Docker에서 Server 이미지 다운 받고 실행하기

```
docker pull ithinkso/testserver

docker run -p 8080:8080 ithinkso/testserver
```

build 아키텍처가 linux/amd64이므로

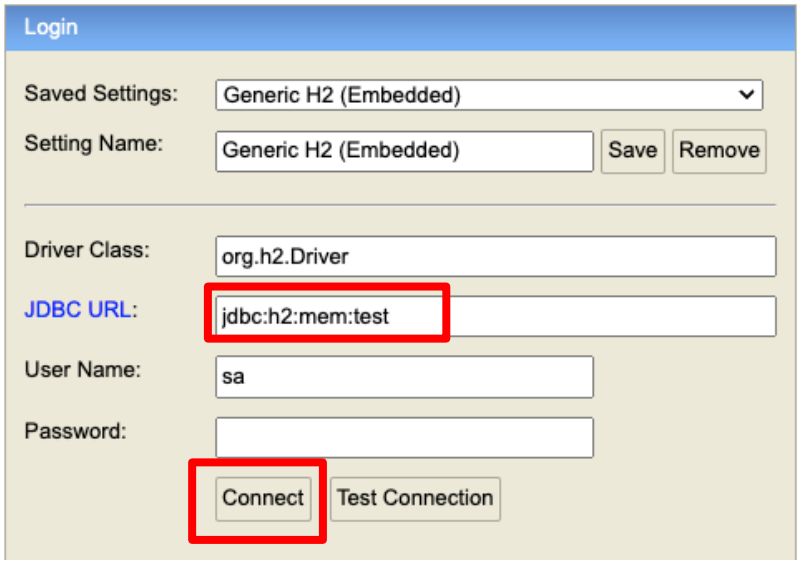
애플 arm cpu를 사용하시는 분은 실행이 느릴 수 있습니다.

실행 log에 아래와 같이 뜨면 서버 실행 완료입니다.

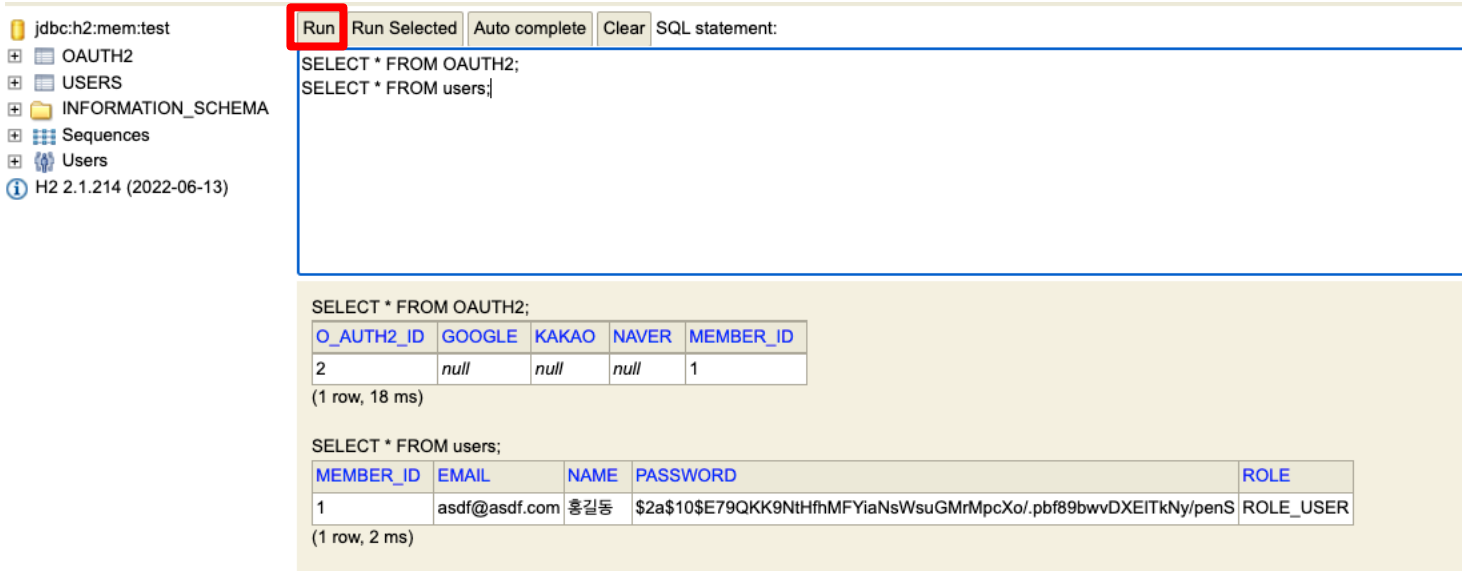
```
2022-07-08 03:53:19.928  INFO 1 --- [           main] o.s.m.s.b.SimpleBrokerMessageHandler     : Started.
2022-07-08 03:53:20.013  INFO 1 --
- [           main] com.example.demo.DemoApplication         : Started DemoApplication in 26.835 seconds (JVM running for 30.136)
2022-07-08 03:53:20.094  INFO 1 --- [           main] com.example.demo.InitDB                  : initialize database
```

