



Основы верстки. Шпаргалка

Верстка веб-страницы

Верстка — это сборка макета дизайна или прототипа с помощью специальных инструментов, таких как HTML и CSS, еще часто используется JavaScript, но это уже входит в работу frontend-разработчика.

Веб-страница — это документ, доступный через Интернет и написанный с использованием языков HTML и CSS. Веб-страница может содержать текст, изображения, видео, ссылки и другие мультимедийные элементы.

HTML и CSS

HTML (HyperText Markup Language) — язык разметки, используемый для создания структуры веб-страниц. Он определяет элементы, такие как заголовки, абзацы, изображения и ссылки, которые отображаются в браузере.

CSS (Cascading Style Sheets) — язык стилей, используемый для описания внешнего вида элементов, созданных с помощью HTML. Он позволяет задавать цвета, шрифты, отступы, границы и другие стили для веб-страниц.

Этапы верстки веб-страницы

Верстка страницы или макета укладывается в несколько шагов:

1. Создание HTML-структурь.
2. Подключение CSS-стилей.
3. Связывание CSS с HTML.

Макет — это план или схема расположения элементов на веб-странице. Определяет структуру и визуальное представление страницы, включая расположение различных компонентов. Проще говоря, макетом называют дизайн или прототип веб-станицы.

Верстальщик — это специалист, который преобразует дизайн макета веб-страницы в код с использованием HTML, CSS и JavaScript, чтобы страница корректно отображалась в браузерах.

Пример структуры HTML-документа:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Title</title>
  <link href="styles.css" rel="stylesheet">
  <script src="scripts.js"></script>
</head>
<body>
  <h1>Header 1</h1>
  <h2>Header 2</h2>
  <h3>Header 3</h3>
  <div>Div</div>
  <span>Span</span>
  <p>Paragraph</p>
  <form>Form</form>
  <button>Button</button>
  <a>Link</a>
  
</body>
</html>
```

Основные HTML-теги

Теги структуры документа

- `<!DOCTYPE html>` — объявляет тип документа и предоставляет основную информацию для браузера.
- `<html> </html>` — определяет корневой элемент HTML-документа. Он содержит всю структуру веб-страницы, то есть все содержимое страницы находится внутри этого тега.
- `<head> </head>` — содержит метаинформацию о документе, такую как заголовок страницы, ссылки на стили CSS, скрипты JavaScript, метатеги для поисковых систем и другую информацию, которая не отображается на странице.
- `<body> </body>` — определяет тело документа. Всё содержимое веб-страницы, видимое для пользователей, размещается внутри этого тега.

Теги, которые помещаются внутри `<head>`

- `<meta charset="UTF-8">` — указывает на кодировку символов, используемую на странице. Это помогает избежать проблем с отображением текстов на разных языках. В данном случае используется кодировка UTF-8.
- `<title> </title>` — задает название страницы, которое отображается в заголовке вкладки браузера, а также используется при закладках и в результатах поиска.
- `<link href="styles.css" rel="stylesheet">` — используется для подключения внешних ресурсов к документу, в данном случае файла стилей CSS. Атрибут `href` указывает путь к файлу стилей, а `rel="stylesheet"` сообщает браузеру, что это файл стилей.
- `<script src="scripts.js"></script>` — используется для подключения и выполнения внешних файлов JavaScript. Атрибут `src` указывает путь к файлу скрипта.

Теги форматирования текста

- `<h1> </h1>` — `<h2> </h2>` — используются для создания заголовков разных уровней. `<h1>` является самым высоким уровнем, а `<h6>` — самым низким.
- `<p> </p>` — используется для создания абзацев текста на странице. Представляет собой блочный элемент, каждый абзац будет отображаться на отдельной строке.
- `ТЕКСТ` — используется для создания гиперссылок. Позволяет указать адрес URL или путь к другой веб-странице, изображению, файлу и т. д.
- `` — используется для вставки изображений на страницу. Требует атрибута `src`, который указывает путь к изображению.
- ` ` — создает неупорядоченный (маркированный) список.
- ` ` — определяет отдельный элемент списка.

Теги для работы с таблицами

- `<table>` `</table>` — создает таблицу,
- `<tr>` `</tr>` — определяет строку в таблице,
- `<th>` `</th>` — определяет ячейку заголовка,
- `<td>` `</td>` — определяет ячейку данных.

Теги для создания форм

- `<form>` `</form>` — используется для создания формы на странице, в которой пользователи могут вводить данные.
- `<input>` `</input>` — создает поле ввода.
- `<button>` `</button>` — создает кнопку для отправки формы.

Группировка элементов внутри блока

`<div>` `</div>` — представляет собой контейнерный элемент, который используется для группировки других элементов и создания блочных областей на странице. Обычно используется с CSS для стилизации и разделения содержимого.

Работа в CSS-файле

