

## 쇼폴의 처음 만나 리액트



## 제 4장 엘리먼트 렌더링



## Q & A

- 언제라도 질문하세요  
모르면 외우면 되지만 코딩 안되면 꼭 질문
  1. 대면수업에서 질문
  2. 한성e-class 질의응답게시판
- 강사의 1번 선생님은 여러분의 질문





# 목차

1. 엘리먼트에 대해 알아보기
2. 엘리먼트 렌더링하기
3. 렌더링된 엘리먼트 업데이트 하기
4. 실습: 시계 만들기
5. 마치며



# 1. 엘리먼트에 대해 알아보기

- Elements are the smallest building blocks of React apps.
- F12의 elements창에서 element 볼 수 있음

이 책의 이름은 처음 만난 파이썬입니다.

이 책은 총 300페이지로 이뤄져 있습니다.

이 책의 이름은 처음 만난 AWS입니다.

이 책은 총 400페이지로 이뤄져 있습니다.

이 책의 이름은 처음 만난 리액트입니다.

이 책은 총 500페이지로 이뤄져 있습니다.

## DOM Elements

```
<!DOCTYPE html>
<html lang="en">
  <head>...</head>
  <body>
    <noscript>You need to enable JavaScript to run this app.</noscript>
    <div id="root">
      <div>
        <h1>이 책의 이름은 처음 만난 파이썬입니다.</h1>
        <h2>이 책은 총 300페이지로 이뤄져 있습니다.</h2>
      </div>
      <div>
        <h1>이 책의 이름은 처음 만난 AWS입니다.</h1>
        <h2>이 책은 총 400페이지로 이뤄져 있습니다.</h2>
      </div>
      <div>
        <h1>이 책의 이름은 처음 만난 리액트입니다.</h1>
        <h2>이 책은 총 500페이지로 이뤄져 있습니다.</h2>
      </div>
    </div>
  </body>
</html>
```

Styles

Filter

element.style { }

div { display: block; }

Inherited from body

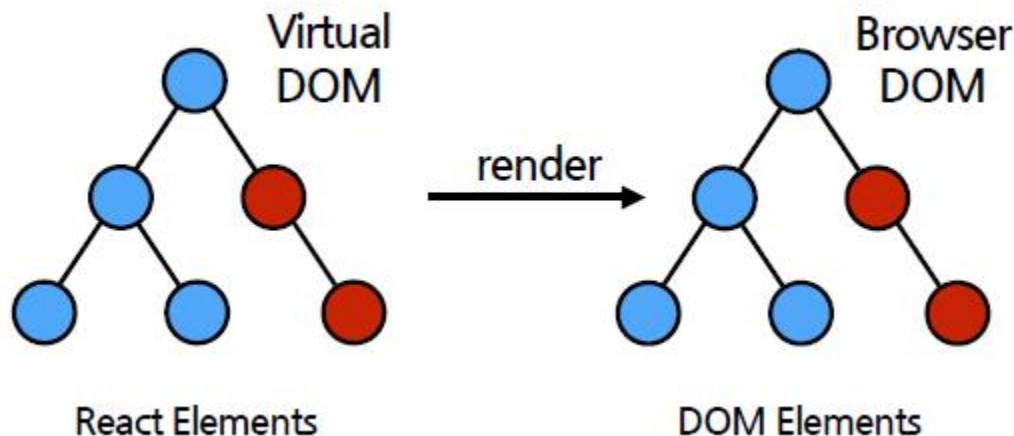
body { margin: 0; font-family: -apple-system, BlinkMacSystemFont, 'Segoe UI', 'Roboto', 'Oxygen', 'Ubuntu', 'Cantarell', 'Fira Sans', 'Droid Sans', 'Helvetica Neue', sans-serif; -webkit-font-smoothing: antialiased; }



1. 엘리먼트에 대해 알아보기 (code)

2. 엘리먼트 렌더링하기

- 리액트 엘리먼트와 DOM 엘리먼트 관계



- reactJS 코드에서 리액트 엘리먼트 생성 및 render  

```
const root = ReactDOM.createRoot(document.getElementById('root'));  
root.render(React.createElement(Clock2));
```
- vanillaJS 코드에서 DOM 엘리먼트 생성 및 DOM 트리에 삽입  

```
let newDIV1 = document.createElement("div");  
newDIV1.innerHTML = "clock2 현재 시간:" + new Date().toLocaleTimeString();  
parent.appendChild(newDIV1);
```



