

HTML шаблоны

№ урока: 9 **Курс:** JavaScript Базовый

Средства обучения: Visual Studio Code
Web Browser

Обзор, цель и назначение урока

Научиться создавать разметку на основе шаблона, определенного через тэг `template` или с помощью библиотеки `mustache.js`.

Изучив материал данного занятия, учащийся сможет:

- Использовать элемент `template` для создания повторяющейся разметки на основе готового блока HTML разметки.
- Использовать библиотеку `mustache.js` для создания разметки на основе шаблона и внедрения в разметку данных из сценария.

Содержание урока

1. Элемент `template`
2. Подключение библиотеки `mustache.js` и сценарии использования
3. Примеры применения шаблонов для построения интерфейсов

Резюме

- Интерфейс бизнес-приложений часто требует вывод однотипной информации пользователю. Данные, полученные со стороны сервера, отображаются в виде таблиц или блоков повторяющейся разметки. При этом разметка может быть достаточно сложной с учетом наличия вложенности элементов, стилей и разнообразных атрибутов. Генерировать подобную разметку через JavaScript код обычно не удобно, так как внесение изменений или исправление ошибок становится затруднительным. Причина этого в том, что `document.createElement` и другие подобные методы для создания разметки, более объемны, чем HTML разметка, полученная в результате выполнения такого кода.

Для того чтобы упростить создания подобного интерфейса, можно применить подход с использованием шаблона HTML кода, который с помощью JS кода копируется, в копию вносится информация, и результат добавляется в тело документа. При этом такая манипуляция выполняется для всех значений, которые нужно отобразить пользователю. Решить подобную задачу можно с помощью элемента `template` или сторонних библиотек.

- `<template>` - элемент для хранения HTML разметки, которая с помощью JavaScript кода будет копироваться и размещаться в теле документа. Браузер полностью игнорирует содержимое данного элемента.

- Пример шаблона:

```
<template id="mytmp">
  <div>content</div>
</template>
```

Для использования шаблона в коде:

```
let template = document.querySelector("#mytmp");
let clone = template.content.cloneNode(true);
document.body.append(clone);
```

- Метод `cloneNode` создает `DocumentFragment`, в котором находится скопированное содержимое шаблона. Когда `DocumentFragment` добавляется в тело документа, добавляется только его содержимое.
- С помощью метода `querySelector` можно выполнять поиск элементов, находящихся в скопированном шаблоне. Это удобно использовать для вставки в копию информации, которая будет отображаться пользователю.
- Шаблонизатор – библиотека (программное обеспечение) использующее html шаблоны для генерации конечных html страниц. Примеры популярных шаблонизаторов – `handlebars.js`, `mustache.js`, `doT`, `pug`.
- При использовании шаблонов `mustache` в шаблоне используются специальные символы `{{ }}`, которыми определяется связь между свойствами объектов в JS коде и результирующей разметкой. Во время преобразования шаблона в разметку все символы `{{}}` будут заменены на значения соответствующих свойств.

Пример шаблона `mustache.js`:

```
<script id="template" type="x-tmpl-mustache">
  Hello {{ name }}!
</script>
```

Пример создания HTML разметки на основе шаблона:

```
function renderHello() {
  var template = document.getElementById('template').innerHTML;
  var rendered = Mustache.render(template, { name: 'Luke' });
  document.getElementById('target').innerHTML = rendered;
}
```

Для работы кода, который выполняет рендеринг (преобразование шаблона в HTML разметку), библиотека `mustache.js` должна быть подключена до этого кода.

Закрепление материала

- За что отвечает элемент `template`, какие особенности поведения разметки в данном элементе?
- Как на основе `template` сделать HTML разметку?

- Что такое шаблонизатор?
- Опишите основной принцип использования шаблонов mustache.js.

Самостоятельная деятельность учащегося

Выполните задания в директории Exercises\Tasks\009 HTML Templates. Текст задач находится в комментариях в тегах script.

Рекомендуемые ресурсы

Элемент template

<https://developer.mozilla.org/ru/docs/Web/HTML/Element/template>

Living Standard, Элемент template

<https://html.spec.whatwg.org/multipage/scripting.html#the-template-element>

Шаблонизатор mustache.js

<https://github.com/janl/mustache.js>