

```

(async () => {
  let json = null;

  try {
    // ajax 처리 구현
    const response = await axios.??(#1)??('??(#2)??', ??(#3)??);
    json = response.data;
  } catch (e) {
    // 에러 발생시 처리
    alert("AJAX 요청에 실패했습니다.");
  }

  if (json != null) {
    ??(#4)??
  }
})();

```

#1. 백엔드에 데이터를 전송하는 방식

- get : 조회
- post : 저장
- put : 수정
- delete : 삭제

#2. 백엔드 URL

http:// 혹은 https://로 시작되어야 함.

프론트엔드가 접근중인 도메인과 동일한 도메인을 갖는 백엔드에만 접근 가능함.

단, 백엔드에서 외부 접근 허용을 위한 코드를 구현했을 때는 다른 도메인 접근 가능.

#3. 백엔드에 전달할 요청변수

GET

```

{
  params: {key: value, key: value ...}
}

```

POST, PUT, DELETE

```

{key: value, key: value ...}

```

#4. 백엔드가 처리 결과로 출력해주는 JSON을 어떻게 활용할지에 대한 구현

대체로 JS로 HTML DOM을 생성하여 JSON 내용을 화면에 표시하는 것으로 마무리.

경우에 따라 기존의 DOM을 갱신하거나 삭제할 수 도 있다.

이러한 처리를 언제 가동할지에 대해 고민하는 것이 이벤트 처리