

DIGITAL
DESIGN



Школа инженера

Основы веб-разработки: html5, css3

Постников Илья

www.digdes.ru



HTML (HyperText Markup Language)

- Язык создания электронных документов в сети интернет
- Выработка спецификаций World Wide Web Consortium (W3C)
- Принятие стандарта HTML5 – 2014 год
- Последние спецификации: <https://www.w3.org/TR/html52/> от 26.05.2023

Что нового:

- Определяет новый алгоритм парсинга для создания структуры DOM
- Переопределяет правила и семантику элементов ранее существовавших в HTML
- Добавлен ряд новых элементов и тегов

Применение:

- Язык гипертекстовой разметки
- Платформа для создания Web-приложений совместно с CSS3 и JavaScript
- Создание приложений для мобильных платформ под Android, iOS, Windows Mobile



Поддержка броузерами

Практически полная поддержка:

- Google Chrome
- Firefox (Mozilla)
- Opera
- Microsoft Edge
- Internet Explorer 11

Частичная:

- Internet Explorer 8, 9

Рабочие инструменты:

- Google Chrome (F12)
- Notepad++



Структура документа HTML (01_firsthtml5.html)

Текстовый файл с расширением *.html

1. `<!DOCTYPE html>` - декларация типа документа (HTML5) для браузера
2. `<link href="external.css" rel="stylesheet">` Открывающий тег `<html>`
3. Заголовок `<head>`
 - 3.1. `<meta charset="utf-8">` - указание кодировки
 - 3.2. `<title>` - текстовый заголовок страницы
 - 3.3. `<link href="external.css" rel="stylesheet">` - подключение внешних файлов стилей
 - 3.4. `<style type="text/css">` - блок стилей непосредственно на странице
4. Содержимое страницы `<body>`
 - 4.1. Основной контент в виде набора тегов html
 - 4.2. `<script src="/.../script.js"></script>` - подключаемые внешние javascript
 - 4.3. `<script> alert('Hello!'); </script>` - внутренние javascript

```
<!DOCTYPE html>
```

```
<html lang="ru">
```

```
<head>
```

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

```
<title>Первый документ HTML5</title>
```

```
</head>
```

```
<body>
```

```
<div>Наш первый документ HTML5</div>
```

```
</body>
```

```
</html>
```

Проверка разметки: <https://validator.w3.org>



Элементы группировки (02_group.html)

- Блок-контейнер `<div>`
- Параграф `<p>`
- Перевод строки `
` или `
`
- Форматирование `<pre>`
- Строчный элемент ``
- Не пытайтесь в строчный элемент вставить блочный!



Заголовки и форматирование текста

(03_header_and_format.html)

- Заголовки `<h1>...<h6>`
- Жирный текст `` и важный ``
- Курсивный текст `<i>` и ``
- Зачеркнутый текст `` и `<s>`
- Подчеркнутый текст `<u>`
- Вставленный или добавленный текст `<ins>`
- Меньший текст `<small>`
- Под строкой `<sub>`
- Над строкой `<sup>`
- Выделение цветом для придания важности `<mark>`



Операции с изображениями (04_image.html)

Элемент ``

- `src` - путь к файлу изображения
- `alt` - текстовое название

```

```

Элемент `base` - определяет базовый (корневой) путь



Списки (05_lists.html)

`` ненумерованный список (unordered list)

Стиль **list-style-type**:

- `disc`: черный диск
- `circle`: пустой кружок
- `square`: черный квадратик

Стиль **list-style-image** задает картинку

`` нумерованный список (ordered list)

Стиль **list-style-type**:

- `decimal`: десятичные числа, отсчет идет от 1
- `decimal-leading-zero`: десятичные числа, которые предваряются нулем, например, 01, 02, 03, ... 98, 99
- `lower-roman`: строчные римские цифры, например, i, ii, iii, iv, v
- `upper-roman`: заглавные римские цифры, например, I, II, III, IV, V...
- `lower-alpha`: строчные римские буквы, например, a, b, c..., z
- `upper-alpha`: заглавные римские буквы, например, A, B, C, ... Z

Атрибут `start="n"` - задает стартовый символ



Навигация с помощью ссылок (06_a-link.html)

