React — JavaScript-библиотека с открытым исходным кодом для разработки пользовательских интерфейсов.

JSX — это XML-подобное расширение синтаксиса языка JavaScript.

Babel — это бесплатный транскомпилятор JavaScript с открытым исходным кодом, который в основном используется для преобразования кода ECMAScript 2015+ в обратно совместимый код JavaScript, который может выполняться старыми движками JavaScript.

Create React App (CRA) является мощным инструментом для быстрого создания и разработки React приложений. Однако, в некоторых случаях, работа с относительными путями может стать неудобной и затруднить поддержку кода.

Single Page Application – сокращенно **SPA**, в переводе на русский язык означает “Приложение одной страницы”.  **SPA** — это web-приложение, размещенное на одной web-странице, которая загружает весь необходимый код вместе с загрузкой самой страницы.

render - это метод жизненного цикла компонента, который вызывается для отображения компонента на странице. Он возвращает React-элемент, представляющий содержимое компонента. Метод render определяет, что должно быть отображено на экране в ответ на изменение состояния или свойств компонента.

Компоненты React – это самодостаточные элементы, которые можно использовать на странице любое количество раз. Во многом компоненты ведут себя как обычные JavaSript функции. Они принимают входные данные, так называемые props, и возвращают React-элементы, описывающие то, что мы хотим увидеть на экране.

Принцип единственной ответственности: каждый компонент должен заниматься какой-то одной задачей.

Архитектура Redux вращается вокруг строго однонаправленного потока данных (Data Flow). Это значит, что все данные в приложении следуют одному паттерну жизненного цикла, делая логику вашего приложения более предсказуемой и легкой для понимания.

Неизменяемым (англ. immutable) называется объект, состояние которого не может быть изменено после создания. Результатом любой модификации такого объекта всегда будет новый объект, при этом старый объект не изменится.

Чистая функция — это функция, которая при одинаковых значениях аргументов всегда возвращает одинаковые значения и не имеет наблюдаемых побочных эффектов.

Компоненты, созданные как функции, называются функциональными. Они принимают объект со свойствами как первый аргумент и также начинаются с большой буквы.

useState возвращает массив с двумя элементами, который содержит: текущее значение состояния и функцию для его обновления. Эту функцию можно использовать где угодно, например, в обработчике событий. Она схожа с this.

Хуки — нововведение в React 16.8, которое позволяет использовать состояние и другие возможности React без написания классов. Примеры хуков: useState, useEffect.

**Что же делает useEffect?** Используя этот хук, вы говорите React сделать что-то после рендера. React запомнит функцию (то есть «эффект»), которую вы передали и вызовет её после того, как внесёт все изменения в DOM. В этом эффекте мы устанавливаем заголовок документа, но мы также можем выполнить запрос данных или вызвать какой-нибудь императивный API.