

# 공식문서로 디테일잡기 고급 마무리

1 지금까지 알게 된 것

## 정리

### Hooks

Hooks 등장

Class의 단점 보완 / 재사용성 강화

Hook 사용 규칙

최상위 / 함수형 컴포넌트 / custom Hook

Class의 state

혹을 먼저 배웠기에 고민 할 필요 X

## 정리

### Hooks 2

useEffect

데이터 fetch / 구독 / Dom 수정

clean up

구독과 구독해지를 한 공간에서

dependency array

필요한 변경시에만 effect 실행

## 정리

### Hooks 3

Custom Hook

컴포넌트들에 반복되는 Hooks 묶기

다양한 Hooks

필요한 타이밍에 참고해서 사용

## 정리

### Composition

Composition

컴포넌트에 컴포넌트 담기

담는 방법

Children / Custom

## 정리

### Composition 2

typeof

type check

확장성

다양한 상황을 품을 수 있도록

## 정리

HOC

HOC



함수를 받아서 함수를 리턴

## 정리

### Memoization





## 정리

### Context

컴포넌트 트리 제약

Props drilling의 한계 해소

재사용성

Context를 사용하면 재사용하기 어려움

API

createContext / Provider / Consumer

useContext

Consumer 대체

## 정리

### Portals

createPortal

부모 컴포넌트 DOM 트리로부터 벗어나기

이벤트

Portal에 있더라도 Event는 트리로 전파

## 정리

### Render Props

render props

무엇을 렌더링할 지 알려주는 함수

render일 필요

없음, children도 되고 뭐든 됨

PureComponent

props, state 비교하여 성능 최적화

## 정리

### PropTypes

개발 모드에서만 동작

유효하지 않은 prop에 대한 경고

custom

RegExp 등으로 사용자 정의 가능

children 제한

원래 제약없던 갯수 제약 가능

## 정리

### Reconciliation

루트부터 비교

무엇을 렌더링할 지 알려주는 함수

트리를 파괴

부모가 바뀌었다면 트리를 버린다

Keys

자식 재귀 처리 시 효율성 확보

## 정리

### Reconciliation 2



## 정리

### React Dev Tools

개발자 도구 확장

React에서 확인하고 싶은 것들

성능 측정

Profiler

# 마무리

## 2 결들임



## 리액트로 사고하기

(<https://ko.reactjs.org/docs/thinking-in-react.html>)

컴포넌트를 나누고 / 데이터를 가져오고 / 상태를 관리하고  
각각 기능을 구현하고 / 디자인을 입히고

공식문서 보면서 무언가를 이해해보기

리엑트는 익숙해졌을지 모르지만,

문서를 보고 라이브러리를 익히는 건 익숙해지려면 **반복** / 연습

20여개의 라이브러리를 보면서 반복 / **연습** 해보자