<a> Обеспечивает навигацию между отдельными документами

- href: определяет адрес ссылки
- target: определяет, как документ по ссылке должен открываться
 - _blank: открытие html-документа в новом окне или вкладке браузера
 - _self: открытие html-документа в том же фрейме (или окне)
 - _parent: открытие документа в родительском фрейме, если ссылка расположена во внутреннем фрейме
 - _top: открытие html-документа на все окно браузера
 - framename: открытие html-документа во фрейме, который называется framename (название фрейма может быть произвольным)

`Компания Digital Design`

Ссылка в виде картинки:

``



Таблицы (07_table.html)

- `<table>`
- `<caption>` - общий заголовок таблицы
- `<thead>` - заголовок таблицы
- `<tbody>` - тело таблицы
- `<tfoot>` - подвал таблицы
- `<tr>` - строка таблицы
- `<td>` - ячейка таблицы
- `<th>` - ячейка заголовка или подвала

Атрибут `colspan="n"` - склеивает n ячеек таблицы по горизонтали

Атрибут `rowspan="m"` - склеивает m ячеек таблицы по вертикали



Фреймы (08_iframe.html)

<iframe> позволяет встраивать на страницу другую html-страницу

- src - путь к загружаемому ресурсу
- width - ширина фрейма
- height - высота фрейма

```
<iframe src="https://digdes.com" width="400" height="200"></iframe>
```



Элемент ввода input

Атрибут type

- **text**: обычное текстовое поле
- **password**: текстовое поле со звездочками
- **radio**: радиокнопка или переключатель. Из группы радиокнопок можно выбрать только одну
- **checkbox**: элемент флажок, который может находиться в отмеченном или неотмеченном состоянии
- **hidden**: скрытое поле
- **submit**: кнопка отправки формы
- **color**: поле для ввода цвета
- **date**: поле для ввода даты
- **datetime**: поле для ввода даты и времени с учетом часового пояса
- **datetime-local**: поле для ввода даты и времени без учета часового пояса
- **email**: поле для ввода адреса электронной почты
- **month**: поле для ввода года и месяца
- **number**: поле для ввода чисел
- **range**: ползунок для выбора числа из некоторого диапазона
- **tel**: поле для ввода телефона
- **time**: поле для ввода времени
- **week**: поле для ввода года и недели
- **url**: поле для ввода адреса url
- **file**: поле для выбора отправляемого файла
- **image**: создает кнопку в виде картинки



Текстовые поля (09_inputtext-textarea.html)

Однострочное `<input type="text" />`

- **value** - устанавливает значение по умолчанию в текстовом поле
- **name** - для идентификации поля ввода
- **readonly** - делает текстовое поле доступным только для чтения
- **required** - указывает, что текстовое поле обязательно должно иметь значение
- **pattern** - определяет шаблон, которому должен соответствовать вводимый текст
- **placeholder** - устанавливает текст приглашения
- **maxlength** - максимально допустимое количество символов в текстовом поле
- **size** - устанавливает ширину текстового поля в видимых символах

Многострочное `<textarea />`

- **cols** - количество столбцов
- **rows** - количество строк



Кнопки

Элемент `<button type="..">`

- `button` - кнопка общего назначения
- `reset` - кнопка сброса значений формы
- `submit` - кнопка для отправки формы

`type="submit"`

- `form` - форма, за которой закреплена кнопка отправки
- `formaction` - устанавливает адрес, на который отправляется форма
- `formenctype` - устанавливает формат отправки данных
- `formmethod` - устанавливает метод отправки формы (post или get)

Элемент `<input type="..">`

- `reset` - кнопка сброса значений формы
- `submit` - кнопка для отправки формы
- `image` - кнопка-иконка `<input type="image" src="digdes.png" />`



Формы и отправка данных на сервер (10_forms.html)

Элемент `<form>...</form>`

- `method` - метод отправки данных на сервер `get` или `post` (`http`)
- `action` - адрес, на который передаются данные формы
- `enctype` - устанавливает тип передаваемых данных

`enctype`

- `application/x-www-form-urlencoded` - кодировка отправляемых данных (по умолчанию)
- `multipart/form-data`: эта кодировка применяется при отправке файлов
- `text/plain`: эта кодировка применяется при отправке простой текстовой информации



Элементы ввода различных параметров (11_inputnumber-other.html)

Числа `<input type="number" .../>`

- `min` - минимально допустимое значение
- `max` - максимально допустимое значение
- `readonly` - доступно только для чтения
- `required` - поле обязательно должно иметь значение
- `step` - шаг увеличения числа в поле
- `value` - значение по умолчанию

