



Frontend Convention

ES6

- spread 연산자 사용
- 구조 분해 할당 사용
- let, const만 사용
- 되도록 arrow function 사용
- 컴포넌트 작성은 function함수로

ex

```
function App(){  
  return // code  
}
```

styled-components

태그명은 S + PascalCase로 작성 (스타일을 입힌 태그라는 의미)

```

```
const SSection = styled.div`
 // code
`;
```

```
const SHeader = styled.header`
 // code
`;
```

```
const SSection = styled.section`
 // code
`;
```

```
const SInput = styled.input`
 // code
`;
```
```

파일명

component는 PascalCase

```

```
Header.js
Footer.js
Home.js
```
```

non-component는 camelCase

```

    ```
 fetchApi.js
    ```

    # 변수명

    camelCase와 영어, 대소문자, 숫자를 사용

    ```
 const isLog = false;
 const list2 = ["a", "b"];
    ```

    변수로 할당되는 값이 boolean인 경우 is를 접두사로

    ```
 const isLoading = false;
 const isOpen = true;
    ```

    상수는 대문자로만 작성

    ```
 const URL = "";
 const API_KEY = "";
    ```

    # 함수명

    handle + 기능 + 이벤트로 작성

    ```
 <button onClick={handleResetClick}>리셋 버튼</button>
    ```

    # 기타

    - 컴포넌트는 무조건 함수형으로 작성한다.
    - 하나의 파일에 하나의 React Component만 만든다.
    - 불필요한 주석은 사용하지 않는다.
    - propTypes를 사용하여 전달되는 타입을 구체화한다.
    - 어려운 비즈니스 로직을 작성할 때는 주석을 적는다.
    - 인라인 함수는 사용하지 않는다.
    - commit시 console.log는 삭제한다.
  
```

Project Folder 구조

```

Page > 유기체 > 분자 > 원자
- Page
  - 한 페이지
- 유기체
  - 상단바
  - 지도
  
```

- 데이터1행
 - api받아서
- 데이터2행
 - api 받아서
- 데이터3행
- 텍스트마이닝
- 뉴스기사
- 분자
 - 상단바
 - 로고
 - 국가&품목Selector
 - 교역지역
 - 모달까지 다 넣기
 - 조회기준
 - 모달까지 다 넣기
 - PDF 다운
 - 일단은 이미지만
 - Excel 다운
 - 일단은 이미지만
 - 지도
 - 수입 수출 수출입 신호등
 - 세계지도
 - 국가 수출 수입 모달
 - 데이터 1행
 - 수출입 비중 도넛 차트
 - `유기체/데이터1행` 에서 데이터 가져오기
 - 수출 수입 무역수지 정보 테이블
 - `유기체/데이터1행` 에서 데이터 가져오기
 - 데이터 2행
 - 꺾은선 그래프
 - 수출 TOP5
 - 수입 TOP5
 - 데이터 3행
 - 텍스트마이닝
 - 뉴스기사
- CSS
 - tailwind
 - 공식문서 참조해서 만들기

Coding Convetion

- <https://euid.notion.site/euid/Vite-3bc2879d2f34408f9f2ddf24c50a0ff9>
 - Eslint, Prettier 참고해서 만들기