# 1. 엘리먼트에 대해 알아보기 (doc)

## 2. 엘리먼트 렌더링하기

- reactJS 코드에서 리액트 엘리먼트 생성 및 render  
`const root = ReactDOM.createRoot(document.getElementById('root'));`  
`root.render(React.createElement(Clock2));`  
`React.createElement()` : <https://react.dev/reference/react/createElement>  
`root.render()` : <https://react.dev/reference/react-dom/client/createRoot#root-render>
- vanillaJS 코드에서 DOM 엘리먼트 생성 및 DOM 트리에 삽입  
`let newDIV1 = document.createElement("div");`  
`newDIV1.innerHTML = "clock2 현재 시간:" + new Date().toLocaleTimeString();`  
`parent.appendChild(newDIV1);`  
// 웹기초 8.4 HTML 문서의 동적구성  
// DOM 객체 동적 생성  
// DOM 트리에 삽입  
// DOM 객체의 삭제

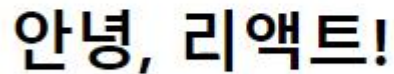
학생> `createElement()`는 생성인데 `render()`는 무슨 뜻인가요?

교수> `render`가 make 뜻인데 여기서는 DOM 트리 만든다는 뜻

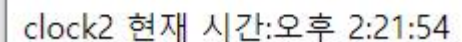
Element 생성 후 DOM 트리 만들면 자동으로 크롬화면에 표시됩니다.



- 코드위치: c:\ws\_react\CDN\chapter\_04\_reactJS
- Clock.jsx를 Babel을 사용하여 Clock.js로 변환
- CDN\chapter\_04\_reactJS폴더를 실행하면 1초마다 시간이 변경되는 것을 볼 수 있음
- Index.js 9번줄의 setInterval()은 w3schools에서 보면 [https://www.w3schools.com/jsref/met\\_win\\_setinterval.asp](https://www.w3schools.com/jsref/met_win_setinterval.asp)
- 9번줄의 () => { ... } 는 화살표 함수로서 function () {...} 와 동일
- Clock은 root div 객체 아래에 child 1개 grandchild 2개  
root-div > div > h1  
                > h2
- Clock2는 root div 객체 아래에 child 1개  
root-div > div  
실행하면 1초마다 시간이 변경



현재 시간: 오후 2:02:34





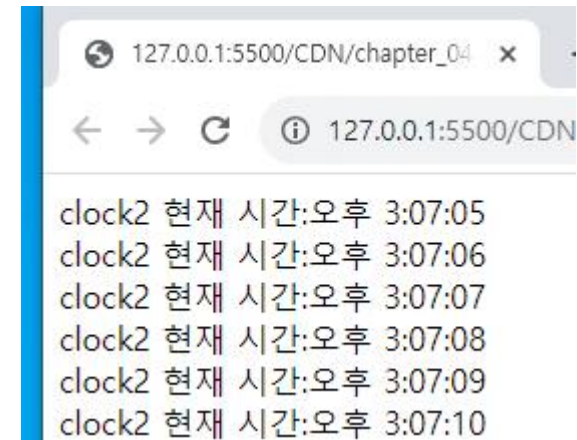
### 3. 렌더링된 엘리먼트 업데이트 하기 (3. vanillaJS 방식)

#### 4. 시계만들기

- 코드위치: c:\ws\_react\CDN\chapter\_04\_vanillaJS
- reactJS의 Clock2.js를 vanillaJS로 구현
- CDN\chapter\_04\_vanillaJS폴더를 실행하면  
1초마다 시간 표시줄이 증가 되는 것을 볼 수 있음
- Index.js에서 Clock2를 Clock3으로 변경하여 실행하면 1  
초마다 시간이 변경
- vanillaJS에서 엘리먼트 업데이트하려면 appendChild  
대신 replaceChild 사용  
[https://www.w3schools.com/jsref/met\\_node\\_replacechild.asp](https://www.w3schools.com/jsref/met_node_replacechild.asp)

학생> 웹기초에서 appendChild는 배웠는데 replaceChild는 배우지 않았는데요

교수> 이번에 배우면 됩니다







### 3. 렌더링된 엘리먼트 업데이트 하기 (4. noJS 방식)

#### 4. 시계만들기

- 코드위치: c:\ws\_react\CDN\chapter\_04\_noJS
- vanillaJS와 동일하게 tag 업데이트를 할 수 없음  
즉 noJS로 구현할 수 없음  
HTML의 특징



# 이제 나는 할 수 있다

1. 나는 엘리먼트의 뜻을 알고 F12 element창에서 element를 확인했다.
  2. 나는 reactJS와 vanillaJS에서 엘리먼트 렌더링하는 방법을 알았다
  3. 나는 reactJS와 vanillaJS로 렌더링된 엘리먼트 업데이트하는 방법을 알았다
- 유튜브 강의 동영상 : <https://www.youtube.com/c/소플TV>
  - Q&A 사이트 : <https://www.frontoverflow.com>