Ползунок `<input type="range" ... />`

Цвет `<input type="color" ... />`

Url `<input type="url" .../>`

Email `<input type="email" .../>`

Телефон `<input type="tel" .../>`

Флажки (чек-боксы) `<input type="checkbox" checked .../>`

Переключатели (радио-кнопки) `<input type="radio" checked.../>`

Список `<select>` - `<option>`



Cascading Style Sheets (Каскадные таблицы стилей)

- Определяет форматирование на html-странице

Структура объявления стиля

- селектор - определяет что форматировать
- блок объявления стиля - набор команд для форматирования
 - команда - это свойство и значение

селектор {

 свойство1: значение1;

 свойство2: значение2;

}

Стиль может быть определен в

- в отдельном файле *.css с подключением в заголовке html-документа: `<link rel="stylesheet" type="text/css" href="styles.css"/>` (низший приоритет применения при конфликте)
- в секции `<style>` заголовке html-документа (средний приоритет)
- в html- элементе через атрибут `style="width: 10px; height: 10px;"` (высший приоритет)

Валидация кода css: <http://jigsaw.w3.org/css-validator/>



Селекторы (12_css-selectors.html)

Селекторами могут быть

- названия html-элементов (div, span, p, h3, input и т.д.)
- классы для элементов с определенным атрибутом `class="black-square"`: `.black-square { color: black; }` - перед названием точка
- идентификаторы для элементов с атрибутом `id="square1"`: `#square1 { color: green; }` - перед названием значек #
- универсальный селектор для всех элементов `* { margin: 0; }`
- групповой - перечисление через запятую `div, .black-square, #square { border: 1px red solid; }`
- объединенный - без пробела `div.black-square { width: 40px; }`
- селектор потомков - для вложенных элементов `div .black-square { width: 40px; }` с пробелом
- селектор дочерних - `div > .black-square { width: 40px; }`
- селектор элементов одного уровня `div ~ .black-square { width: 40px; }` - тильда
- селектор атрибутов `input[type="text"] { border: 1px red solid; }`



Наследование и каскадность

- Вложенные html элементы наследуют стили своих контейнеров
- Не наследуются:
 - Внешние отступы (margin)
 - Внутренние отступы (padding)
 - Границы (border)

Каскадность определяет правила применения множества стилей к одному элементу

- селекторы тегов - 1 балл
- селекторы классов, атрибутов и псевдоклассов - 10 баллов
- селекторы идентификаторов - 100 баллов
- встроенные inline-стили (атрибут `style`) - 1000 баллов
- отмена правил и хардлок - `!important`



Блочная модель и отступы

- Каждый html элемент - это блок или контейнер с различным содержимым - текстом, изображением, таблицей и другими элементами, а также внутренними блоками
- Внешний отступ от других элементов - свойство `margin`

`margin`: отступ_сверху отступ_справа отступ_снизу отступ_слева

`margin-top`, `margin-right`, `margin-bottom`, `margin-left`

- Границы элемента - свойство `border`
 - **`border-width`**: устанавливает ширину границы
 - **`border-style`**: задает стиль линии границы
 - **`border-color`**: устанавливает цвет границы

`border`: толщина_границы цвет_границы стиль_линии

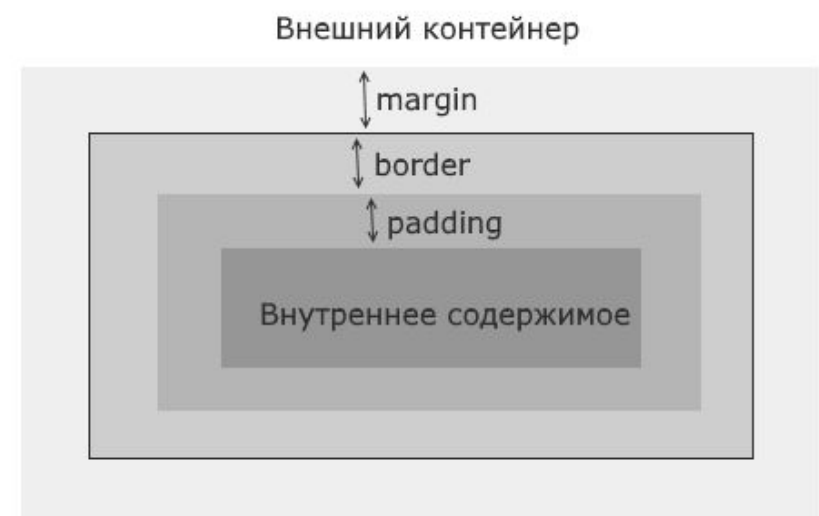
- Задание радиуса:

