Видео недели:

1. [Семантическая верстка (теги HTML5)](https://youtu.be/Vq7LlmUSToI)
2. [Теги форм](https://www.youtube.com/watch?v=aK0FzFPfPQE)
3. [Что такое хорошая разметка](https://www.youtube.com/watch?v=4deYHrpi2y4)

**1. Теги HTML5 (семантические)**

**Что это и зачем они нужны**

**Семантические теги** - это одно из самых значимых нововведений в **HTML5**. До их появления вся разметка веб-страниц строилась при помощи элементов <div>, которым присваивались соответствующие его смыслу атрибуты (например, <div id="header">). С их помощью в HTML-документе размещали "шапку" и "подвал", боковые панели, навигацию и многое другое. Это приводило к тому, что для поисковых систем и браузеров все элементы страницы были одинаковыми, ведь тег <div> - это по сути обычная прямоугольная область на странице, не несущая какую-то конкретную информацию, кроме той, которую в нее заложит разработчик. Для решения этой проблемы было принято решение ввести семантические теги, которые делают структуру страницы более понятной и логичной.

Семантические теги HTML5 четко описывают, какой контент содержат (например, было <div id="footer">, стало <footer>). Для отображения внешнего вида элементов не задано никаких правил - элементы можно стилизовать по своему усмотрению. Для всех элементов доступны [‎](https://html5book.ru/html-attributes/)универсальные атрибуты (id, class, style и другие).

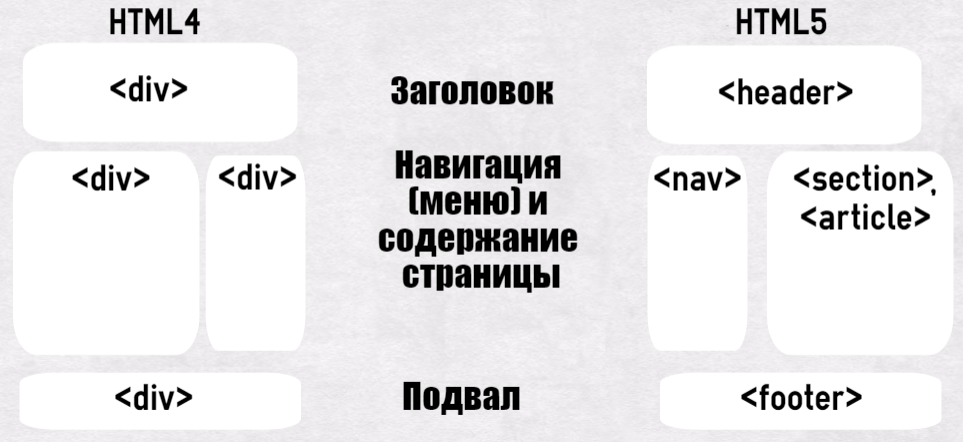
Это важно для смартфонов и ридеров – например, используя браузер Safari на iPhone или iPad, можно перейти в режим чтения, когда на экране выводится только текстовая информация страницы – и воспринимать ее гораздо легче, ведь нет отвлекающих элементов в виде картинок, баннеров, рекламы. Но этот режим будет доступен лишь в том случае, если страница имеет семантическую разметку, благодаря которой браузер может понять, где находится заголовок, где текст, а где навигация.

Не стоит забывать и о поисковых роботах – семантическая разметка поможет им правильно проанализировать страницу, отделив важную информацию, касающуюся содержимого страницы, от второстепенной (шапка, подвал, навигация). И это обязательно положительно скажется на SEO-продвижении сайта.

Таким образом, главное предназначение семантической разметки – делать структуру документа более понятной на всех этапах использования, от разработки до публикации в сети и индексации поисковиками.

**Структура документа в HTML4 и HTML5**

Смотрите какая наглядная разница в разметке:



Пример посложнее:

Несложно заметить, что обновленная разметка представляет собой более прозрачную и интуитивно понятную структуру, которая ко всему прочему удобна при разработке.

**Новые семантические элементы разметки HTML5**

Причем, в HTML5 не исчезли какие-то старые элементы – наоборот, теперь разработчики могут использовать новые, более удобные теги, каждый из которых выполняет свою особенную функцию.

Вот основные из них:

* **header**
* **nav**
* **article**
* **section**
* **main**
* **aside**
* **footer**
* **address**
* **figure**

**2. Теги форм**

Сложно представить себе страницу без интерактивности. Практически всегда, когда у пользователя есть возможность как-то взаимодействовать со страницей, это делается благодаря формам и полям ввода в них.

Логика работы форм - пользователь вводит данные, которые потом обработает клиентское, либо серверное приложение. Если необходимо отправлять данные на сервер, то все теги форм необходимо поместить внутрь формы (тег <form>).

Ключевые теги форм:

* Поля **input**
* Большой текст **textarea**
* Выпадающие списки **select**
* Кнопка **button**
* Метки **label**
* Группы **fieldset**
* **Навигация по полям**

Пример простой готовой формы:

<form>

<p>Имя:</p>

<p><input type="text" name="name" value="Ваше имя" /></p>

<p>Комментарий:</p>

<p><textarea name="comments" rows="5" cols="20">Ваш комментарий</textarea></p>

<p>Вы:</p>

<p><input type="radio" name="areyou" value="male" /> Мужчина</p>

<p><input type="radio" name="areyou" value="female" /> Женщина</p>

<p><input type="radio" name="areyou" value="hermaphrodite" /> Ни то ни другое</p>

<p><input type="submit" /></p>

<p><input type="reset" /></p>

</form>

**3. Спецсимволы HTML**

Помимо тегов в языке HTML используется специальный управляющий символ **&** — амперсант. Этот символ используется для вывода специальных символов и символов, которые нельзя ввести с клавиатуры.

