

Основы React

№ урока: 3 Курс: React Essential

Средства обучения: Браузер Chrome, редактор кода VS Code или любой другой.

Обзор, цель и назначение урока

Познакомиться с подвидами компонентов, узнать, что такое state в React компонентах, научиться добавлять события к элементам. Научиться работать со списками.

Изучив материал данного занятия, учащийся сможет

- Базово использовать state в компонентах.
- Добавлять события к элементам.
- Работать со списками в React.

Содержание урока

1. Виды компонентов (stateless, stateful)
2. Знакомство с State
3. Добавление событий
4. Работа со списками

Резюме

- Компоненты в React, содержащие внутреннее состояние (state), называются – **Stateful**, без него — **Stateless**.
- Классовые компоненты всегда должны вызывать базовый конструктор с аргументом props. Чтобы получить доступ к state нужно вызвать this.state. Чтобы изменить state, нужно вызвать метод this.setState.
- **Обработка событий в React-элементах.** События в React именуются в стиле camelCase вместо нижнего регистра. С JSX вы передаёте функцию как обработчик события вместо строки. Ещё одно отличие — в React нельзя предотвратить работу обработчика события по умолчанию, вернув false. Нужно явно вызвать preventDefault.
- **Synthetic event** – ваши обработчики событий получают экземпляр SyntheticEvent, это кроссбраузерная обёртка над нативным экземпляром события. У неё такой же интерфейс, как и у нативного события, включая методы stopPropagation() и preventDefault(). Эта обёртка помогает событиям работать одинаково во всех браузерах. React создает свое синтетическое событие. При использовании React обычно не нужно вызывать addEventListener, чтобы добавить обработчики в DOM-элемент после его создания. Вместо этого добавьте обработчик сразу после того, как элемент отрендерился.
- Если вам всё-таки нужно получить нативное браузерное событие, обратитесь к атрибуту nativeEvent.
- **События в классовых компонентах.** При обращении к this в JSX-колбэках необходимо учитывать, что методы класса в JavaScript по умолчанию не привязаны к контексту. Если вы забудете привязать метод this.handleClick и передать его в onClick, значение this будет undefined в момент вызова функции.

- **Списки.** Вы можете создать коллекцию элементов и встроить её в JSX с помощью фигурных скобок {}. У каждого элемента массива должен быть «ключ» (key).

Ключ — это специальный строковый атрибут, который нужно указывать при создании списка элементов.

- Если вы опустите ключ для элемента в списке, то React по умолчанию будет использовать индексы как ключи. Ключи нужно определять непосредственно внутри массивов. Как правило, элементам внутри map() нужны ключи. Ключи внутри массива должны быть уникальными только среди своих соседних элементов. Им не нужно быть уникальными глобально. Можно использовать один и тот же ключ в двух разных массивах.

Закрепление материала

- Что такое Synthetic event?
- Как получить доступ к браузерному событию?
- Зачем нужно определять атрибут key в списках?
- Что такое State в React компонентах?

Дополнительное задание

Задание

Создать приложение с помощью Create React App, вывести на странице текст «hello world».

Самостоятельная деятельность учащегося

Задание 1

Выучите основные понятия, рассмотренные на уроке.

Задание 2

Создайте компонент, который будет выводить список имен. Имена не должны повторяться.

Задание 3

Создайте компонент, который будет выводить 2 кнопки и счетчик. По клику на кнопку «+» счетчик увеличивается, по клику на кнопку «-» - уменьшается.

Рекомендуемые ресурсы

<https://ru.reactjs.org/docs/state-and-lifecycle>

<https://ru.reactjs.org/docs/handling-events>

<https://ru.reactjs.org/docs/events>

<https://ru.reactjs.org/docs/lists-and-keys>

Function.prototype.bind()

https://developer.mozilla.org/ru/docs/Web/JavaScript/Reference/Global_Objects/Function/bind

Встраивание выражений в JSX

<https://ru.reactjs.org/docs/introducing-jsx.html#embedding-expressions-in-jsx>