

# Компоненты во Vue.js, часть 2

---

JS  
COURSE  
ORT DNIPRO

---

ORTDNIPRO.ORG/JS

# Vue.js – фреймворк для разработки SPA

## The Progressive JavaScript Framework

An approachable, performant and versatile framework for building web user interfaces.



Why Vue

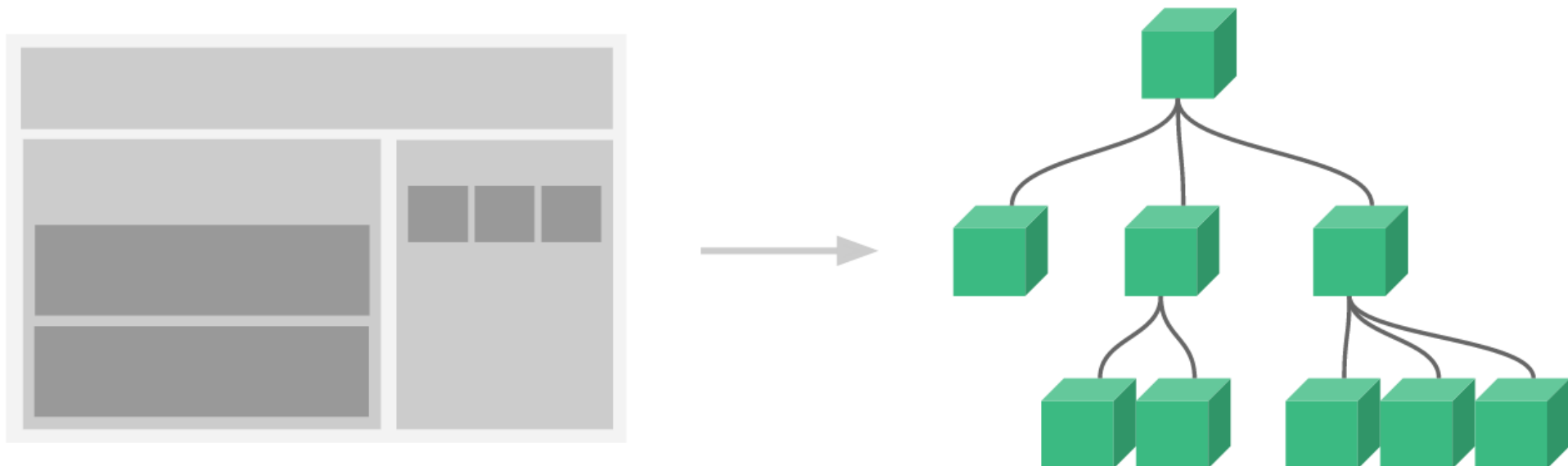
Get Started →

Install

<https://ua.vuejs.org/>

# 1. Компоненты во Vue.js

# Компоненты – разделение приложения на блоки



Идея **компонентов Vue.js** аналогична идеи **компонентов** в библиотеке **Bootstrap** – отдельные, относительно независимые, части приложения которые можно переиспользовать.