```
/* styles.css */  
h1 {  
    color: black;  
}  
.my-input {  
    border: none;  
}
```

Подключение CSS в HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
    ...
    <link href="styles.css" rel="stylesheet">
</head>
<body>
    ...
</body>
</html>
```

Селекторы

```
/* style.css */
p {
    color: blue;
}
.highlight {
    background-color: yellow;
}
#main-header {
    font-size: 24px;
}
```

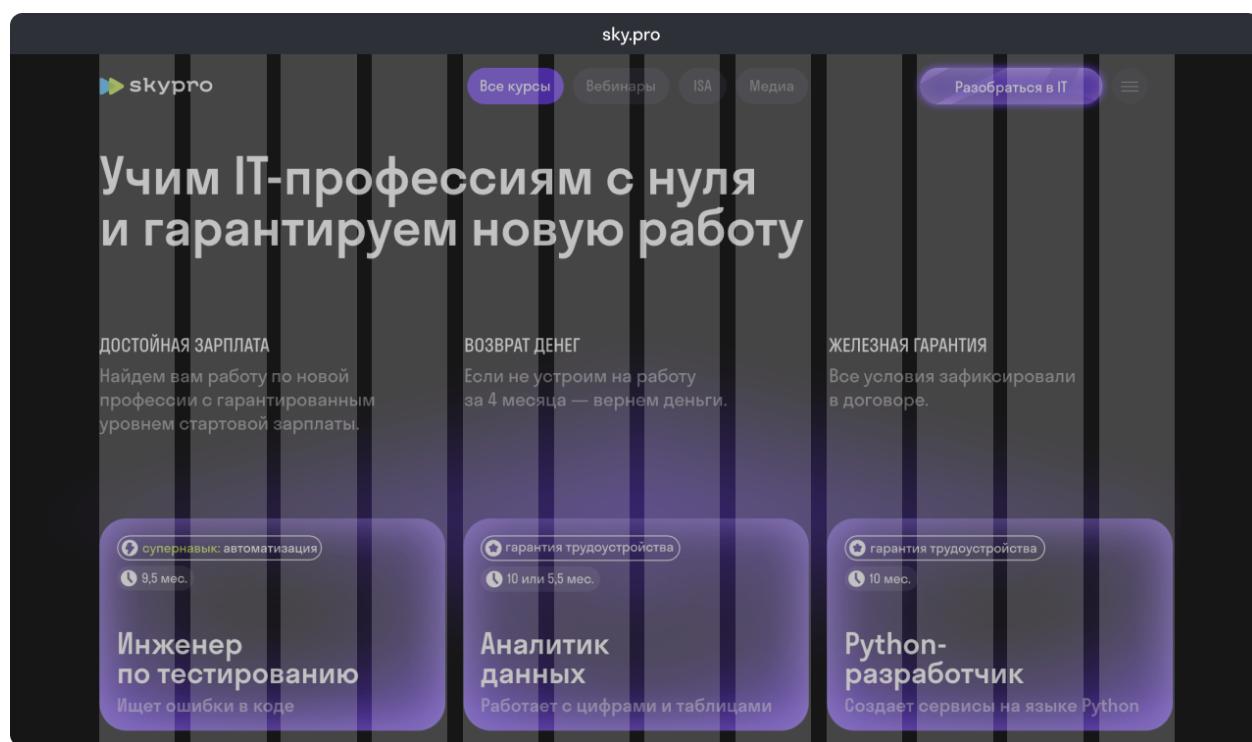
Применение в HTML:

```
<!DOCTYPE html>
<html lang="en">
<head>
    ...
    <link href="styles.css" rel="stylesheet">
</head>
<body>
    <p>Параграф с синим цветом текста</p>
    <div class="highlight">Блок с желтым цветом фона</div>
    <div id="main-header">Блок с размером шрифта 24 пикселя</div>
</body>
</html>
```

Bootstrap

UI Kit (User Interface Kit) в веб-разработке — это набор предварительно разработанных элементов пользовательского интерфейса, таких как кнопки, иконки, формы, карточки и другие компоненты. UI kits упрощают и ускоряют процесс создания веб-интерфейсов, обеспечивая единообразие стиля и функциональности. С их помощью разработчики и дизайнеры могут легко собирать интерфейсы, не создавая каждый элемент с нуля.