`border-radius`: верх_лев верх_прав ниж_прав ниж_лев

`border-top-left-radius`, `border-top-right-radius`,

`border-bottom-right-radius`, `border-bottom-left-radius`

- Внутренний отступ к внутреннему содержимому - свойство `padding` (аналогично `margin`)





Стили для шрифтов

- Семейства шрифтов
`font-family: Arial, Verdana, "Times New Roman";`
- Толщины шрифтов
`font-weight: 200; - 100, 200, 300, ... , 900`
- Курсив/наклон
`font-style: italic-курсив / normal-простой / oblique-простой с наклоном`
- Цвет - свойство `color`
- Высота или размер - свойство `font-size`
 - В пикселях `px`
 - В процентах относительно базового или унаследованного шрифта `%`
 - В единицах `em` (по умолчанию, `1em = 16px`)
 - В строковых константах:
 - **medium**: базовый размер шрифта браузера (16 пикселей)
 - **small**: 13 пикселей
 - **x-small**: 10 пикселей
 - **xx-small**: 9 пикселей
 - **large**: 18 пикселей
 - **x-large**: 24 пикселя
 - **xx-large**: 32 пикселя



Цвета и фон элементов (13_background.html)

- Задание цвета фона - свойство `background-color`
- Значение цвета можно задать:
 - названиями (black, white, green, red и т.д.)
 - шестнадцатичным значением: `#rrggbb`, например: `#1C25FF`
 - десятичными значениями RGB 0-255, например: `rgb(112, 34, 99)`
 - процентными значениями RGB 0%-100%, например: `rgb(20%,30%,40%)`
 - с дополнительным компонентом прозрачности RGBA 0-1, например: `rgba(21,34,45, .7)`
- Прозрачность - свойство `opacity` от 0-1
- Задание фона в виде изображения - свойство `background-image`
 - С абсолютным путем `background-image: url(http://somesite/picture.png)`
 - С относительным путем `background-image: url(../images/picture.png)`



Верстка с помощью Flexbox (14_flexbox.html)

- Включение контейнера в режим Flexbox
display: flex
- Направление оси задается свойством flex-direction
 - row - элементы располагаются в виде строки слева направо (по умолчанию)
 - colum - элементы располагаются в столбик сверху вниз
- Выравнивание элементов: justify-content
 - flex-start – по умолчанию (по левому краю)
 - flex-end
 - center
 - space-between
 - space-around
- Выравнивание по поперчной оси: align-items
 - stretch – по умолчанию (растягивание по высоте)
 - flex-start
 - flex-end
 - center
 - baseline

Flex-end



Center



Space-between



Space-around



Flex-start



Flex-end



Center



Baseline





Домашнее задание

Сделать страничку, с описанием любимого языка программирования, любимой картины, любимого места и т.д.

Страница должна иметь:

- Заголовок
- Центральную часть из двух и более колонок
 - Картинка (элемент ``)
 - Описание
 - Полезные ссылки (элемент `<a />`)
 - Форма опроса с отправкой на фейковый адрес (чекбоксы, радиокнопки, выбор из списка и другие контролы)
- Футер со своим копирайтом :)
- Необходимо использовать фон, отступы, выравнивание

Можно использовать любые режимы верстки:

- Flexbox верстку(см. [14_flexbox.html](#))
- Табличную верстку
- Блочную верстку

Используйте любые материалы из текущего курса и инета



Частые ошибки при верстке в ДЗ

- Страница не имеет структуры, хотя требования указаны – отсутствуют блоки, блочные столбцы и т.д.
- При изменении разрешения или переключении в режим смартфона страница “ломается” – блоки налезают друг на друга, текст выходит за рамки и т.д.
- Отсутствуют контролы
- Отсутствуют отступы, фон, рамки, выравнивание блоков и контролов
- Валидируйте ваш код! html на <https://validator.w3.org/> и css на <https://jigsaw.w3.org/css-validator/>
- Код html/css должен быть четко отформатирован!
- Желательно не использовать дополнительные фреймворки типа bootstrap или material-ui



Спасибо за внимание



Контакты



www.digdes.ru



info@digdes.com



Санкт-Петербург

наб. реки Смоленки, д. 33
телефон: +7 812 346 58 33

Москва

Варшавское шоссе, д. 36, стр. 8
телефон: +7 499 788 74 94