DB 확인하기

http://localhost:8080/h2-console



초기 데이터가 들어가 있습니다.



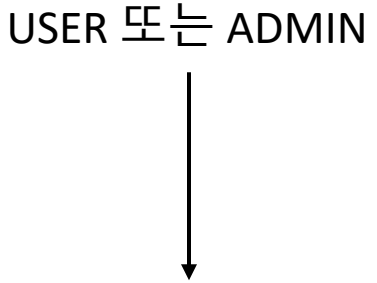
Test Api – postman을 이용하시면 편리하게 Api를 호출할 수 있습니다.

1. 회원가입

Url : http://localhost:8080/signup

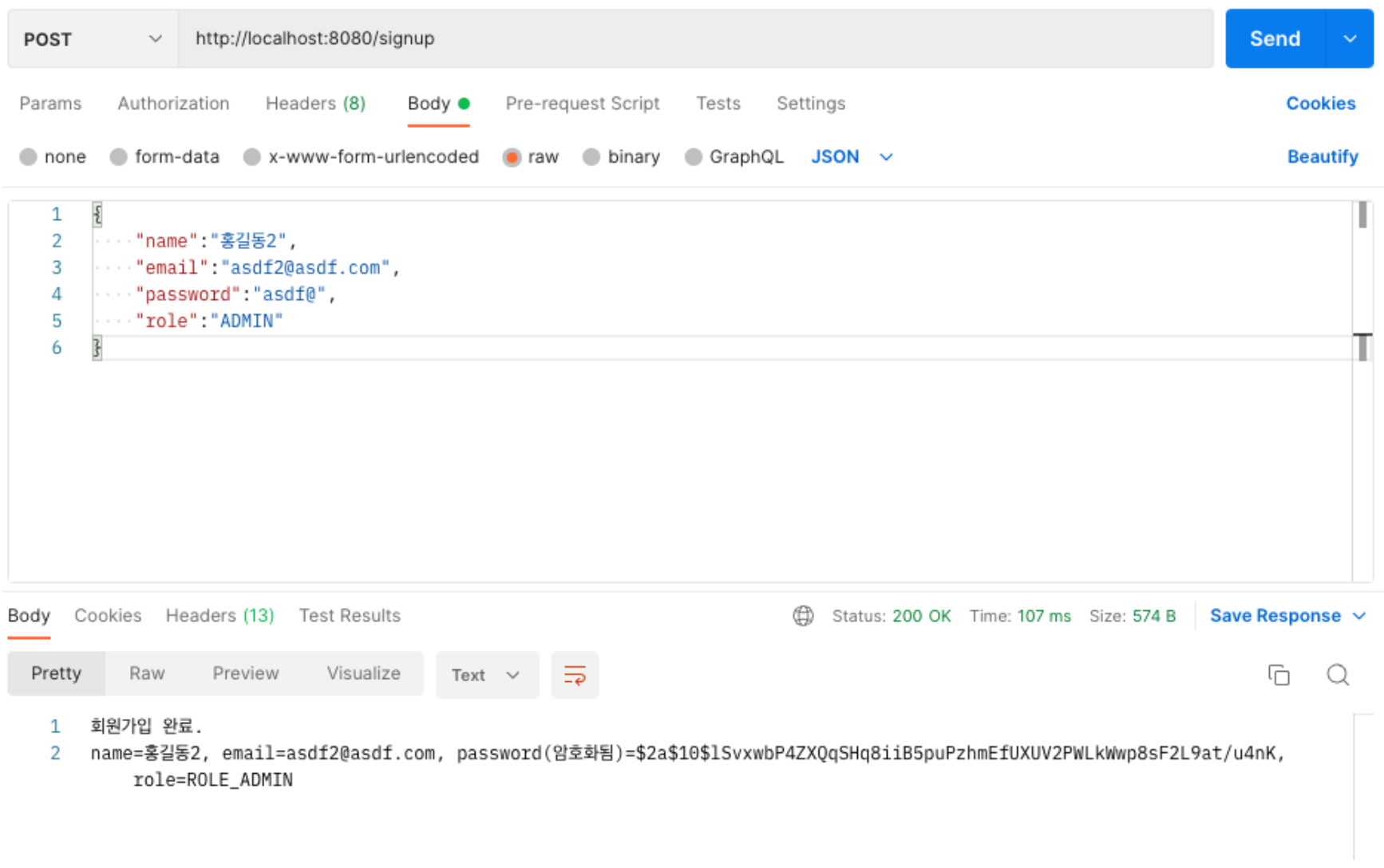
Method : Post

