

Взаимодействие между компонентами

Урок 6



Кадочников Алексей

Frontend-разработчик

- ☀ Веб-разработчик со стажем более 9 лет
- ☀ Преподаватель GeekBrains с 2015 года
- ☀ Автор курсов по Frontend на портале Geekbrains
- ☀ Работал в таких компаниях как VK и Wizard-C



План курса

1

Введение во Vue.js

2

Методы и работа с данными

3

Компоненты

4

Vue CLI

5

Встроенные директивы и работа с данными компонента

6

Взаимодействие между компонентами

7





Управление данными приложения с Vuex

8

Навигация в приложении Vue-Router



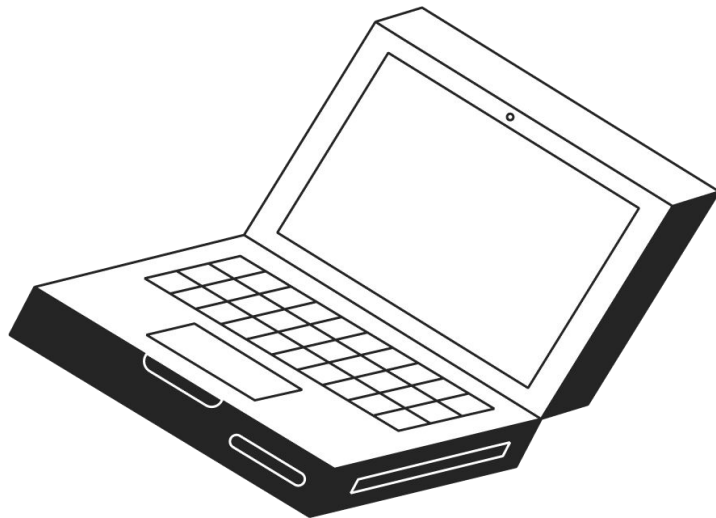
Что будет на уроке сегодня

-  Введение
-  Жизненный цикл
-  Практическая часть
-  Итоги



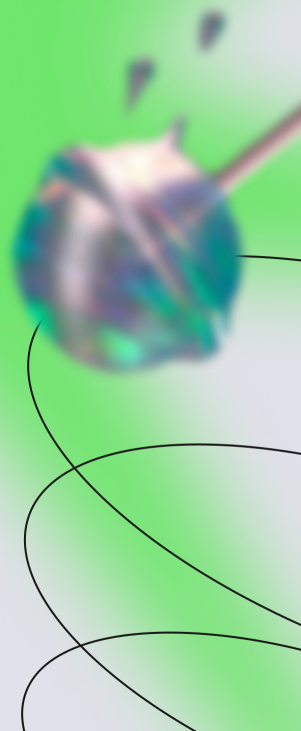
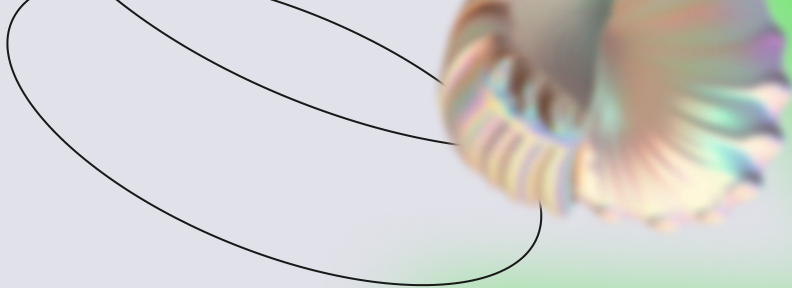
Введение

Компонент – логически независимая часть приложения, которая отвечает за обработку и визуализацию конкретной задачи. Композиция таких независимых частей должна представлять из себя рабочий функционал.



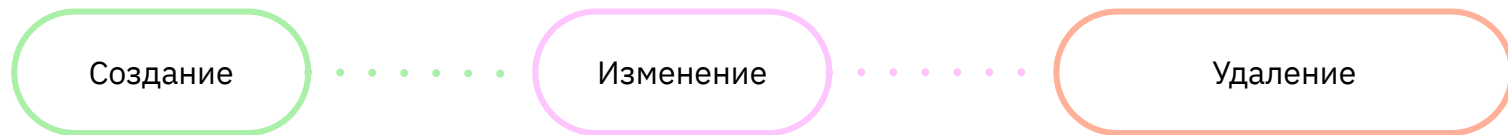


Жизненный цикл





Жизненный цикл





Жизненный цикл

- beforeCreate
- created
- beforeMount
- mounted
- beforeUpdate
- updated
- beforeDestroy
- destroyed



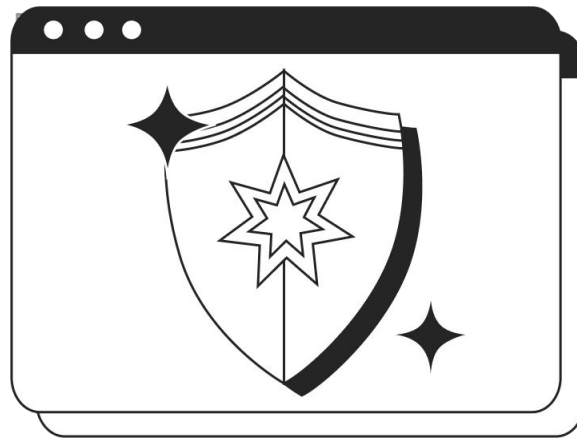
Создаем
приложение





\$emit – генерация событий и передача данных

Любой компонент может в себе генерировать событие, на которое родительский компонент может подписаться. С помощью события дочерний компонент сообщает родителю, что произошло какое-то событие, и вместе с этим сообщением можно передать любые данные.






Итоги



Подробно разобрали что такое жизненный цикл

Спасибо 
за внимание

