7월 2일 기술면접 문제

1. 컴포넌트 설계가 갖는 장점은 무엇인가. 컴포넌트란 무엇인가. 컴포넌트 설계는 어떻게 해야 하는가. 컴포넌트의 리랜더링은 언제 이루어지는가.

- 컴포넌트: 개념적으로, 컴포넌트는 자바스크립트 함수입니다. 컴포넌트는 임의의 props라 불리는 인풋을 받아서 화면에 무엇이 보여질지 기술(describing)하는 리액트 엘리먼트를 반환합니다.

- 컴포넌트 기반 개발(component-based development, CBD): 컴포넌트 기반 개발은 기존의 시스템이나 소프트웨어를 구성하는 컴포넌트를 조립해서 하나의 새로운 응용 프로그램을 만드는 소프트웨어 개발방법론이다. 기업들은 쇼핑바구니, 사용자 인증, 검색엔진, 카탈로그 등 상업적으로 이용 가능한 컴포넌트를 결합하여 그들의 전자상거래 응용 프로그램을 개발하는 컴포넌트 기반 개발을 사용한다.

- 컴포넌트 기반 개발의 장점: CBD는 개발 방법론의 핵심이 되는 '재사용성', '생산성' 등을 향상시키며 요구사항이 수시로 바뀌는 현대의 서비스 개발에 특화되어 빠르게 서비스를 피벗하거나 사용자가 급등할 수 있는 환경을 고려해 유연한 환경을 조성하는데 도움이 됩니다.

- 컴포넌트 리렌더링: 리액트가 렌더링을 수행하는 과정은 다음과 같다.

1. Props가 변경되었을 때

2. State가 변경되었을 때

3. forceUpdate() 를 실행하였을 때

4. 부모 컴포넌트가 렌더링되었을 때

1~3번의 과정을 통해 컴포넌트가 렌더링될 때, 자식 컴포넌트 또한 같은 과정으로 렌더링이 진행됩니다. 하지만 컴포넌트에서 렌더링 결과에 전혀 영향을 미치지 않는 변경사항이 발생하게 된다면, 불필요한 렌더링이 발생 하므로 성능 손실이 발생합니다. 이는 렌더링에서 수행하는 로직이 많을 수록, 많은 컴포넌트를 출력할 수록 손실은 배가 됩니다.

출처:

<https://velog.io/@rjs1197/프론트엔드-React를-중심으로-컴포넌트-기반-개발-방법을-알아보자2019>

https://velog.io/@jakeseo\_me/리액트-공식문서로-배워보자-4-컴포넌트와-Props

<https://link.medium.com/g6HuR13aB7>

2. 리액트와 뷰의 차이점과 공통점은 무엇인가.

- 공통점:

1. 컴포넌트 기반 SPA(Single Page Application: 단일 페이지 어플리케이션) 라이브러리 라는 점이다. 컴포넌트 단위로 변화사항을 감지하고 렌더링이 필요한 부분만 바꾸어 주는 기능은 양쪽의 프레임워크가 목적론적으로나 방법론적으로나 동일하다. 둘 다 virtual DOM을 기반으로 하며, tree알고리즘을 통해 변화사항을 찾아낸다. 때문에 몇 가지 제한사항 같은 것들도 서로 닮은 점이 있다. 가령 각 컴포넌트의 HTML에 해당하는 코드는 단일 노드로 감싸져 있어야 한다든지, 목록을 출력할 때 각각의 항목이 구분될 수 있는 고유한 문자열을 할당해주어야 한다든지 (key props) 등.

2. 둘 다 컴포넌트 단위로 코드를 작성하는 친구들이다보니, 컴포넌트의 기본적인 구성요소도 겹치는 것들이 있다. 대표적인 게 props와 state, 그리고 lifecycle이다. 세부적인 명칭이나 사용법 등은 물론 조금씩 다르긴 하지만, 지향점은 같다.

3. SPA를 통해 여러 페이지를 보여주기 위한 라우터 역시 리액트와 뷰 둘 다 존재한다. 다만, 차이점이라면 리액트는 3rd party library에서 그걸 지원해주고 있고, 그 상황을 지켜보면서 만들어진 뷰는 공식 라이브러리도 함께 만들어서 내놨다는 점 정도이다. 전역 상태관리 도구 역시 비슷하다. 리액트는 제 3자에 의해 redux 같은 게 만들어졌고, 후발주자인 vue는 시작할 때부터 redux와 redux-saga가 합쳐진 형태의 전역 상태관리 도구인 vuex를 만들어서 공식으로 내놨다.

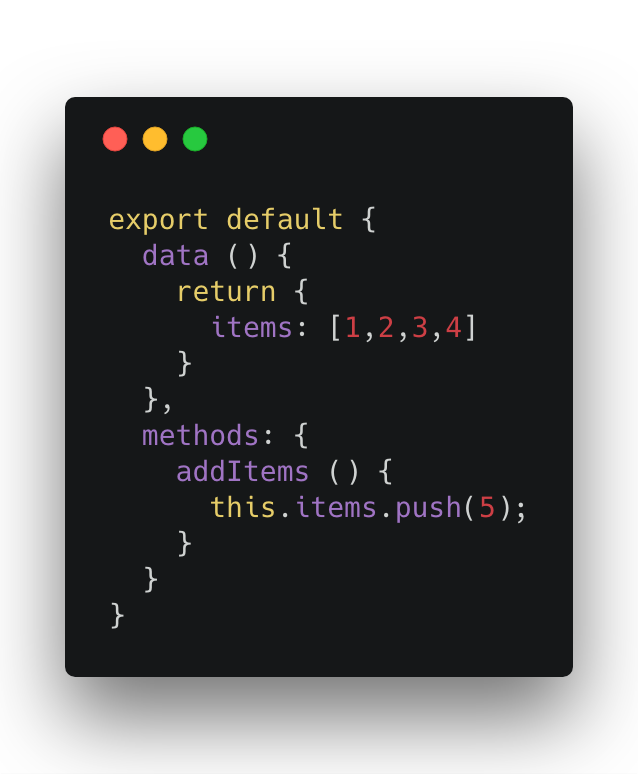
4. 둘 다 기본 프로젝트 구조를 만들어주는 툴을 cli로 제공하고 있다. webpack 기반으로 번들링 해주는 기능 역시 마찬가지로 존재한다.

