



HTML и CSS. Уровень 3.

Продвинутые методологии и инструменты верстки



Модуль 3. Компонентная модель верстки

Web-components

(это *черновик* набора стандартов)

web-components

Компонентная модель состоит из четырёх составляющих, которые, при совместном использовании, позволяют разработчикам web-приложений создавать виджеты с богатыми визуальными возможностями, легкие в разработке и повторном использовании.

Стандарт web-components

- Templates (шаблоны)
- Custom Elements (пользовательские элементы)
- Shadow DOM (теневая модель DOM-a)
- Imports (HTML-импорты)

Templates

В Web-components шаблоны - это фрагменты HTML, которые программист собирается использовать в будущем.

В основе шаблонов лежит элемент `<template>`.

Содержимое элемента `<template>` анализируется парсером, но оно неактивно: скрипты не запускаются, изображения не загружаются.

Templates

```
<template>  
  <h2>Flower</h2>  
    
</template>
```

...

```
<script>  
  var temp = document.querySelector("template");  
  var clon = temp.content.cloneNode(true);  
  document.body.appendChild(clon);  
</script>
```

Custom Elements

Когда-нибудь мечтали о том, чтобы в HTML был тег `<menu>` или `<user-info>`?

Пользовательские элементы позволяют создавать и определять API собственных HTML элементов, и тем самым дают возможность создавать полноценные виджеты вроде `<audio>`, `<video>` или `<input type="date">`.

Согласно стандарту, пользовательские элементы **должны содержать дефис в своём названии**. По умолчанию они наследуют `HTMLElement`. Таким образом, когда браузер натывается на разметку вида `<my-super-element>`, он парсит его как `HTMLElement`. В случае `<mysuperelement>`, результат будет `HTMLUnknownElement`.

Custom Elements

`<my-element></my-element>`

...

`<script>`

```
var proto = Object.create(HTMLElement.prototype);
proto.createdCallback = function() {
  this.innerHTML = "<h1>Hello!</h1>"
};
var el = document.registerElement("my-element", {
  prototype: proto
});
```

`</script>`

Shadow DOM

Shadow DOM (теневая модель документа) — часть документа, реализующая инкапсуляцию в DOM дереве. Является частью документа и встраивается непосредственно внутрь страницы.

Позволяет изменять внутреннее представление HTML элементов, оставляя внешнее представление неизменным. Отличный пример — элементы `<audio>` и `<video>`. В коде мы размещаем один тег, а браузер отображает несколько элементов (слайдеры, кнопки, окно проигрывателя).

Стили, определённые в Shadow DOM с помощью `<style>`, не распространяются на родительский документ.

HTML Imports

HTML импорт - это способ вставки на страницу других HTML страниц.

До сих пор импорт html страниц поддерживается только Google Chrome!

```
<link rel="import" href="myfile.html">
```

Polymer

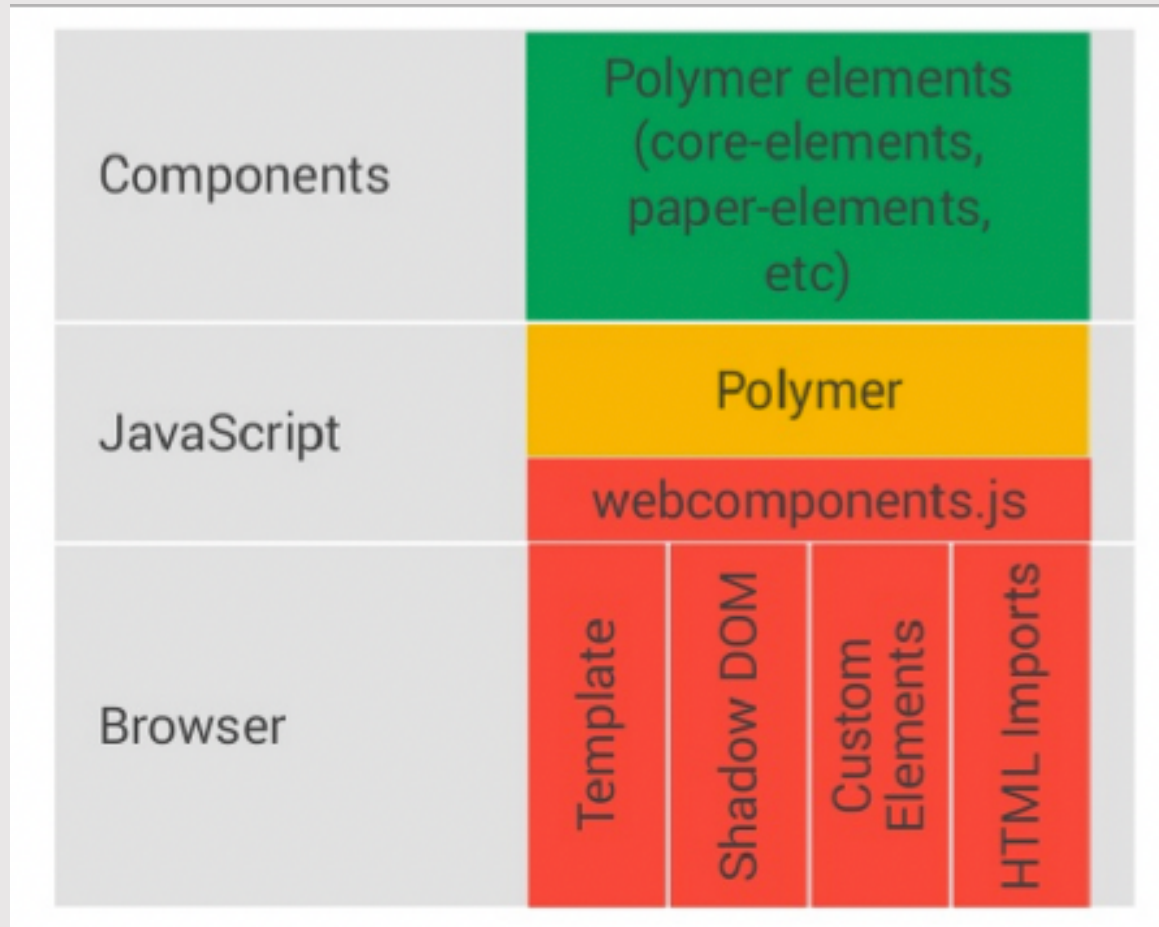
(библиотека для создания и использования Web-компонент)

Polymer

Не все браузеры поддерживают web-components.

Благодаря библиотеке Polymer, разработанной в Google, мы можем это исправить.

Polymer обеспечивает набор Polyfills, который позволяет нам использовать веб-компоненты в несовместимых браузерах.



Установка Plymer

➤ Установить Node.js <https://nodejs.org/en/download/>

➤ Установить git <https://git-scm.com/download/>

➤ Установить Polymer-cli

```
npm install -g polymer-cli  
polymer init  
polymer serve
```

www.polymer-project.org