

# FE는 왜 어렵다고 느껴지는가?

몹시 주관적인 시선으로 접근해보는

# 목차

- 눈을 가리는 정말 다양한 Framework / Library
- 브라우저 및 버전별 파편화된 기능
- 번들러의 “보이지 않는 손”
- 발상의 전환

눈을 가리는 정말 다양한  
Framework / Library

# 눈을 가리는 정말 다양한 Framework / Library

- UI Framework / Library
- Styling
- Data fetching
- State management
- Animation / Interaction / Motion
- Bundler
- Accessibility
- 기타 Linter / Formatter / Testing Tool 등은 제외

# UI Framework / Library

눈을 가리는 정말 다양한 것들

- React
  - Next.js
  - Remix.js
  - astro.js
- Preact
- qwik
- Vue
- Angular
- Svelte
  
- Whatever ton's of framework

# Styling

눈을 가리는 정말 다양한 것들

- Utility-First
  - tailwindcss
- css in js
  - emotion
  - styled-components
  - stitches
  - linaria
- CSS 방법론
  - OOCSS
  - SMACSS
  - Atomic CSS
- sass, scss
- UI Component library
  - MUI
  - Antd
  - ...

# Data fetching

눈을 가리는 정말 다양한 것들

- Fetching only
  - axios
  - ky
  - fetch
  - superagent
- w/ cache
  - RTK Query
  - Tanstack Query (React Query)
  - Swr
  - Realy
  - Apollo

# State management

눈을 가리는 정말 다양한 것들

- Redux
- xstate
- Zustand
- Jot
- mobx
- Mobx-state-tree
- Recoil

# Animation / Interaction / Motion

눈을 가리는 정말 다양한 것들

- Framer motion
- Lottie
- Rive
- fullpage.js
- swiper.js
- react-transition-group
- react-spring

