7/7

1. **서버사이드 렌더링이란 무엇인가**

서버사이드 렌더링이란, 페이지 렌더링 시에 서버 쪽에서 html을 구성하여 브라우저로 전송해주는 렌더링 방식이다. 서버 쪽에서 매번 html을 구성하여 브라우저로 전송해주면 브라우저가 css를 분석/로딩하고 js를 덧붙인다. 클라이언트 사이드 렌더링 방식에서는, 초기 렌더링 시에 브라우저 측에서 JS가 처리해야 할 일이 많을 경우 전체 페이지 로딩속도가 느려 질 수 있는데, 서버사이드 렌더링을 하면 서버 쪽에서 HTML을 만들어서 브라우저로 전송해주기 때문에, 브라우저 쪽에서 JS가 다 로딩되지 않아도 화면 구성이 이루어진다(단 interactive 기능까지 완전히 로딩되기까지는 시간이 걸린다.). 사이트 렌더링 시에 서버 쪽의 부담이 좀 더 커진다는 단점이 있다.

1. **SPA란 무엇이고 어떤 장점이 있는가.**

SPA는 signle page application를 뜻한다. 브라우저가 서버로부터 받은 index.html 안의html,css,JS를 직접 분석/로딩하고, 사용자는 페이지 리로딩이 아닌 api를 통해 SPA를 이용한다. 페이지 리로딩 없이 한 페이지(index.html)안에서 많은 기능을 수행하게끔 하기 때문에 로딩 시간이 빠르고, 프론트엔드와 백엔드의 분업이 가능하므로 서버 쪽의 부담이 덜하다는 장점이 있다.

**3. 리액트에 사용되는 디자인패턴은 어떤 것들이 있는가. 각각 설명하시오.**

\*좋은 디자인 패턴이란? – **유지/보수성, 상호소통성(협업시) 이 좋아야 함.** 리액트에서는 많은 디자인 패턴들이 렌더링과 관련된 이슈들을 해결해주고, props전달을 쉽게 해준다. Render prop이란 함수형태의 props를 사용해서 컴포넌트들끼리 코드를 주고받는것이다. 별도의 컴포넌트를 만들지 않고 컴포넌트 간에 state를 공유할 수 있게끔 해준다. \*

1. HOC(higher order compoment)

: 컴포넌트를 인자로 받아서 새로운 컴포넌트를 리턴하는 것.

(SPA) <https://www.youtube.com/watch?v=iE29lbjbow0&list=PLpO7kx5DnyIFyT_q9gVtg4CzYU0EYcI5C&index=17>  
(서버사이드 렌더링) <https://www.youtube.com/watch?v=Y2spCNZDt84>  
(리액트 디자인 패턴)

<https://www.youtube.com/watch?v=NazjKgJp7sQ>  
<https://www.youtube.com/watch?v=N1yqq9Az_XY>

\*출처\*

1. Presentational & container component

: 컴포넌트를 presentation(보여지는 부분)과 container(동작하는 부분)으로

나누는 것.

* UI의 App들을 더 쉽게 파악할 수 있다.
* 여러 App에 걸쳐서 사용 가능하므로 재사용성이 좋다.
* 지저분한 컴포넌트 정리하는 효과.