

[IT Essential] Hello React

날짜 : 2020.08.12

진행 : 황수재 컨설턴트

Hello React

- IT Essential에서 React를 다루는 이유
 - 라이브 방송에서 소개해줬으면 하는 기술
 - 프로젝트 진행 시 도움이 될 React의 기본 개념 설명
 - 현업에서 많이 사용되고 있는 기술 중 하나
- Framework인지 Library인지 (실제 홈페이지 내용)
 - React - JavaScript Library
 - Vue - JavaScript Framework

React는 왜 만들었을까?

- React 공식 홈페이지 참고(<https://ko.reactjs.org/blog/2013/06/05/why-react.html>)
 - React MVC 프레임워크가 아니다
 - React는 라이브러리이다
 - 재사용가능한 UI Components
 - 데이터를 표시
 - 시간에 따른 데이터 변경

React를 학습하는 방법

- React 공식 홈페이지 참조
- 실습을 통해서 학습
 - React 자습서 활용(Tip도 존재)
 - <https://ko.reactjs.org/tutorial/tutorial.html>
- 개념들을 단계적으로 학습
 - React 문서 활용
 - <https://ko.reactjs.org/docs/getting-started.html>

UI Components

- 코드를 재사용하기 위해서 구조를 설계
- 구조 설계만 있고 데이터는 없음

Data(Props)

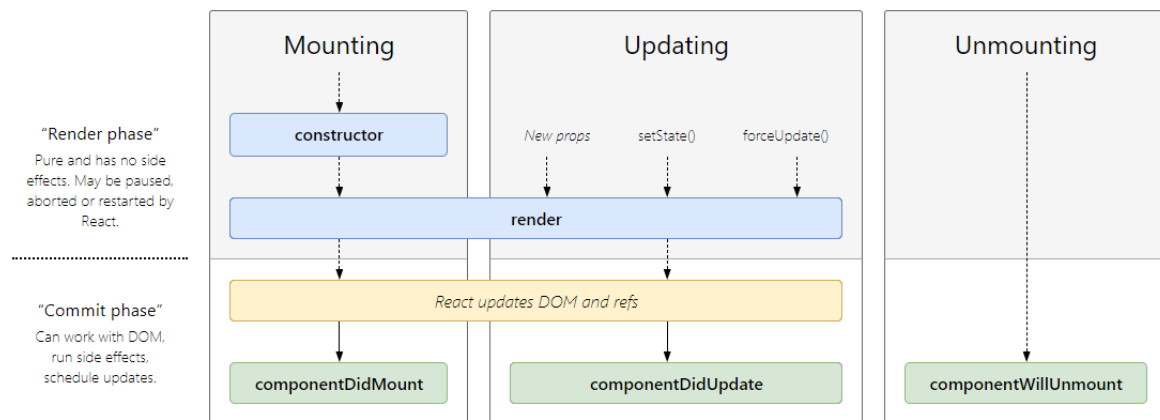
- UI Components가 데이터를 받기 위한 수단
- this.props로 사용
- Components를 이용하는 곳에서 Key값으로 넣어서 전달

Data(State)

- 시간의 흐름에 따라 바뀌는 데이터 또는 상태를 관리

Lifecycle

- 단순한 구조



- Mounting : 만들어 질때 , `componentDidMount`(UI Component가 데이터를 가지고 표현하고 그 뒤에 무엇을 할지를 관리)
- updating : 시간의 흐름에 따라 변경이 될 때, `componentDidUpdate`(props or state가 변경되었을 때 다시 `render`를 할 때)
- Unmounting : 없어질 때, `componentWillUnmount`(UI Component가 없어질 때)
- 디테일한 구조는 공식 홈페이지 참고

추가 학습 자료

- Awesome React : <https://github.com/enaqx/awesome-react>
- Awesome React Components : <https://github.com/brillout/awesome-react-components>
- Awesome React Talks : <https://github.com/tiaanduplessis/awesome-react-talks>