# ETC

눈을 가리는 정말 다양한 것들

- 접근성
  - React Aria
- 기타 번들러들
- 기타 테스트
- 기타 e2e 테스트
- 기타 경우에 맞는 설정

당연히 필요한 경우가 아니면 몰라도 됩니다

하지만 본질을 가리기에는 충분하죠

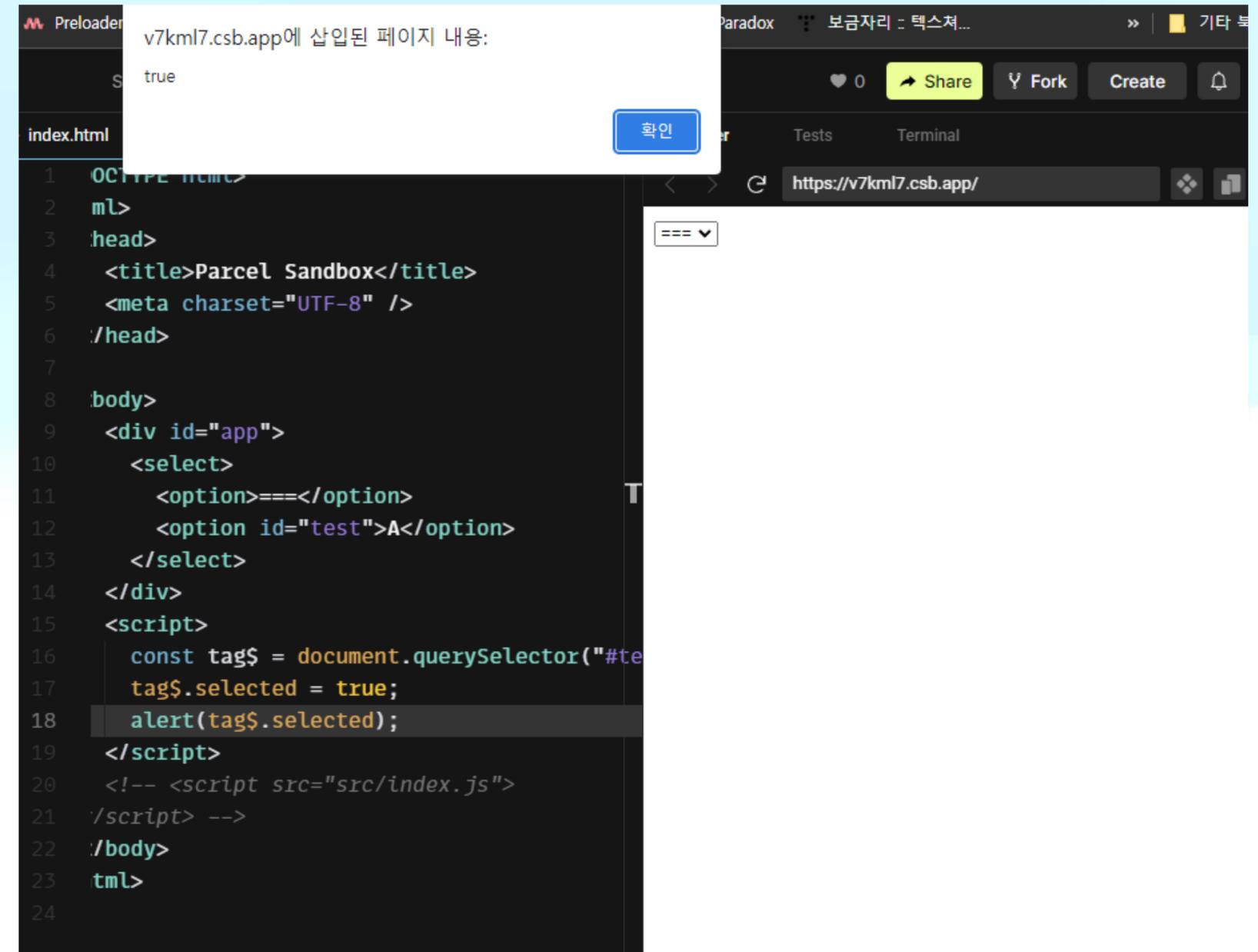
브라우저 및 버전별 파편화된 기능

# 크로스 브라우징

- <https://caniuse.com/>
- <https://github.com/postcss/autoprefixer>

# 기묘한 HTML/CSS

- Play It



The image shows a code editor on the left and a browser window on the right. The code editor displays the following HTML and JavaScript code:

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Parcel Sandbox</title>
5   <meta charset="UTF-8" />
6 </head>
7
8 <body>
9   <div id="app">
10     <select>
11       <option>===</option>
12       <option id="test">A</option>
13     </select>
14   </div>
15   <script>
16     const tag$ = document.querySelector("#te
17     tag$.selected = true;
18     alert(tag$.selected);
19   </script>
20   <!-- <script src="src/index.js">
21 </script> -->
22 </body>
23 </html>
24
```

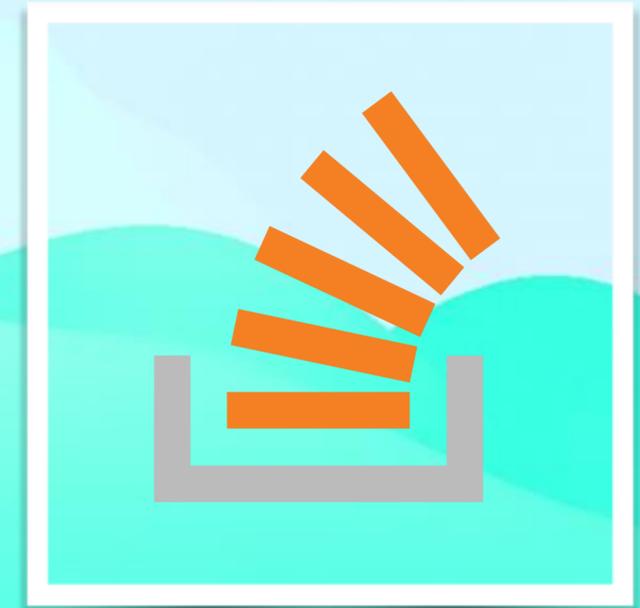
The browser window shows the URL `https://v7kml7.csb.app/`. An alert box is displayed in the foreground with the text: "v7kml7.csb.app에 삽입된 페이지 내용: true". A blue button labeled "확인" (Confirm) is visible on the alert box.

번들러의 “보이지 않는 손”

# 번들러의 “보이지 않는 손”

- CRA기준
  - Minify
  - Tree shaking
  - 청크 기반으로 번들 나누기
  - 각종 로더를 기반으로 번들로서 추가 및 최적화
  - (TS 및 그와 유사한 슈퍼셋 언어를 쓰고 있다면) Transpile
  - postcss 및 관련된 플러그인을 통한 css보조
  - HMR (hot module loading)
  - Polyfill
  - 환경변수 설정
  - 그 외 경우에 따라 다양한 설정
- 참고
  - <https://github.com/facebook/create-react-app/blob/main/packages/react-scripts/package.json>

# 발상의 전환 - 필요로 하는 백그라운드 지식 및 기법



이 두 서비스의 공통점은?

딱히 모던하지 않지만  
이용함에 있어서 큰 지장이 없다는 점!

중요한 건 필요에 따라 맞춤형의 기술 및 기법들

# Conclusion

- 공식문서를 가까이 하시고 이상한 카더라 는 멀리하십시오
- 적합한 기술은 항상 변함
  - 지금 해야 하는 것, 혹은 아닌 것들을 구분할 필요가 있음
  - 왜 필요한가를 다소 잘 알 필요가 있음

결국 미지의 것이라는 인식이  
해야할 일을 가리는 것 같습니다.

# End.

Happy...? FE?