**1. REALDOM VS VIRTUALDOM 각각 설명하시오**

|  |  |
| --- | --- |
| **Real Dom** | **Virtual Dom** |
| - DOM에 변경이 일어남(data의 update) -> DOM Tree가 다시 만들어짐 -> Render Tree도 다시 만들어짐 -> 레이아웃-페인트-합성)  - 10개의 list 중 하나의 list만 변경이 있어도  10개 list 전체를 Reload 해야 됨.  - 비용이 큰 작업 | - Virtual DOM에서 Real DOM의 스냅샷을 찍어 놓음 -> DOM에 변경이 일어남 -> 실제로 브라우저가 관리하는 Real DOM과 리액트가 메모리상에서 관리하는 Virtual DOM을 shallow comparison(얕은 비교)하여 변경된 부분이 있나 관찰 -> 변경된 부분이 있으면 Real DOM에서 바꾸어 줌  - 10개의 list 중 하나의 list만 변경이 있으면  그 한가지 아이템만 dom에서 바꿔준다.  - Reload가 딱 한번만 일어남. (대신 규모가 크다)  - 부하가 걸리지 않고, (상대적으로) 비용이 적게 든다.  - Real DOM의 가벼운 사본이며, Virtual DOM은 메모리에 존재한다. ( Real DOM과 같은 properties를 갖는다.)  - 실제 DOM에 접근하여 조작하는 대신, 이를 추상화시킨 자바스크립트 객체를 구성하여 사용  - Virtual DOM은 DOM의 상태를 **메모리** 위에 계속 올려두고 변경 전과 변경 후의 상태를 비교한 뒤, DOM에 변경이 있을 때만 해당 변경을 반영한다. |

**2. 버츄얼돔은 왜 좋은가. 어떻게 동작하는가. JQUERY의 돔 직접참조에 비해서 무엇이 개선되었는가.**

### **Virtual DOM 그리고 JQUERY의 DOM 직접참조**

jQuery는, 내부적으로 가능하다면 순수 JavaScript의 브라우저 내장 method를 사용하게 된다. (속도 향상을 위해) 하지만 브라우저 내장 method가 지원되지 않는 경우 (jQuery만의 확장된 선택자. ex) :last, :first, :even, :eq() ), 모든 DOM node를 순회해야 하기 때문에 속도가 느려진다.

Virtual DOM은 실제로 브라우저가 관리하는 Real DOM과 리액트가 메모리상에서 관리하는 Virtual DOM을 shallow comparison(얕은 비교)하여 변경된 부분이 있나 관찰하고, DOM에 변경이 있을 때만 해당 변경을 반영하기 때문에 속도가 개선된다.

**3. 버츄얼돔이 동작하는 DIFF알고리즘에 대해 설명하시오.**

**Diffing Algorithm**

Diffing

- 사전 업데이트 된 Virtual DOM과 업데이트 된(ex- setState) Virtual DOM을 비교하는 과정을 diffing이라고 한다.

- React는 그래서 매번 두 개의 Virtual DOM을 유지/관리한다.

- Real DOM에서는 변경된 내용만 업데이트 된다.