Body : name(string), email(string), password(string), role(string)



Test Api

1. 회원가입 예시



Test Api

2. 로그인

Url : http://localhost:8080/login

Method : Post

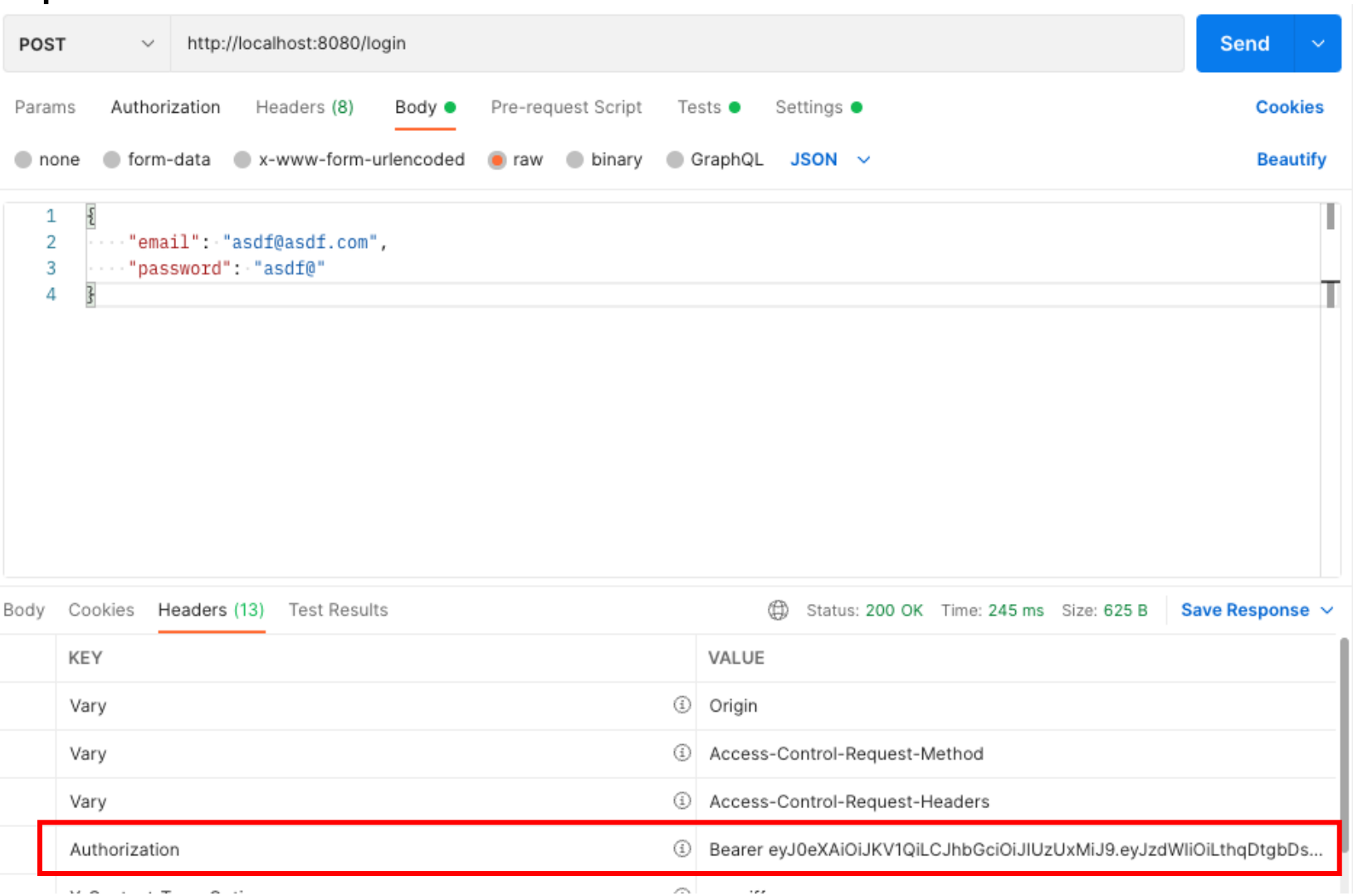
Body : email(string), password(string)

