**Лекция 2: Создание и управление компонентами Vue**

**1. Что такое компоненты Vue?**

Компоненты — это переиспользуемые блоки интерфейса, которые помогают разбивать приложение на независимые логические части. Каждый компонент может содержать свою разметку, стили и логику.

**Преимущества компонентного подхода:**

* Улучшает читаемость и поддержку кода.
* Обеспечивает переиспользуемость кода.
* Упрощает масштабирование приложения.

**2. Создание компонента с использованием SFC (Single File Components)**

**2.1 Пример простого компонента**

Создадим компонент HelloWorld.vue:

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

**2.2 Подключение компонента в приложении**

Файл App.vue:

Изображение выглядит как текст, снимок экрана, программное обеспечение, дисплей

Автоматически созданное описание

**3. Свойства (Props) и события**

**3.1 Передача данных через Props**

Создадим компонент UserCard.vue:

Изображение выглядит как текст, снимок экрана, программное обеспечение, Шрифт

Автоматически созданное описание

**3.2 Обработка событий с помощью $emit**

Создадим компонент ButtonCounter.vue:

**Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание**

**4. Слоты (Slots)**

Создадим компонент AlertBox.vue:

Изображение выглядит как текст, снимок экрана, программное обеспечение, дисплей

Автоматически созданное описание

**5. Жизненный цикл компонентов**

Создадим компонент LifecycleDemo.vue:

**Изображение выглядит как текст, снимок экрана, дисплей, программное обеспечение

Автоматически созданное описание**

Использование SFC делает разработку на Vue.js более структурированной и удобной. В следующей лекции мы рассмотрим, как работает реактивность и управление состоянием данных в Vue.