Парадигмы программирования:

1. Функциональное программирование
2. Логическое программирование
3. Математическое программирование

Что такое декларативный подход?

Вы общаетесь любым компонентом на прямую.

React позволяет работать именно с тем DOM элементом, с которым мы хотим.

Функциональное программирование:

1. Заменяет наследование композицией

2. Использует функции

Hooks – функции.

Экосистема React:

1. Сборка (WebPack, rollup) – браузер загружает наш js одним файлом, хотя код мы пишем во множестве файлов + подключенные библиотеки. Сборщик нужен для того, чтобы собрать всё это воединно и создать один bundle. У Webpack есть сложная конфигурация. Часто на собесах может быть вопрос «А приходилось ли вам настраивать Webpack?”
2. Транспилятор – Babel, TypeScript. Это то что позволяет нам писать современный код, но превращать его в старый. Так как не все браузеры поддерживает разные фичи новых Ecmascript.
3. Статический анализатор кода – TypeScript, flow.
4. React – это библиотека. Вес = 3 кб.
5. State manager – Redux.
6. Middleware и утилиты для state manager – Redux-Thunk, Redux-Saga, Redux-Dev Tools
7. Роутинг - React Router
8. Тесты – Jasmine, Mocha, AVA, Jest
9. Axios, React click outsider, Storybook, Classnames,