

Раздел

[Веб-компоненты](#)


Навигация по уроку

Использование template

Итого

Комментарии

Поделиться

[Редактировать на GitHub](#) → [Веб-компоненты](#) 10-го июля 2019

Элемент "template"

Встроенный элемент `<template>` предназначен для хранения шаблона HTML. Браузер полностью игнорирует его содержимое, проверяя лишь синтаксис, но мы можем использовать этот элемент в JavaScript, чтобы создать другие элементы.

В теории, для хранения разметки мы могли бы создать невидимый элемент в любом месте HTML. Что такого особенного в `<template>` ?

Во-первых, его содержимым может быть любой корректный HTML-код, даже такой, который обычно нуждается в специальном родителе.

К примеру, мы можем поместить сюда строку таблицы `<tr>` :

```
1 <template>
2   <tr>
3     <td>Содержимое</td>
4   </tr>
5 </template>
```

Обычно, если элемент `<tr>` мы поместим, скажем, в `<div>`, браузер обнаружит неправильную структуру DOM и «исправит» её, добавив снаружи `<table>`. Это может оказаться не тем, что мы хотели. `<template>` же оставит разметку ровно такой, какой мы её туда поместили.

Также внутри `<template>` можно поместить стили и скрипты:

```
<
1 <template>
2   <style>
3     p { font-weight: bold; }
4   </style>
5   <script>
6     alert("Привет");
7   </script>
8 </template>
```

Браузер рассматривает содержимое `<template>` как находящееся «вне документа»: стили, определённые в нём, не применяются, скрипты не выполняются, `<video autoplay>` не запустится и т.д.

Содержимое оживёт (скрипт выполнится), когда мы поместим его в нужное нам место.

Использование template

Содержимое шаблона доступно по его свойству `content` в качестве [DocumentFragment](#) – особый тип DOM-узла.

Можно обращаться с ним так же, как и с любыми другими DOM-узлами, за исключением одной особенности: когда мы его куда-то вставляем, то в это место вставляется не он сам, а его дети.

Пример:

```
1 <template id="tpl">
2   <script>
3     alert("Привет");
4   </script>
5   <div class="message">Привет, Мир!</div>
6 </template>
7
8 <script>
```



Раздел

Веб-компоненты

Навигация по уроку

Использование template

Итого

Комментарии

Поделиться



Редактировать на GitHub



```
9   let elem = document.createElement('div');
10
11   // Клонировать содержимое шаблона для того, чтобы переиспользовать
12   elem.append(tmpl.content.cloneNode(true));
13
14   document.body.append(elem);
15   // Сейчас скрипт из <template> выполнится
16 </script>
```

Давайте перепишем пример Shadow DOM из прошлой главы учебника с помощью <template>:

```
1 <template id="tpl">
2   <style> p { font-weight: bold; } </style>
3   <p id="message"></p>
4 </template>
5
6 <div id="elem">Нажми на меня</div>
7
8 <script>
9   elem.onclick = function() {
10     elem.attachShadow({mode: 'open'});
11
12     elem.shadowRoot.append(tmpl.content.cloneNode(true))
13
14     elem.shadowRoot.getElementById('message').innerHTML
15   };
16 </script>
```

Нажми на меня

Когда мы клонируем и вставляем `tmpl.content` в строке (*), то, так как это `DocumentFragment`, вместо него вставляются его потомки (`<style>`, `<p>`).

Именно они и формируют теневой DOM:

```
1 <div id="elem">
2   #shadow-root
3     <style> p { font-weight: bold; } </style>
4     <p id="message"></p>
5 </div>
```

Итого

Подводим итоги:

- Содержимым <template> может быть любой синтаксически корректный HTML.
- Содержимое <template> считается находящимся «вне документа», поэтому оно ни на что не влияет.
- Мы можем получить доступ к `template.content` из JavaScript, клонировать его и переиспользовать в новом компоненте.

Элемент <template> уникальный по следующим причинам:

- Браузер проверяет правильность HTML-синтаксиса в нём (в отличие от строк в скриптах).
- ...При этом позволяет использовать любые HTML-теги, даже те, которые без соответствующей обёртки не используются (например `<tr>`).
- Его содержимое оживает (скрипты выполняются, `<video autoplay>` проигрывается и т. д.), когда помещается в документ.

Элемент <template> не поддерживает итерацию, связывания данных или подстановки переменных. Однако эти возможности можно реализовать поверх него.

Раздел

[Веб-компоненты](#)

Навигация по уроку

Использование template

Итого

Комментарии

Поделиться



[Редактировать на GitHub](#)



Комментарии

перед тем как писать...

© 2007–2020 Илья Кантор | [о проекте](#) | [связаться с нами](#) | [пользовательское соглашение](#) | [политика конфи](#)

