделить на 5 блоков:

Семантика. В HTML5 появился ряд новых семантических тегов, позволяющих более осмысленно организовывать внутреннюю структуру веб-страниц. Это включает как блочные теги вроде header, footer, article, так и теги для разметки текста, например, mark, ruby, details. Ряд существующих тегов HTML4 признан устаревшим, отдельные теги поменяли свое значение, определенные изменения претерпели атрибуты.

Мультимедиа. HTML5 добавляет изначальную поддержку мультимедийного контента (аудио и видео) прямо в HTML-разметке — с соответствующим API для управления воспроизведением (и некоторыми проблемами с кодеками).

Графика. Работать с графикой на стороне клиента стало заметно проще. В HTML5 добавлен

элемент canvas и специальный API на JavaScript для работы с ним. Canvas представляет собой динамическую «поверхность», поверх которой можно программно рисовать. Также в HTML5 официально включен тег svg, позволяющий внедрять векторную графику, описываемую соответствующим веб-стандартом (SVG, Scalable Vector Graphics).

Веб-формы. Новые элементы веб-формы: как типы, так и атрибуты, позволяющие расширить возможности традиционных форм встроенными средствами без использования дополнительных библиотек — от подсказок в поле ввода (placeholder) и проверки вводимых значений до специальных элементов для ввода дат и цвета.

JavaScript APIs. Как обозначенные выше API для работы с графикой и мультимедиа, так новые возможности по перемещению объектов (Drag & Drop) и работе с историей переходов (History API), а также ряд мелочей, вроде возможности сделать контент редактируемым прямо в текущем месте с помощью Content Editable атрибутов.

Новая спецификация HTML несет в себе множество интересных изменений, сильно отличающихся по своему масштабу: от тонкостей смысла уже существующих элементов до принципиально новых возможностей.

Большинство возможностей HTML5 направлено на создание интерактивных приложений, многие из них будут интересны как тем, кто занимается версткой, так и тем, кто отвечает за написание скриптов. В HTML5 добавлены 28 элементов с новой семантикой, которые можно использовать при создании традиционных «статических» веб-страниц, а также ряд элементов управления формами, позволяющих полностью отказаться от необходимости валидации форм с помощью JavaScript. Разметка, использующая элементы `<div>`, превращается в семантически организованную систему с помощью новых структурных элементов HTML5: `<nav>`, `<header>`, `<footer>`, `<aside>`, `<article>` и др.

При открытии редактора кода зададим тип документа и предварительные сведения.

Тип документа:

```
<!doctype html>
```

Такое объявление типа скорее напоминает не правило, а заклинание: оно предназначено для тех браузеров, которым для перехода в стандартный режим требуется указание типа документа, а такой код является одновременно и кратким, и надежным.

Далее определим кодировку символов документа. Если этого не сделать, появляется неясная, но реальная угроза безопасности. Кодировка должна быть указана в первых 512 байтах документа; лучше всего использовать UTF-8 (если нет серьезных оснований для выбора другой кодировки).

```
<!doctype html>
```

```
<meta charset=utf-8>
```

Если обратить внимание на тег `<meta>`, то, этот тег гораздо короче (раньше он бы выглядел так: `<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">`). Старый вариант остается допустимым, однако новый использовать все же предпочтительнее, поскольку он везде работает и его проще напечатать.

Вы также заметите, что мною не использованы кавычки для атрибута `charset="utf-8"`. Кроме того, мною не закрыт тег `<meta charset=utf-8 />`.

Поскольку HTML5 не является XML-языком, этого делать не нужно. Однако это и не запрещено. Все варианты, приведенные ниже, одинаково допустимы.

```
<META CHARSET=UTF-8>
```

```
<META CHARSET=UTF-8 />
```

```
<META CHARSET="UTF-8">
```

```
<META CHARSET=""UTF-8" />
```

```
<meta charset=utf-8>
```

```
<MeTa CHARset=utf-8>
```

Откуда такая свобода в синтаксисе? Ответ прост, для браузеров синтаксис XHTML-документов типа `text/html` никогда не был важен — его проверкой занимался валидатор XHTML. Поэтому если бы синтаксис HTML5 требовал соблюдения каких-то конкретных

правил, набор этих правил был бы полностью произвольным, а при их несоблюдении страница, будучи невалидной, все равно правильно отображалась бы во всех браузерах.

Поскольку речь здесь идет о невероятной свободе в синтаксисе (в сравнении с XHTML), давайте попробуем нарушить все существующее правила и после задания заголовка документа перейдем сразу к контенту.

```
<!doctype html>
<meta charset=utf-8>
<title>Interesting blog</title>
<p>Today I drank coffee for breakfast.
14 hours later, I went to bed.</p>
```

Как ни странно, этот увлекательный блог прекрасно проходит валидацию, хотя в нем нет ни тега `<html>`, ни тега `<head>`, ни тега `<body>` (рис. 1.1).

