

4. 엘리먼트 렌더링

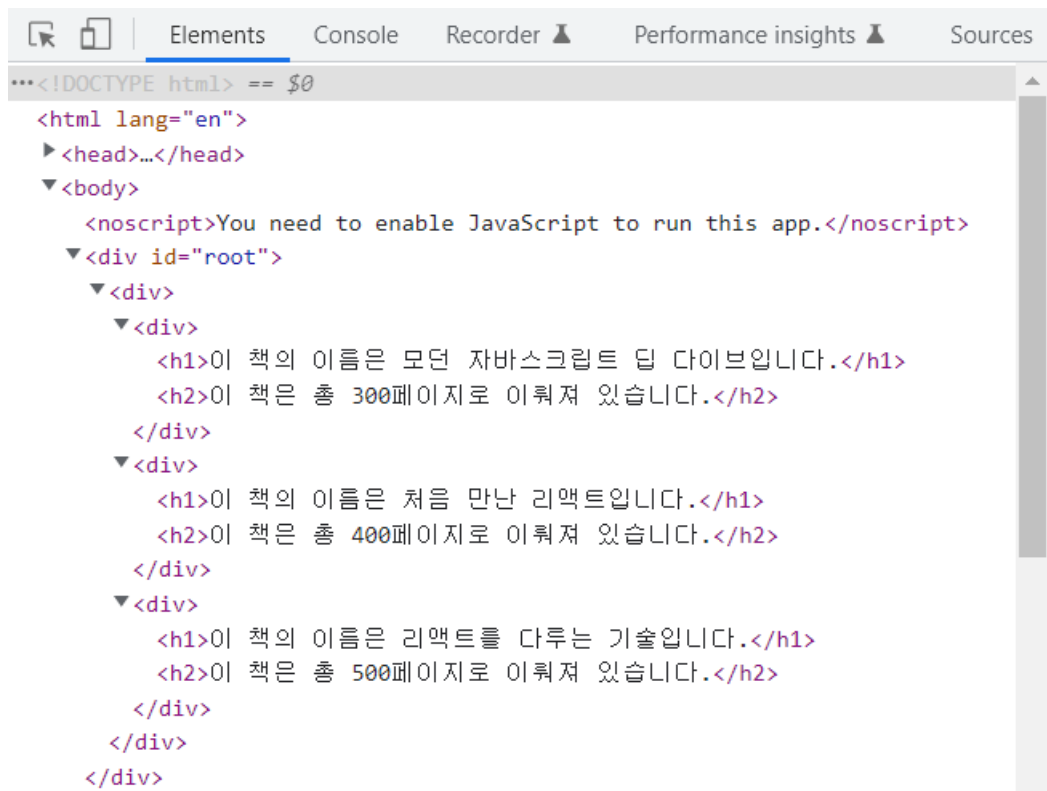
4.1 엘리먼트(Elements)란?

- Elements are the smallest building blocks of React apps.
- 엘리먼트는 리액트 앱을 구성하는 가장 작은 블록들
- 컴포넌트의 "구성 요소"
- 화면에 표시할 내용을 기술하는 자바스크립트 객체
 - 컴포넌트 유형과 속성, 내부의 모든 자식 정보를 포함하는 일반 객체
- 한 번 생성되면 변경할 수 없는 객체(불변성, immutable)

```
const element = <h1>Hello, world</h1>;
```

4.1 엘리먼트(Elements)란?

