

# useContext

- 컴포넌트 호출 tree
  - 호출이 넓고 깊게 이루어질 수 있음
  - 끝부분의 컴포넌트 에서 사용되는 내용을 파라미터(props) 로 전달하는 경우, 중간 부분에서는 이 내용을 사용하지 않더라도 아래 부분 전달을 위해 이를 명시적으로 파라미터에 표현해 주어야 한다.
  - 코드의 가독성에 불편함을 유발함
- useContext 이용
  - props를 **명시적으로 패스할 필요없이**, context를 설정하여, tree의 모든 레벨에서 공급자와 사용자가 이 내용을 공유하도록 하는 기능.

# Context 사용 예

- Context는 리액트 컴포넌트 tree를 위한 **전역적 데이터 공유**를 가능케 함
- Examples
  - the current authenticated user
  - UI Theme
  - preferred language (locale)

# Context의 사용이 정말 필요한가?

- Context의 전역적 특성은 컴포넌트의 reuse를 어렵게 할 수 있으므로,
- 꼭 필요하지 않는 경우라면 사용하지 않을 것을 권고함.
- Context 대신 컴포넌트 합성 (**component composition**) 을 이용함으로써 props 패스를 대신할 수 있음.

# UseContextEx1.jsx 의 컴포넌트 흐름

컴포넌트 호출 밑으로 context가 흐르도록 구성하고, 각 컴포넌트는 필요에 따라 context 를 뽑아서 사용