Самый часто используемый спецсимвол – это неразрывный пробел (он пишется так - ).

Он используется для контроля переноса строки (после данного символа автоматический перевод строки невозможен) и для вставки в текст подряд идущих пробелов (обычно браузер сокращает идущие подряд пробелы и показывает не больше одного сразу).

Часто используемые спецсимволы:

**4. Про разметку**

**Алгоритм - как мы делаем разметку**

* Определяем повторяющиеся элементы (header, footer, возможно какие-то доп. меню и пр.) и основное содержимое.
* Внутри них размечаем основные смысловые элементы.
* Находим заголовки на странице и в каждом блоке (если они есть).
* Размечаем всё оставшееся методом исключения:
  + если получилось найти самый подходящий тег — берем его
  + для блоков с переносами строки <div>
  + для абзацев в тексте <p>
  + для маленьких элементов без переноса строки <span>

**Валидация кода**

Валидация - это автоматическая проверка кода на соответствие каким-то правилам. Хороший сервис валидации можно найти тут <http://validator.w3.org>

Валидность кода определяет то, как будет выглядеть страница или веб-приложение в разных браузерах и на различных платформах. Валидный код во многих браузерах отображается предсказуемо. Он загружается быстрее невалидного. Валидность влияет на восприятие страниц и сайтов поисковыми системами.

Валидация зависит от того, какой указан тип документа <!DOCTYPE> (мы обычно используем <!DOCTYPE html>, что соответствует современному стандарту HTML5).

Ошибки (**Error**) обязательно исправляем. Предупреждения (**Warning)** исправлять не обязательно, но это повод задуматься и почитать спецификацию.

**Как определить, хорошая разметка или плохая**

1. Она работает!
2. Она простая и понятная для других людей
3. В ней нет ничего лишнего
4. Соблюдается определенный в компании стиль написания кода
5. Она проходит проверку на валидаторе

**Что такое простая?**

В ней минимально необходимое число тегов. Имена классов читаемые и говорят о том, для чего этот элемент нужен. Есть границы смысловых блоков, легко понять к какому блоку принадлежит тег.

**Примеры**

Пример простой верстки страницы:

Пример посложнее:

**Домашнее задание на эту неделю:**

**Практическое задание 💎**

1. **Сделать** **страницу-опросник** для обладателей котиков:
   1. Кличка питомца (*текстовое поле*)
   2. Имя и контакты владельца (*текстовые поля*)
   3. Порода питомца (*выпадающий список пород*)
   4. Чем питается (*группа галочек, можно выбрать несколько*): сухой корм, влажный корм, натуральный корм
   5. Самец или самка (*группа радиокнопок, можно выбрать одну*)
   6. Комментарий владельца (*большое текстовое поле*)
   7. Приложить фото питомца (*можно выбрать файл с диска*)
   8. Кнопка "Сохранить"
2. **Сделать страницу** блогера по правилам **сематической разметки** (тематика блога на твое усмотрение), используя не менее 80% семантических тегов из урока
3. Проверить свои страницы **валидатором** и исправить ошибки (если они есть)

**Вопросы по темам 💎**

1. Что такое валидация?
2. Какой спецсимвол используется для указания знака ©?
3. Найдите пример еще тега HTML5, который не вошел в урок
4. Какой тег мы используем для создания меню?
5. Чем плохо использовать div для всего?
6. Самостоятельно разберитесь, как вставлять в страницу видео тегами HTML5 и приложите пример кода
7. Как сделать блок радиокнопок, чтобы выбирался только один пункт?
8. Каким тегом можно сделать поле ввода даты?
9. Как определяется, какой именно пункт будет выбран по умолчанию в выпадающем списке?
10. Что такое хорошая разметка?

**Ссылка на свою страничку из википедии 💎**

Наша внутренняя википедия:

<http://wiki.itgirlschool.ru/doku.php?id=start>

Там нужно зарегистрироваться, чтобы начать создавать страницы. Проще всего создать страницу через форму поиска. Ищете, есть ли такая страница в строке поиска (лучше использовать английские слова для имен страниц) и если нет, то создаем.

Форма редактирования страницы похожа на Word и интуитивно понятна. Либо со страницы HTML, там уже есть примеры тегов. Можно начать с любого полюбившегося тега и сделать страничку про него - красивую, понятную и с примерами кода. Или сделать хорошие и наглядные примеры для уже существующих тегов.

ОТВЕТЫ:

1. Валидация – это автоматическая проверка кода на соответствие каким-то правилам.
2. &copy; &#169;
3. <table> - тег для создания таблиц
4. <nav>
5. <div> не стоит использовать для всего потому, что для поисковых страниц и браузеров все элементы страницы с тегом <div> одинаковые, им сложно проанализировать страницу и понять где какой элемент. Для решения этой проблемы была введена семантическая разметка.
6. Тег <video> помогает добавить видео на страницу.

Пример кода:

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=>, initial-scale=1.0">

    <title>Добавление видео на страницу</title>

</head>

<body>

    <video src="https://www.youtube.com/watch?v=dnPd4yYi8mA" width="400" height="300" preload="auto" controls loop>

    </video>

    <p>смешное видео с котиками</p>

</body>

</html>

1. Чтобы ограничить выбор пользователя до чего-то одного, нужно установить для всех вариантов одинаковое имя.
2. <input type=”date” />
3. За то, будет ли выбран элемент из списка по умолчанию отвечает атрибут тега <option> selected
4. Хорошая разметка:

-рабочая

-простая

-ничего лишнего

-соблюден стиль компании

-проходит проверку на валидаторе