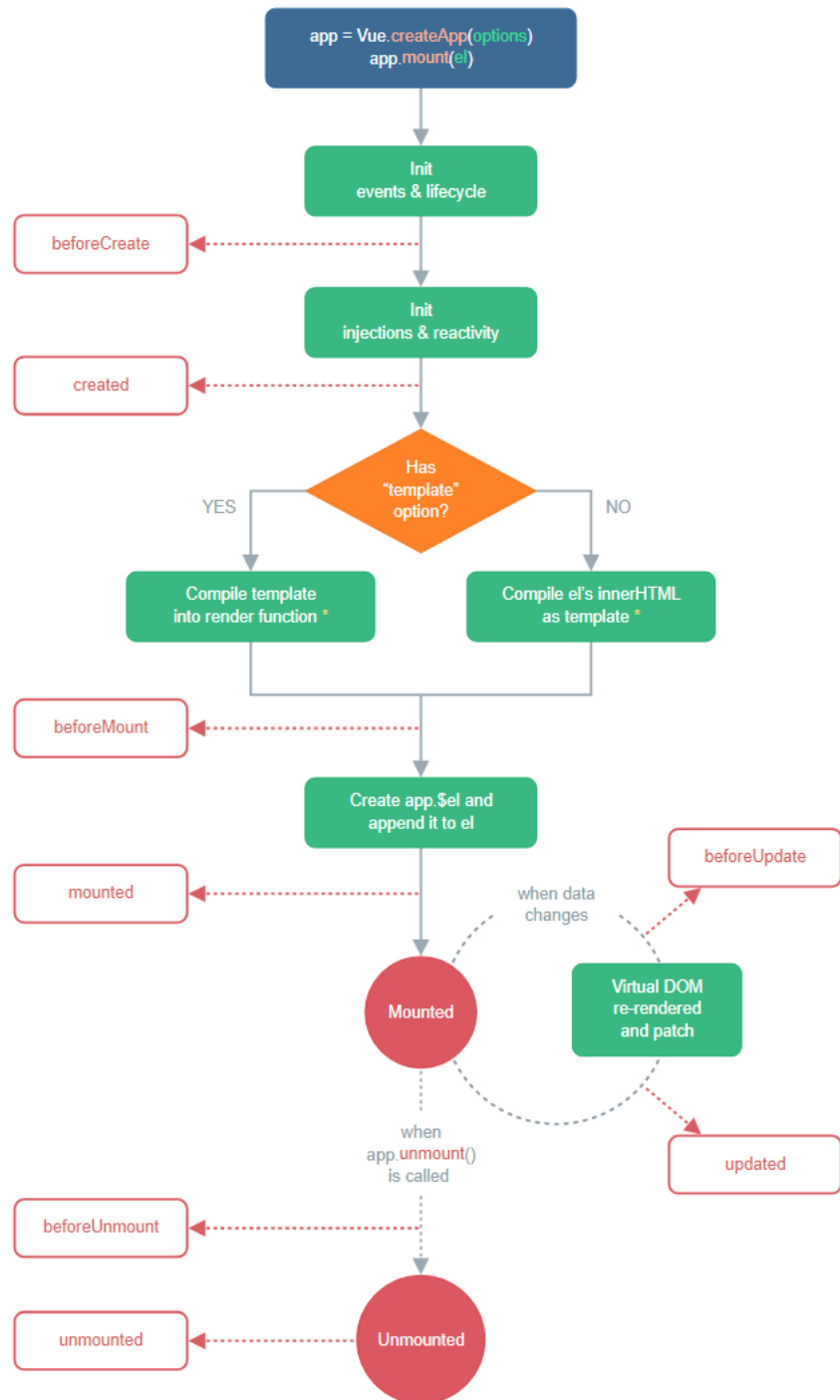
<https://ua.vuejs.org/guide/essentials/component-basics.html>

## **2. Жизненный цикл компонента (приложения)**

# Жизненный цикл компонента (и приложения)

Этапы создание и функционирования каждого компонента (и приложения), представляют интерес тем, что можно связать свой код с тем или иным хуком жизненного цикла. Одним из самых полезных является хук **onMounted()** позволяющий запустить код когда компонент перемонтирован в документ и отображается пользователю (что полезно для запуска AJAX-запросов и прочих «долгих» действий).

<https://ua.vuejs.org/guide/essentials/lifecycle.html#lifecycle-diagram>



# 3. «Загрузка» приложения

# «Загрузка» приложения и директива **v-cloak**

```
13 <body>
14   <div id="app" v-cloak>
15     <h1>{{title}}</h1>
16   </div>
17   <div id='app-loading' >
18     <i class="fas fa-spinner fa-spin"></i>
19   </div>
20
21 </body>
```

```
1
2 [v-cloak] {
3   display: none;
4 }
5
6 #app-loading {
7   display: none;
8 }
9
10 [v-cloak] + #app-loading {
11   display: block;
12 }
13
14
```

Воспользуйтесь шаблоном в репозитории занятия [./src/app-load-imitation](#)

При помощи директивы **v-cloak** и небольшого фрагмента стилей есть возможность скрыть разметку приложения, до того момента пока оно не будет смонтировано и готово к работе. Подробнее: <https://ua.vuejs.org/api/built-in-directives.html#v-cloak>