Response Header에 Authorization : Bearer Token(JWT)이 전송됩니다.

이후 로그인이 필요한 Api를 호출하실 때에는 반드시 Request Header에
해당 JWT가 포함되어 있어야 합니다.

Test Api

2. 로그인 예시



Test Api

3. User와 Admin이 접근 가능한 API

Url : http://localhost:8080/user

Method : Get

Header : Authorization에 JWT가 포함되어 있어야 합니다.

JWT 인증 실패 또는 접근 권한이 없으면 403 에러가 뜹니다.

Test Api

4. Admin만 접근 가능한 API

Url : http://localhost:8080/admin

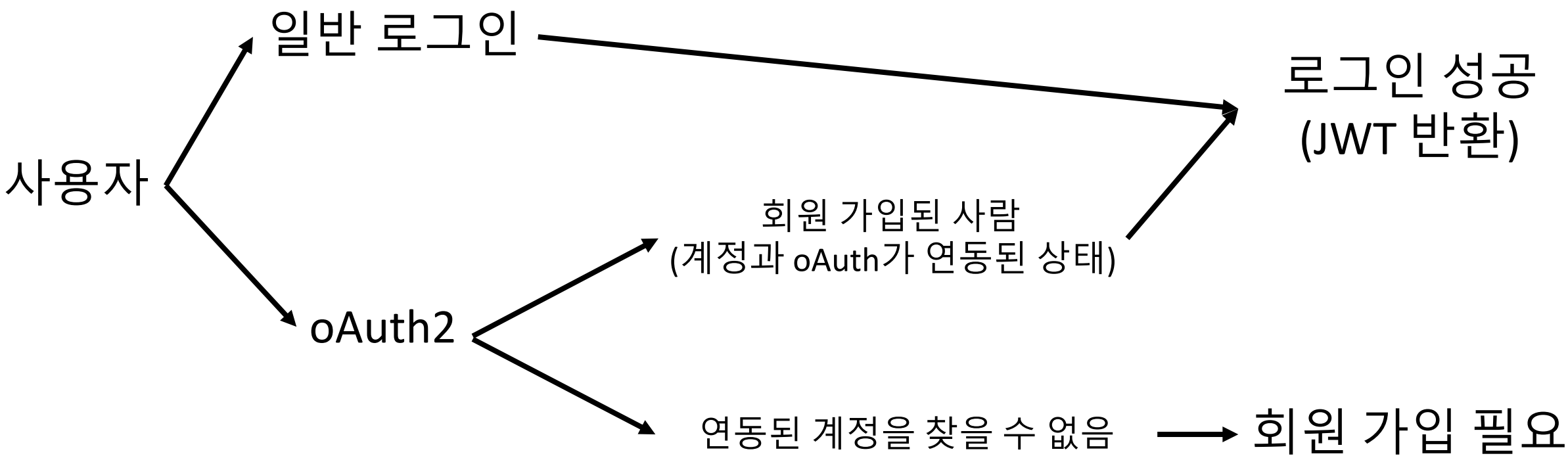
Method : Get

Header : Authorization에 JWT가 포함되어 있어야 합니다.

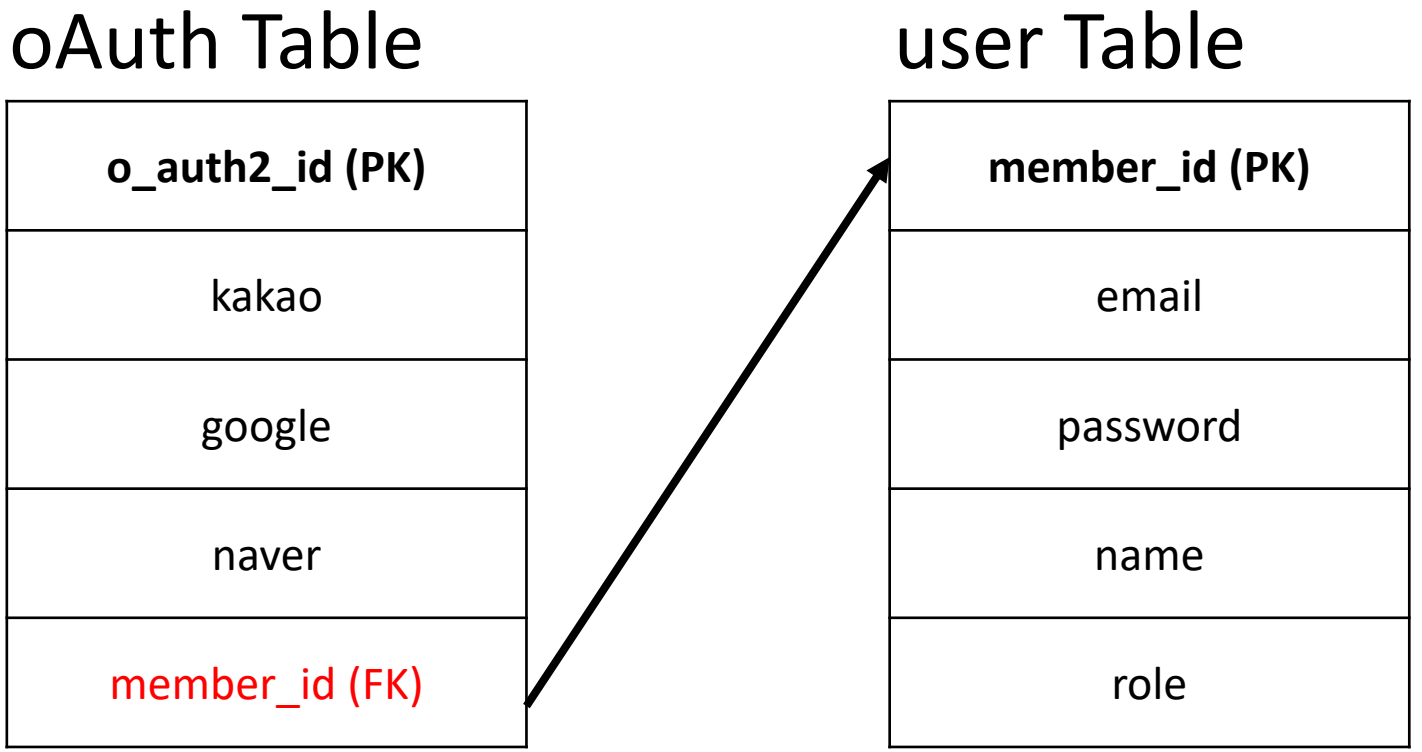
JWT 인증 실패 또는 접근 권한이 없으면 403 에러가 뜹니다.