5. 둘 다 모바일 앱 개발용 도구를 지원한다. 이 부분은 신기하게도 앞서 말했던 부분과 정 반대의 관계인 게 있다. 리액트 네이티브는 페이스북에서 공식적으로 지원하는 라이브러리이며, 뷰 네이티브는 서드파티 라이브러리이다. 아무래도 페이스북은 리액트로 모바일 앱을 직접적으로 개발하고 있다 보니, 다른 건 서드파티한테 맡기더라도 이런 건 직접 하는 것으로 보인다.

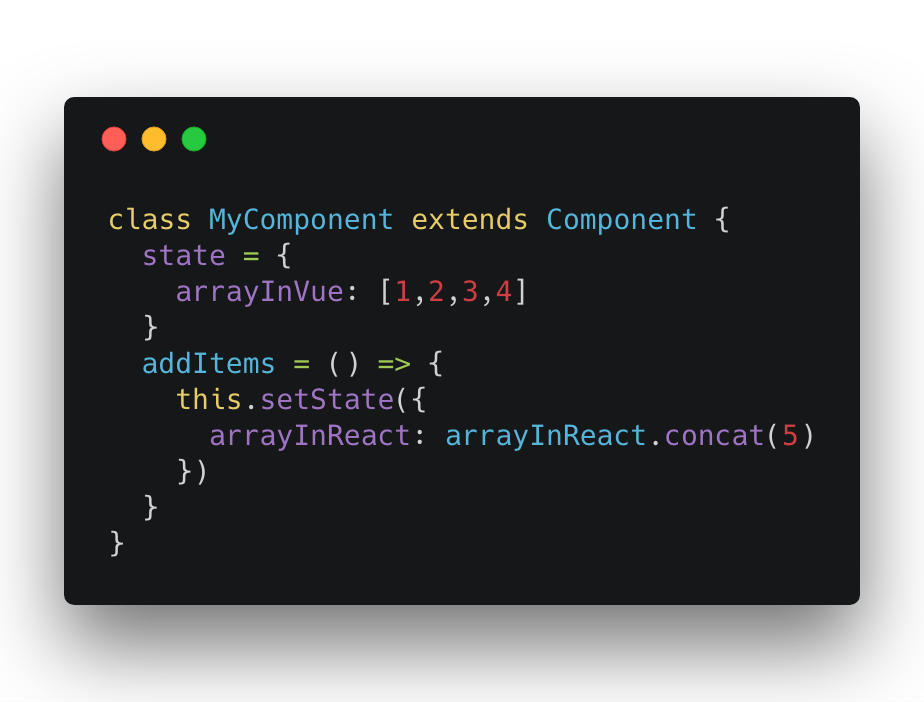
6. 둘 다 서버사이드 렌더링용 웹서버 프레임워크(SSR Framework)가 있다. webpack 기반 자바스크립트이다보니 둘 다 nodeJS 기반이다. 리액트는 nextjs, 뷰는 nuxtjs.

- 차이점:

1. 데이터를 핸들링하는 방식. 아래의 예제는 Array 타입의 데이터를 핸들링 하는 코드이다. 만약 Array 타입의 데이터에 값을 추가하는 경우, Vue를 이용할 때는 아래와 같이 array의 내장 메소드인 push 메소드를 이용하면 쉽게 추가할 수 있다.



하지만 React의 경우에는 기존의 배열의 깊은 복사를 이용하여 setState에서 값을 변경해주어야 한다.



2. 서술 방식. 일단 Vue 컴포넌트의 가장 큰 특징으로는 SFC(Single File Component) 라는 점이다. SFC는 하나의 파일 안에 DOM, Script, Style을 포함하고 있다는 특징이 있다.



하지만 반대로 React의 경우는 다음과 같다.



출처:

<https://velog.io/@vraimentres/react-vs-vue-1>

<https://blog.martinwork.co.kr/review/2018/11/01/between-vuejs-and-reactjs.html>

3. STATE와 PROPS를 설명하시오.

- State: 제품을 구현하는 데에 필요한 내부적 조작장치라고 말할 수 있다. props의 값에 따라 내부에 필요한 데이터들이라 말하는데, 컴포넌트 내부에서 선언하며 내부에서 값을 변경할 수 있다. 즉, 컴포넌트에서 관리하는 상태 값으로 유동적인 데이터를 다룰 때, state를 사용한다. state는 변경이 가능하고 변경할 때는 setState메서드를 사용해 상태를 변경한다.

- Props: 사용자가 제품을 조작하는 장치라고 말할 수 있다. 사용자가 컴포넌트를 사용하는 입장에서 중요한 것인데, 부모 컴포넌트가 자식 컴포넌트에게 주는 값이다. 자식 컴포넌트에서는 props를 받아오기만 하고, 받아온 props를 직접 수정할 수는 없다. 즉, 컴포넌트에서 사용할 데이터 중 변경되지 않는 데이터를 다룰 때 사용한다.

출처:

https://velog.io/@dev\_glennn/생활코딩-React-State-Props

https://velog.io/@kyusung/리액트-교과서-React-컴포넌트와-상태-객체