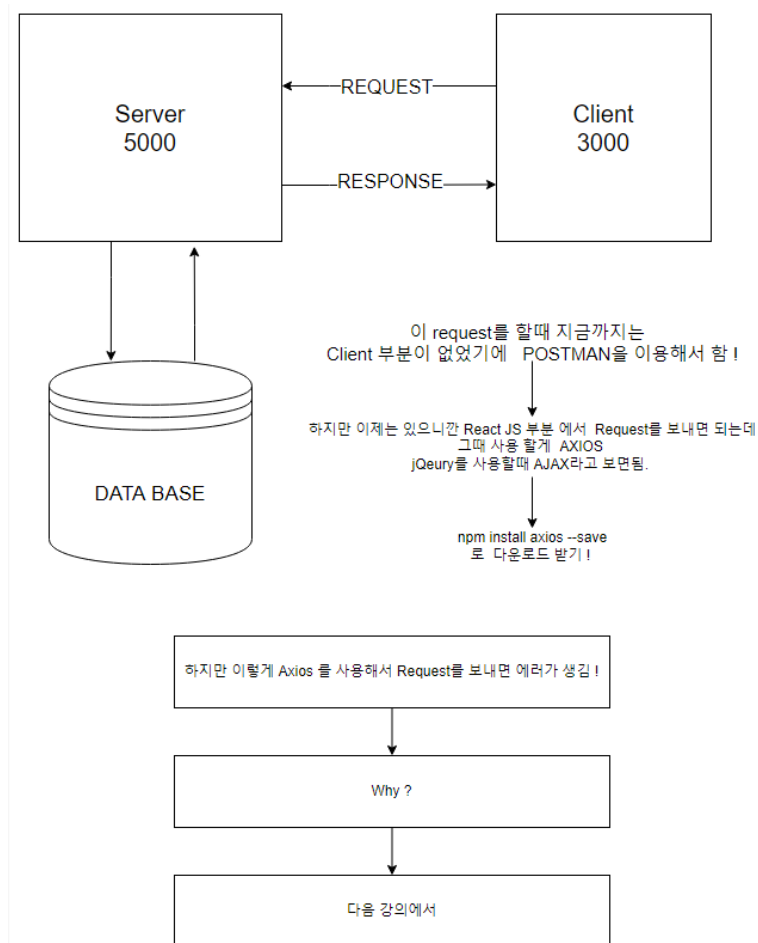


#21 데이터 Flow & Axios



- Axios

- node.js와 브라우저를 위한 Promise API를 활용하는 HTTP **비동기** 통신 라이브러리
- 백엔드와 프론트엔드 통신을 쉽게 하기 위해 AJAX와 더불어 사용
- 자바스크립트에는 fetch api가 있지만 프레임워크에서 ajax를 구현할 때 axios를 쓰는편?
- AJAX vs Axios vs Fetch



Asynchronous JavaScript And XML의 약자이며, 자바스크립트를 이용해 클라이언트와 서버 간에 데이터를 주고받는 비동기 HTTP 통신입니다. XMLHttpRequest(XHR) 객체를 이용해서 전체 페이지가 아닌 필요한 데이터만 불러올 수 있습니다.

장점

- JQuery를 통해 쉽게 구현 가능
- Error, Success, Complete의 상태를 통해 실행 흐름 조절 가능

단점

- JQuery를 사용해야 간편하고 호환성이 보장됨
- Promise 기반이 아님

XMLHttpRequest(XHR)를 사용한 코드

```
var xhr = new XMLHttpRequest();
xhr.onreadystatechange = function() { // 요청에 대한 콜백
  if (xhr.readyState === xhr.DONE) {
    // 요청이 완료되면
    if (xhr.status === 200 || xhr.status === 201) {
      console.log(xhr.responseText);
    } else {
      console.error(xhr.responseText);
    }
  }
};
xhr.open('GET', 'https://localhost:3000'); // 메소드와 주소 설정
xhr.send(); // 요청 전송
// xhr.abort(); // 전송된 요청 취소
```

JQuery를 사용한 코드

```
var serverAddress = 'https://localhost:3000';

// jQuery의 .get 메소드 사용
$.ajax({
  url: serverAddress,
  type: 'GET',
  success: function(data) {
    console.log(data);
  },
  error: function(error) {
    console.error(error);
  }
});
```

확실히 JQuery를 사용할 때 코드가 훨씬 간단해지고 직관적인 것 같습니다.

★ axios

axios는 Node.js와 브라우저를 위한 Promise API를 활용하는 HTTP 통신 라이브러리입니다. 비동기로 HTTP 통신을 할 수 있으며 return을 promise 객체로 해주기 때문에 response 데이터를 다루기 쉽습니다.

💎 장점

- response timeout (fetch에는 없는 기능) 처리 방법이 존재
- Promise 기반으로 만들어졌기 때문에 데이터 다루기 편리
- 브라우저 호환성이 뛰어남

💣 단점

- 사용을 위해 모듈 설치 필요 (npm install axios)

📋 코드

```
axios({
  method: 'post',
  url: 'https://localhost:3000/user',
  data: {
    userName: 'Cocoon',
    userId: 'co1234'
  }
}).then((response) => console.log(response));
```

★ fetch

ES6부터 들어온 JavaScript 내장 라이브러리입니다. Promise 기반으로 만들어졌기 때문에 axios와 마찬가지로 데이터 다루기가 쉽고, 내장 라이브러리라는 장점으로 상당히 편리합니다.

💎 장점

- 자바스크립트의 내장 라이브러리 이므로 별도로 import 할 필요가 없음
- Promise 기반으로 만들어졌기 때문에 데이터 다루기 편리
- 내장 라이브러리가기 때문에 업데이트에 따른 에러 방지가 가능

💣 단점

- 지원하지 않는 브라우저가 존재 (IE11...)
- 네트워크 에러 발생 시 response timeout이 없어 기다려야 함

- JSON으로 변환해주는 과정 필요
- 상대적으로 axios에 비해 기능이 부족

코드

```
fetch("https://localhost:3000/user/post", {
  method: "POST",
  headers: {
    "Content-Type": "application/json",
  },
  body: JSON.stringify({
    id: "asd123",
    description: "hello world",
  }),
}).then((response) => console.log(response));
```

요약

간단한 통신의 경우엔 fetch를 사용하고 기능이 좀 더 필요하다면 axios를 사용하는 게 좋아 보입니다. 하지만 React-Native와 같이 업데이트가 잦은 경우에는 fetch가 내장 라이브러리이기 때문에 좀 더 안정적일 수도 있겠다고 생각합니다!

- axios 설치

```
npm install axios --save
```