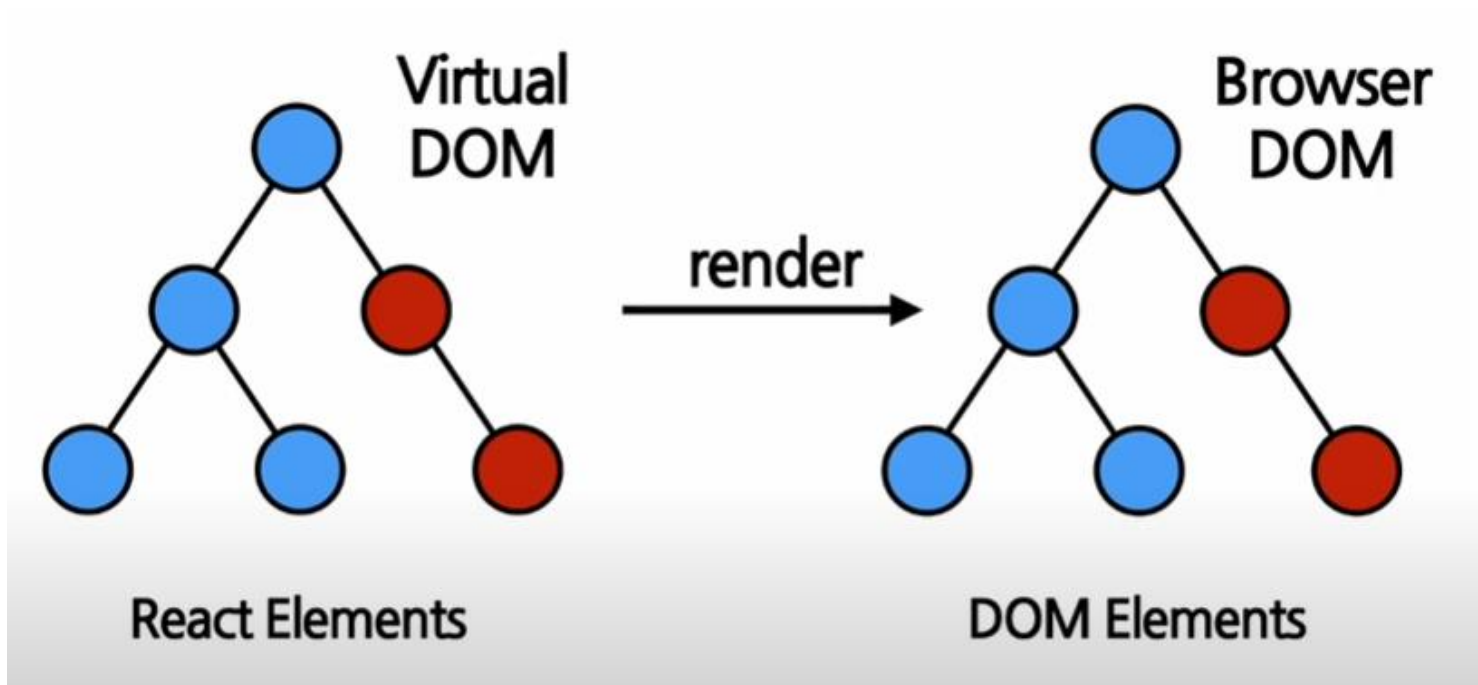
- 우리가 알고 있는 DOM Elements



```
...<!DOCTYPE html> == $0
<html lang="en">
  <head>...</head>
  <body>
    <noscript>You need to enable JavaScript to run this app.</noscript>
    <div id="root">
      <div>
        <div>
          <h1>이 책의 이름은 모던 자바스크립트 딥 다이브입니다.</h1>
          <h2>이 책은 총 300페이지로 이뤄져 있습니다.</h2>
        </div>
        <div>
          <h1>이 책의 이름은 처음 만난 리액트입니다.</h1>
          <h2>이 책은 총 400페이지로 이뤄져 있습니다.</h2>
        </div>
        <div>
          <h1>이 책의 이름은 리액트를 다루는 기술입니다.</h1>
          <h2>이 책은 총 500페이지로 이뤄져 있습니다.</h2>
        </div>
      </div>
    </div>
  </body>
</html>
```

4.1 엘리먼트(Elements)란?

- 리액트 엘리먼트는 DOM 엘리먼트의 가상 표현



4.2 엘리먼트의 특징

- 불변성(immutable) = 변하지 않는 성질
 - 한번 생성된 엘리먼트는 변하지 않음
 - 즉, **엘리먼트 생성 후**에는 자식이나 속성을 바꿀 수 없음
 - 예시: 붕어빵 틀(컴포넌트) – 붕어빵(엘리먼트)
- 엘리먼트가 변할 수 없다면 화면 갱신이 안되는 것 아닌가..?
 - 기존 엘리먼트를 변경하는 것이 아니라 새로운 엘리먼트를 만드는 것
 - 새로운 엘리먼트를 기존 엘리먼트와 바꿔치기

4.3 엘리먼트 렌더링하기

- 이 안에 들어가는 모든 엘리먼트를 React DOM에서 관리하기 때문에 이것을 루트(root) DOM 노드라고 부름
- React로 구현된 애플리케이션은 일반적으로 하나의 루트 DOM 노드가 있음

```
<div id="root"></div>
```

```
const root = ReactDOM.createRoot(  
  document.getElementById('root')  
);  
const element = <h1>Hello, world</h1>;  
root.render(element);
```

- root.render(): 리액트 엘리먼트를 DOM 엘리먼트로 렌더링하는 역할
- 리액트 엘리먼트가 렌더링되는 과정 = 가상 DOM에서 실제 DOM으로 이동하는 과정

4.4 렌더링 된 엘리먼트 업데이트하기

- React 엘리먼트는 불변객체
- 엘리먼트를 생성한 이후에는 해당 엘리먼트의 자식이나 속성을 변경할 수 없음
- UI를 업데이트하는 유일한 방법은 새로운 엘리먼트를 생성하고 이를 `root.render()`로 전달

4.5 시계 만들기(실습)

- my-app/src/chapter4 실습
 - Clock.jsx
 - index.js
 - 전체 UI를 다시 렌더링 하도록 만들었지만 React DOM은 내용이 변경된 텍스트 노드만 업데이트