The screenshot shows a web browser window with the address bar displaying `https://html5.validator.nu`. The page title is "(X)HTML5 validation results". Below the title, there is a "Validator Input" section with a "File Upload" button, an "Обзор..." button, and a "Файл не выбран." message. There are also checkboxes for "Show Image Report" and "Show Source", and a "Validate" button. Below the input section, a green-bordered box contains the message: "The document is valid HTML5 + ARIA + SVG 1.1 + MathML 2.0 (subject to the utter previewness of this service).". Below this, the "Source" section displays the HTML code entered by the user, numbered 1 to 10. At the bottom, it shows "Total execution time 2 milliseconds." and links for "About this Service" and "More options".

(X)HTML5 validation results

Validator Input

File Upload Обзор... Файл не выбран.

☐ Show Image Report

☒ Show Source

Validate

The document is valid HTML5 + ARIA + SVG 1.1 + MathML 2.0 (subject to the utter previewness of this service).

Source

```
1. <!doctype html>↵
2. ↵
3. ↵
4. <meta charset=utf-8>↵
5. <title>Interesting blog</title>↵
6. ↵
7. ↵
8. <p>Today I drank coffee for breakfast.↵
9. 14 hour later, I went to bed.</p>↵
10.
```

Total execution time 2 milliseconds.

[About this Service](#) • [More options](#)

Рис 1.1. Потрясающий факт: документ без тегов `<head>`, `<body>` и `<html>` успешно проходит валидацию

В 2004 году главный редактор спецификации HTML5 Ян Хиксон проиндексировал с помощью Google 1 млрд веб-страниц, чтобы посмотреть, из чего сделана «реальная» сеть. Одна из статей, в которых впоследствии были опубликованы результаты содержит список самых часто используемых имен классов в этих HTML-документах. (рис. 1.2)[2].

Можно заметить, что если исключить из рассмотрения классы, явно относящиеся к оформлению, мы можем получить представление о структурах, которые авторы пытаются реализовать при создании веб-страниц.

Точно так же, как в HTML4 обнаруживаются характерные черты раннего поколения Сети - мира ученых и инженеров (именно поэтому в нем есть такие элементы как `<kbd>`, `<samp>` и `<var>`), HTML5 служит отражением состояния Сети в период его разработки: добавлено 28 новых элементов, и выбор многих из них основан на данных нижеприведенного списка имен классов и id, поскольку именно то, что разработчики на

самом деле используют [3].

