

Proxy 사용하기

🕒 생성일	@2024년 11월 4일 오후 12:36
🏷️ 태그	

- 프록시(proxy) 서버
 - 클라이언트가 자신을 통해서 다른 네트워크 서비스에 간접적으로 접속 가능하게 해주는 응용 프로그램
 - 클라이언트와 서버 사이에서 HTTP 요청과 응답을 대신 처리해주는 역할을 하는 서버
- CORS(Cross-Origin Resource Sharing)란
 - 웹 애플리케이션에서 출처가 다른 곳으로 HTTP 요청을 보낼 수 없도록 브라우저가 요청을 막는 보안 정책
 - 출처(Origin) = Protocol + Host + Port
 - ▼ SOP(Same-Origin Policy)
 - 같은 출처에서만 리소스를 공유할 수 있도록 하는 정책
- CORS 해결방법
 1. 서버에서 접근을 허용
 2. 프록시 서버를 이용
- Proxy 설정하기
 - vue.config.js

```
// proxy 요청을 보낼 서버 주소
const target = "http://127.0.0.1:3000";

module.exports = {
  devServer: {
    port: 8080,
    proxy: {
      "^/api": {
        target,
        changeOrigin: true,
      },
    },
  },
};
```

- Vue에서 proxy 설정하고 클라이언트에서 HTTP 요청 (포트: 8080)하면 Proxy 서버가 웹 서버로 HTTP 요청 (포트: 3000)을 대신 함으로써 CORS 문제 해결
 - 운영 환경에서는 제대로 실행 안되므로 서버에서 처리 해야 함
- 서버에서 CORS 해결
 - API를 요청받는 서버에서 서버 헤더 중 Access-Control-Allow-Origin 필드에 CORS를 허용해 줄 도메인 주소를 등록해서 해결
 - 설정
 - Server

```
const cors = require("cors");

let corsOption = {
```

```
    origin: 'http://localhost:8080', // 허락하는 요청 주소
    credentials: true // true로 하면 설정 내용을 response 헤더에 추가해준다
}

app.use(cors(corsOption)); // CORS 미들웨어 추가
```

- Client

```
import axios from 'axios';
axios.defaults.baseURL = 'http://localhost:3000'; // 서버 주소
axios.defaults.headers.post['Content-Type'] = 'application/JSON;charset=utf-8';
axios.defaults.headers.post['Access-Control-Allow-Origin'] = '*';
```