OAuth2

로그인


일반 로그인
사용자
로그인 성공
(JWT 반환)
회원 가입된 사람
(계정과 oAuth가 연동된 상태)
oAuth2
연동된 계정을 찾을 수 없음
회원 가입 필요

DB Table 예시


oAuth Table
o_auth2_id (PK)
kakao
google
naver
member_id (FK)
user Table
member_id (PK)
email
password
name
role

1. 회원가입시

oAuth table과 user table이 1:1 관계로 생성

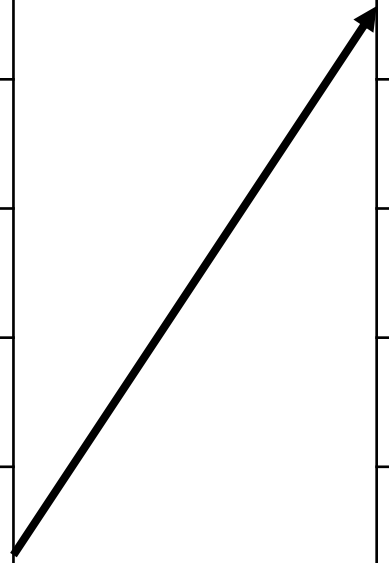
실제 서비스에서는
user1만 저장됩니다.

oAuth Table

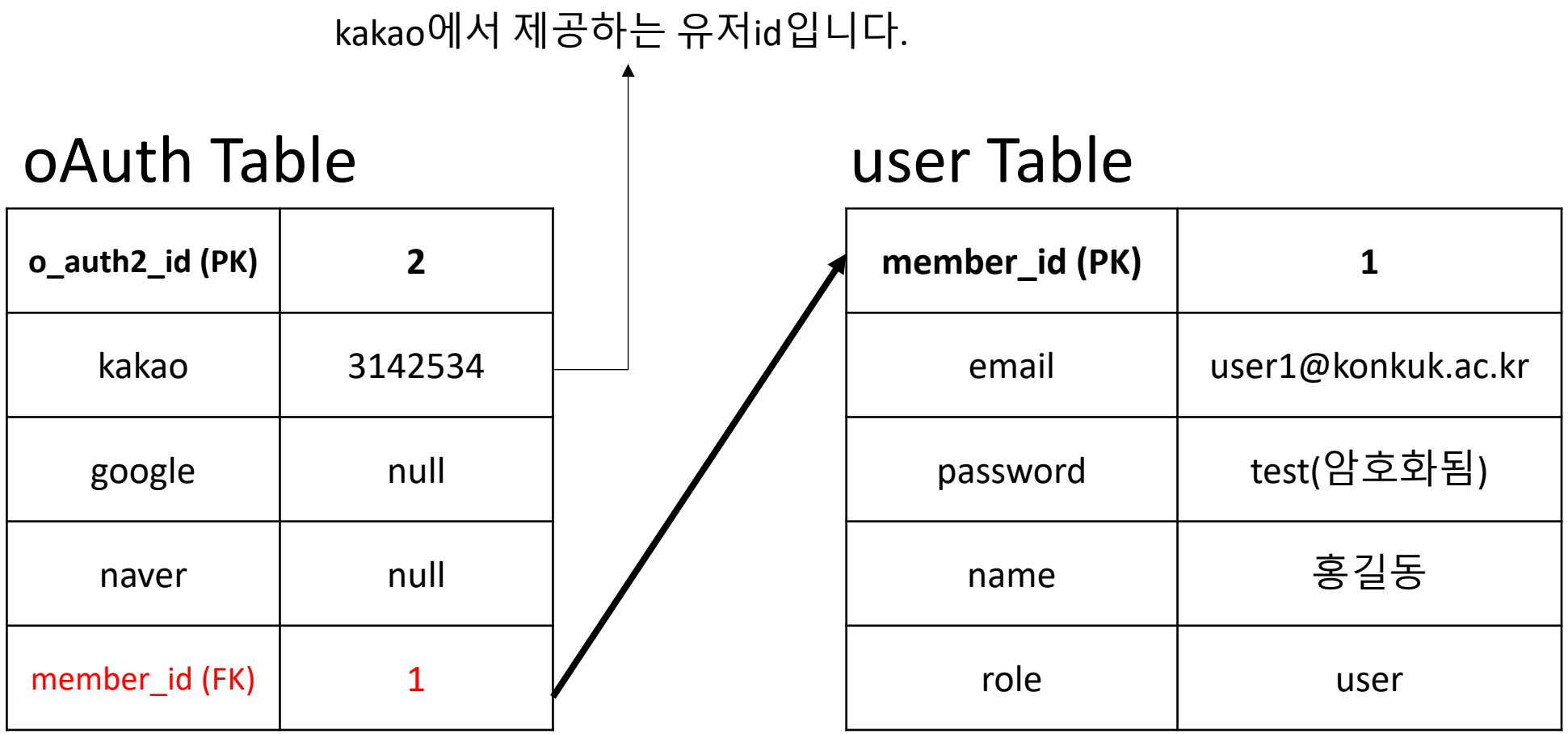
o_auth2_id (PK)	**2**
kakao	null
google	null
naver	null
member_id (FK)	1