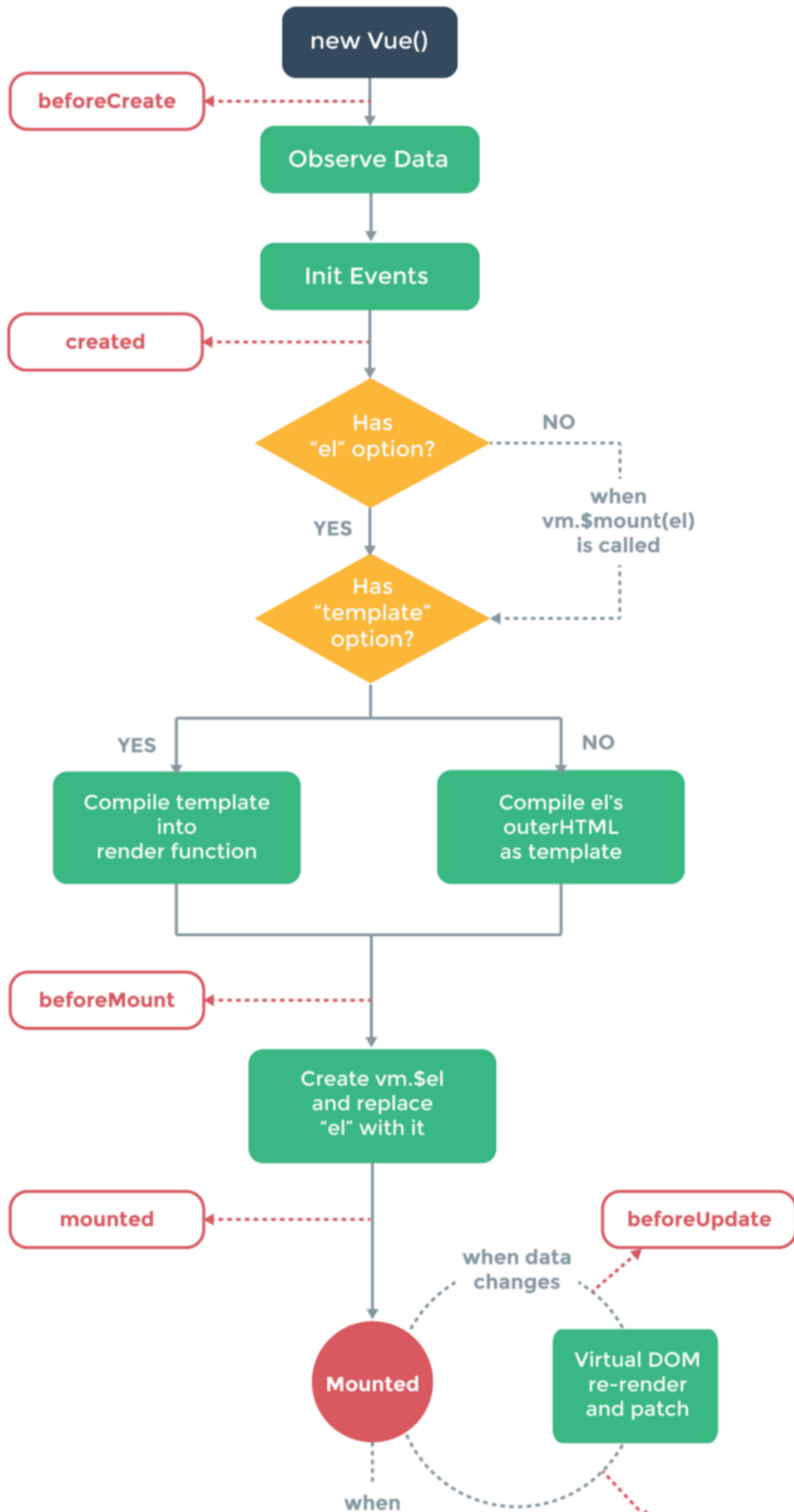
퀴즈

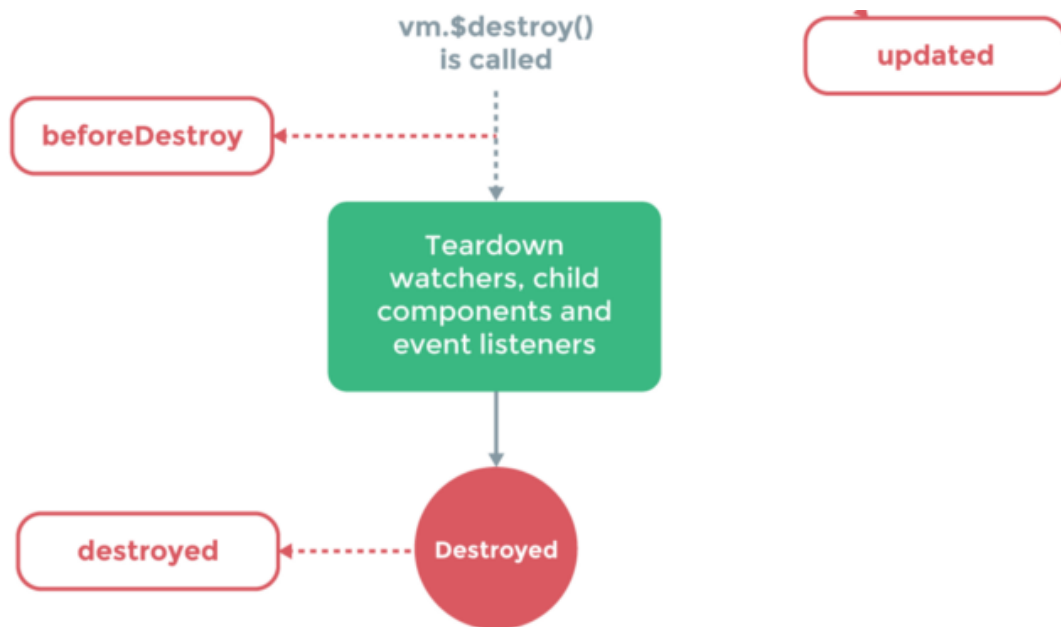
- React를 학습하기 가장 좋은 사이트는?
 - React 공식 사이트

추가

Vue.js

- Vue는 사용자 인터페이스를 만들기 위한 프로그레시브 프레임워크
- v-model을 이용한 양방향 바인딩
- 공식 홈페이지 - <https://kr.vuejs.org/>
- LifeCycle





React 공통점

- 가상 DOM을 활용
- 반응적이고 조합 가능한 컴포넌트를 제공
- 코어 라이브러리에만 집중하고 있고 라우팅 및 전역 상태를 관리하는 컴패니언 라이브러리가 존재

React 차이점

- 렌더링 성능
 - Vue와 React 모두 가상 DOM을 사용하여 구현하지만 Vue의 가상 DOM 구현은 훨씬 가벼워서 React보다 더 적은 오버 헤드가 발생
 - 2014 년 MacBook Air의 Chrome 52에서 실행한 결과는 Vue가 더 빠름
- 갱신 성능
 - React는 컴포넌트의 상태가 변경되면 해당 컴포넌트에서 루트로 시작하는 전체 컴포넌트 하위 트리를 다시 렌더링
 - Vue에서 컴포넌트의 종속성은 렌더링 중 자동으로 추적되어서 실제로 다시 렌더링해야하는 컴포넌트를 정확히 알 수 있음