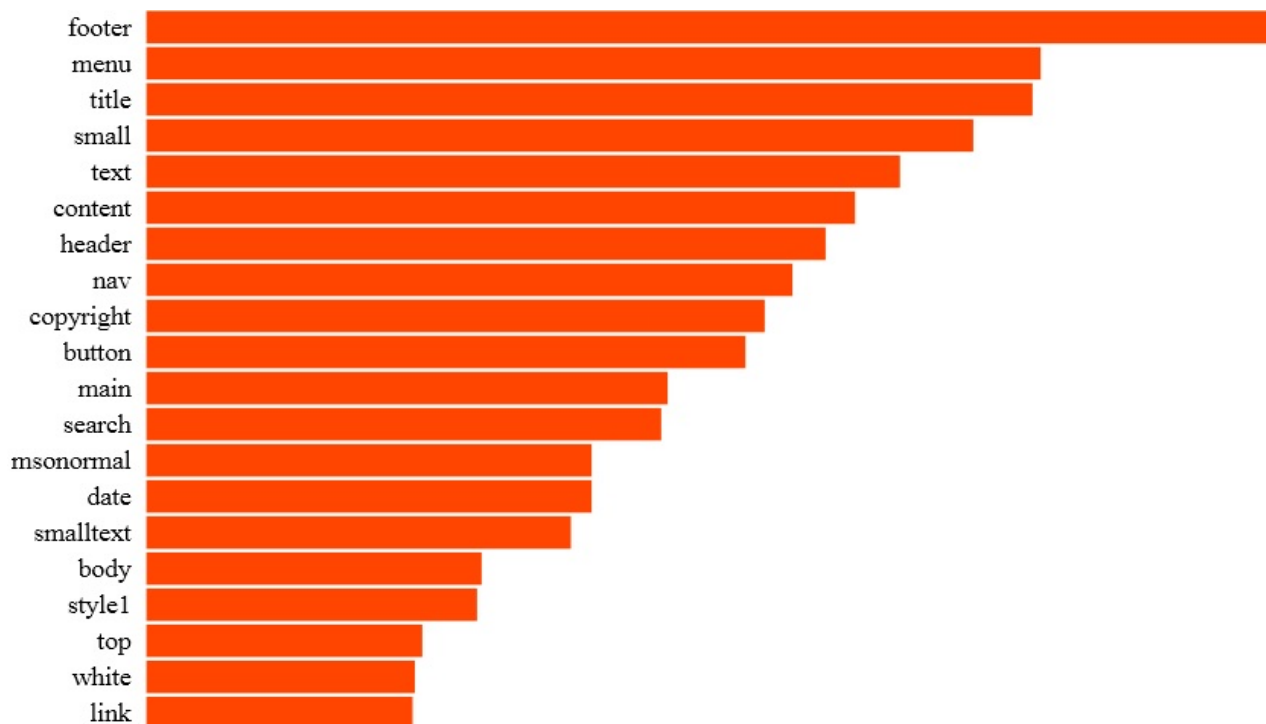


Рис. 1.2. Имена классов

Новые теги:

<article>

Тег <article> задает содержание сайта вроде новости, статьи, записи блога, форума или др.

<aside>

Определяет блок сбоку от контента для размещения рубрик, ссылок на архив, меток и другой информации. Такой блок, как правило, называется «сайдбар» или «боковая панель».

<audio>

Добавляет, воспроизводит и управляет настройками аудиозаписи на веб-странице. Путь к файлу задается через атрибут src или вложенный тег <source>. Внутри контейнера <audio> можно написать текст, который будет выводиться в браузерах, не работающих с этим тегом.

<bdi>

Элемент <bdi> указывает фрагмент текста, который должен быть изолирован от изменения направления вывода текста. Такое поведение важно для текстов, одновременно содержащих разные языки, читающихся слева направо и справа налево.

<canvas>

Создает область, в которой при помощи JavaScript можно рисовать разные объекты, выводить изображения, трансформировать их и менять свойства. При помощи тега <canvas> можно создавать рисунки, анимацию, игры и др.

<command>

Создает команду в виде переключателя, флажка или обычной кнопки. Тег <command> должен располагаться внутри <menu>, в противном случае он не будет показан.

<datalist>

Создает список вариантов, которые можно выбрать при наборе в текстовом поле. Изначально этот список скрыт и становится доступным при получении полем фокуса.

<details>

Тег <details> используется для хранения информации, которую можно скрыть или

показать по требованию пользователя. По умолчанию содержимое тега не отображается, для изменения статуса применяется атрибут `open`.

<figcaption>

Содержит описание для тега `<figure>`. Тег `<figcaption>` должен быть первым или последним элементом в группе

<figure>

Используется для группирования любых элементов, например, изображений и подписей к ним.

<footer>

Тег `<footer>` задаёт «подвал» сайта или раздела, в нём может располагаться имя автора, дата документа, контактная и правовая информация.

<header>

Тег `<header>` задает «шапку» сайта или раздела, в которой обычно располагается заголовок.

<hgroup>

Используется для группирования заголовков веб-страницы или раздела. Внутри располагаются теги заголовков от `<h1>` до `<h6>`.

<mark>

Тег `<mark>` помечает текст как выделенный. Такой текст ничем не отличается от обычного, но его вид может быть изменен с помощью стилей. В браузере Chrome и Firefox фоновый цвет текста внутри `<mark>` выделяется желтым цветом.

<meter>

Используется для вывода значения в некотором известном диапазоне. Используется преимущественно для отображения числовых значений, например, количества результатов поиска, объема жидкости, давления и др.

<nav>

Тег `<nav>` задает навигацию по сайту. Если на странице несколько блоков ссылок, то в `<nav>` обычно помещают приоритетные ссылки. Также допустимо использовать несколько тегов `<nav>` в документе. Запрещается вкладывать `<nav>` внутрь `<address>`.

<progress>

Используется для отображения прогресса завершения задачи. Изменение значения происходит через JavaScript.

<section>

Задаёт раздел документа, который может включать в себя заголовки, шапку, подвал и текст. Допускается вкладывать один тег `<section>` внутрь другого.

<source>

Вставляет звуковой или видеофайл для тегов `<audio>` и `<video>`. Обобщенно такие файлы называются медийными.

<time>

Помечает текст внутри тега `<time>` как дата, время или оба значения. Может указываться непосредственно внутри контейнера `<time>`, либо задаваться через атрибут `datetime`.

<video>

Добавляет, воспроизводит и управляет настройками видеоролика на веб-странице. Путь к файлу задается через атрибут `src` или вложенный тег `<source>`.

Список использованных источников

1. <https://ru.wikipedia.org/wiki/HTML5>
2. <https://developers.google.com/webmasters/state-of-the-web/2005/classes>.
3. Лоусон Б., Шарп Р. Изучаем HTML5. 2-е издание – СПб.: Питер, 2011. – 272 с.: ил. – (Серия «Библиотека специалиста»), ISBN 978-5-459-01156-2.