

서버와 연결하기

```
//client 는 port가 3000번 대 이고,  
useEffect(() => {  
  axios.get('/api/hello')  
    .then(response => console.log(response.data))  
}, [])
```

```
//server는 port가 5000번대이다  
app.get('/api/hello', (req, res)=>{  
  res.send("안녕하세요~!")  
})
```

→ 서로간 cors에러가 발생 할 수 있음

Proxy를 통해서 해결하자!

1.npm install http-proxy-middleware --save


2.src/ setupProxy.js 생성

js파일에 아래와 같이 넣음

```
const { createProxyMiddleware } = require('http-proxy-middleware');  
  
module.exports = function(app) {  
  app.use(  
    '/api',  
    createProxyMiddleware({  
      target: 'http://localhost:5000',  
      changeOrigin: true,  
    })  
  );  
};
```

React 개발환경에서의 CORS를 위한 proxy 설정

React를 개발할 때, 보통 react-script start를 통해 React dev server를 띄어놓고 개발을 한다. React dev server의 접속 주소는 보통 http://localhost:3000 을 사용한다. 만약 api server의 주소가

 <https://velog.io/@tw4204/React-%EA%B0%9C%EB%B0%9C%ED%99%98%EA%B2%BD%EC%97%90%EC%84%9C%E>

```
Middleware } from './http-proxy-middleware';  
Options } from './types';  
  
createProxyMiddleware(context: Filter | Options,  
e } = new HttpProxyMiddleware(context, option  
;  
  
Options, RequestHandler } from './types';
```

C%9D%98-CORS%EB%A5%BC-%EC%9C%84%ED%95%9C-
proxy-%EC%84%A4%EC%A0%95

이를 nextJS에서도 proxy를 통한 해결

CORS를 조심해야됨

(보통 boot는 포트를 8080을 사용하게 되는데, client사용 port와 다름)

next.config.js

```
/** @type {import('next').NextConfig} */  
  
const nextConfig = {  
  reactStrictMode: true,  
  async rewrites() {  
    return [  
      {  
        source: '/api/:path*',  
        destination: 'http://172.28.1.247:8080/:path*',  
      },  
    ],  
  },  
}  
  
module.exports = nextConfig
```

사용예제

```
useEffect(() => {  
  axios.get('/api/test')  
    .then(response => setHello(response.data))  
    .catch(error => console.log(error))  
}, []);  
이런식으로  
http://localhost:3001/api/test 로 서버에 값을 찍을 수 있음
```

-> 이렇게 값을 json 타입으로 넘겨 보자

```
axios.get('URL').then().catch()  
axios.post('URL').then().catch()  
axos({  
  key: value  
  key2: value2  
})  
  
axios.post("/sample", {
```

```
id: this.state.id, pw: this.state.pw  
}).then(response => {console.log(response)});
```