**На следующем занятии**

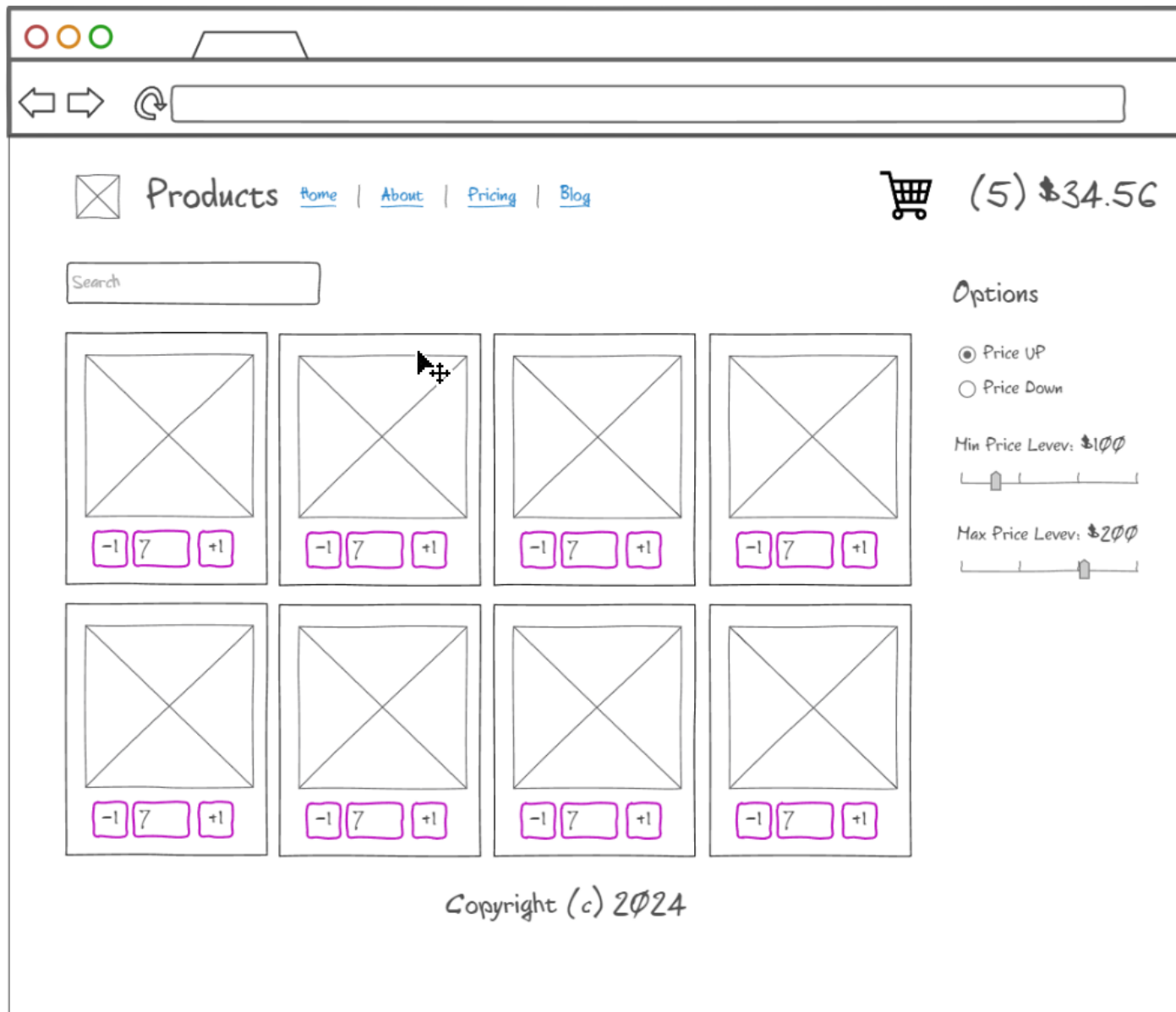
На следующем занятии

Vite

Система сборки Vue-приложений

**Домашнее задание**  
**/сделать**

# Домашнее задание #Т.1



Воспользуйтесь разделом API <https://fakestoreapi.com/products> для формирования страницы со списком товаров. Реализуйте вывод списка товаров, а также обеспечьте работу **сортировки** и **фильтрации** по цене, **поиска** которые должны работать **одновременно**. **Изначально** (после загрузки страницы) должен выводиться список в той же последовательности, в которой пришли данные от API.

На каждой карточке товара должен быть **счётчик количества**. А в верхней части страницы указание **сколько суммарно единиц товаров** (всех) выбрал пользователь. И какая **суммарная стоимость** всех выбранных товаров. Всё это должно работать в динамике. **Vue** и **компоненты** вам в помощь.