Bootstrap — свободный набор инструментов для создания сайтов и веб-приложений. Включает в себя HTML- и CSS-шаблоны оформления для типографики, веб-форм, кнопок, меток, блоков навигации и прочих компонентов веб-интерфейса, включая JavaScript-расширения.



Построение макета

Для построения интерфейса не нужно создавать стили, достаточно их только скачать, взять из документации необходимый элемент и использовать.

Адаптивность страницы

Bootstrap поддерживает адаптивность. Это значит, что созданный интерфейс будет динамически подстраиваться под разрешение устройства, на котором открывается страница.

Мобильная

HEADER
A1
A2
A3
A4
A5
A6
B1
B2
B3
B4
FOOTER

Планшет

HEADER	
A1	A2
A3	A4
A5	A6
B1	B2
B3	B4
FOOTER	

Десктоп

HEADER	A1	A2	A3
A4		A5	A6
B1	B1	B1	B1
	B2	B3	B4
FOOTER			

Адаптивность поддерживается с помощью блоков `row` и `col[-разрешение]-N`:

```
<div class="row">
  <div class="col-6"></div>
</div>
```

Работа с колонками в Bootstrap

```
<div class="row"> <!-- Основной блок для работы с колонками -->
  <div class="col-2"></div>
  <div class="col-8"> <!-- Колонка шириной 8/12 для размещения
    основного контента --> </div>
  <div class="col-2"></div>
</div>
```

Компонент «карточка» в Bootstrap

```
<div class="card text-bg-primary"> <!-- Карточка для визуального  
отделения вложенных данных -->  
    <div class="card-body"> <!-- Тело карточки, внутри которого  
размещаются данные -->  
        <h2>Контактные данные:</h2>  
        <div class="row">  
            <div class="col-6">Москва</div>  
            <div class="col-6">+7 777 777 7777</div>  
        </div>  
        <div class="row">  
            <div class="col-6">Санкт-Петербург</div>  
            <div class="col-6">+7 888 888 8888</div>  
        </div>  
    </div>  
</div>
```

Формы в Bootstrap

```
<form>  
    <h2>Оставьте заявку</h2>  
    <div class="mb-3">  
        <input type="text" class="form-control" placeholder="Имя">  
    </div>  
    <div class="mb-3">  
        <input type="email" class="form-control" placeholder="Email">  
    </div>  
  
    <div class="mb-3">  
        <textarea class="form-control" placeholder="Сообщение" rows="3"></t  
extarea>  
    </div>  
    <button type="submit" class="btn btn-primary w-100">Отправить</button>  
</form>
```

Страница с подключенными элементами Bootstrap

```
<!doctype html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <title>Bootstrap demo</title>
  <link href="static/css/bootstrap.min.css" rel="stylesheet"/>
</head>
<body>
<div class="row"> <!-- Основной блок для работы с колонками -->
  <div class="col-2"></div>
  <div class="col-8"> <!-- Колонка шириной 8/12 для размещения
  основного контента -->
    <div class="row mt-5"> <!-- Дополнительный вложенный блок для
    работы с колонками -->
      <div class="col-7"> <!-- Колонка для отображения контактов -->
        <div class="card text-bg-primary"> <!-- Карточка для
        визуального отделения вложенных данных -->
          <div class="card-body"> <!-- Тело карточки, внутри
          которого размещаются данные -->
            <h2>Контактные данные:</h2>
            <div class="row">
              <div class="col-6">Москва</div>
              <div class="col-6">+7 777 777 7777</div>
            </div>
            <div class="row">
              <div class="col-6">Санкт-Петербург</div>
              <div class="col-6">+7 888 888 8888</div>
            </div>
          </div>
        </div>
      </div>
    </div>
  </div>
  <div class="col-5"> <!-- -->
    <div class="card text-bg-secondary">
      <div class="card-body">
        <form>
          <h2>Оставьте заявку</h2>
          <div class="mb-3">
            <input type="text" class="form-control" placeholder="Имя">
          </div>
        </form>
      </div>
    </div>
  </div>
</div>
```

```
<div class="mb-3">
    <input type="email" class="form-control" placeholder="Email">
</div>

<div class="mb-3">
    <textarea class="form-control" placeholder="Сообщение" rows="3"></textarea>
    </div>
    <button type="submit" class="btn btn-primary w-100">Отправить</button>
</form>
</div>
</div>
</div>
</div>
<div class="col-2"></div>
</div>
</body>
</html>
```