

# CORS

hh99-R반-홍종훈

# CORS 오류란?

CORS (Cross-Origin Resource Sharing) 오류는 보안상의 이유로,

서로 다른 출처(origin) 간에 리소스를 공유할 때 발생하는 오류입니다.

# 출처(origin)란?

URL 스키마(http, https), 호스트 (도메인) 이름, 포트번호로 구성된 URL을 말합니다.

만약에, 웹사이트 A가 `http://example.com:3000`에 있고,  
웹사이트 B가 `https://example.com:4000`에 있다면,  
이 두 사이트는 서로 다른 출처(origin)입니다.

# CORS 오류를 해결하는 방법?

1. 서버 측에서 CORS 정책을 수정하거나
2. 클라이언트 측에서 CORS 우회 방법을 사용하여 서로 다른 출처 간에 리소스를 공유할 수 있습니다.

# 1. 서버에서 CORS정책을 수정하기

1-1.서버의 응답 헤더에 Access-Control-Allow-Origin 헤더 추가

1. 이 헤더는 클라이언트에서 요청한 출처(origin)에 대해 허용할 출처를 지정하는 역할을 합니다.
2. 예를 들어, Access-Control-Allow-Origin 헤더에 "http://localhost:3000"을 지정하면,
3. "http://localhost:3000" 출처에서 온 요청에 대해서만 응답을 허용하게 됩니다.

# 1. 서버에서 CORS정책을 수정하기

## 1-2. Access-Control-Allow-Headers 헤더 지정

- 예를 들어, 클라이언트에서 "Authorization" 헤더를 추가하여 요청을 보내는 경우,
- 서버에서는 Access-Control-Allow-Headers 헤더에 "Authorization"을 추가하여 해당 헤더에 대한 응답을 허용할 수 있습니다.

# 1.서버에서 CORS정책을 수정하기

## 1-2. Access-Control-Allow-Headers 헤더 지정

- "Content-Type", "Accept", "X-Requested-With" 등 다양한 요청 헤더를 지정할 수 있습니다.
- 서버에서 Access-Control-Allow-Headers 헤더를 적절히 설정하면, 클라이언트에서 보낸 요청 헤더에 대한 응답을 보다 자유롭게 제공할 수 있습니다.

# 1.서버에서 CORS정책을 수정하기

## 1-2. Access-Control-Allow-Headers 헤더 지정

1. "Content-Type" : 요청 본문의 데이터 타입을 명시합니다.
2. "Accept" : 클라이언트가 서버에 요청할 수 있는 응답의 데이터 타입을 명시합니다.
3. "X-Requested-With" : 클라이언트에서 보낸 요청이 Ajax 요청인지 여부를 판단합니다.

따라서 이들 헤더는 서버에서 마음대로 설정하기 보다는, 클라이언트와의 상호 운용성(interoperability)을 고려하여 표준적인 값을 사용하는 것이 좋습니다.



## 2. 클라이언트 측에서 CORS 우회 방법을 사용하기

```
const addPost = async (newPost) => {
  try {
    await axios.get(
      "https://cors-proxy-hj.herokuapp.com/http://3.38.191.164/user",
      {
        headers: {
          Authorization: `Bearer ${Cookies.get("auth")}`,
          "x-requested-with": "your-app-name",
        },
      },
    );

    await axios.post(`${process.env.REACT_APP_SERVER_URL}/posts`, newPost);
  } catch (error) {
    if (error.response && error.response.data) {
      toast.error(error.response.data.message);
    } else {
      toast.error(error.message);
    }
  }
};
```

**END**