user Table

member_id (PK)	**1**
email	user1@konkuk.ac.kr
password	test(암호화됨)
name	홍길동
role	user



2. oAuth와 연동(kakao)



oAuth2 Api

Url :카카오에서 제공하는 웹 페이지이므로 브라우저에서 접속해주세요.

https://kauth.kakao.com/oauth/authorize?client_id=437bc2fb95b24ca5a80d5763e4619f54&redirect_uri=http://localhost:8080/auth/kakao/callback&response_type=code

Method : Get

2가지 기능

1. 로그인 서비스 : oAuth Table 조회 후 존재하면 연동된 계정으로 로그인 성공.
   이때 계정과 oAuth가 연동된 상태이어야 합니다.

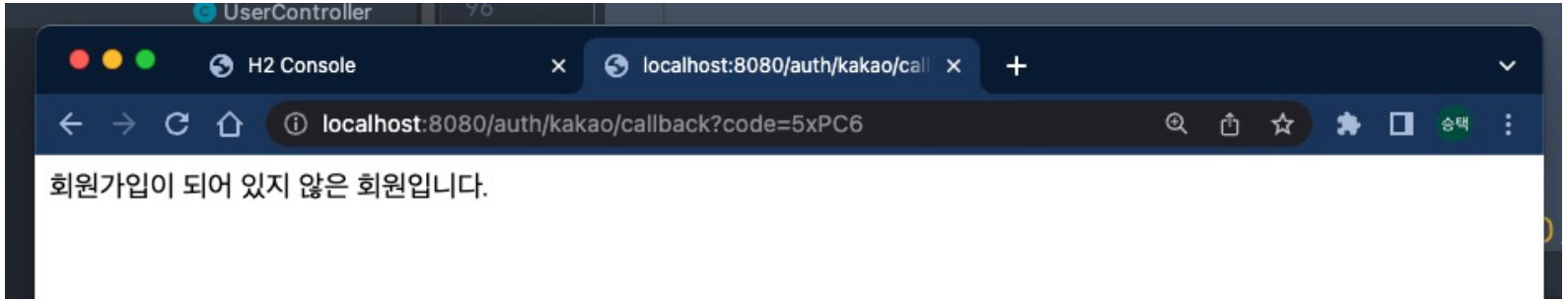
2. 계정과 연동 서비스 : Request header의 Authorization에 JWT가
   포함되어 있어야 합니다.

크롬 확장프로그램 ModHeader 사용하시면 request header 조작을 편하게 하실 수 있습니다.

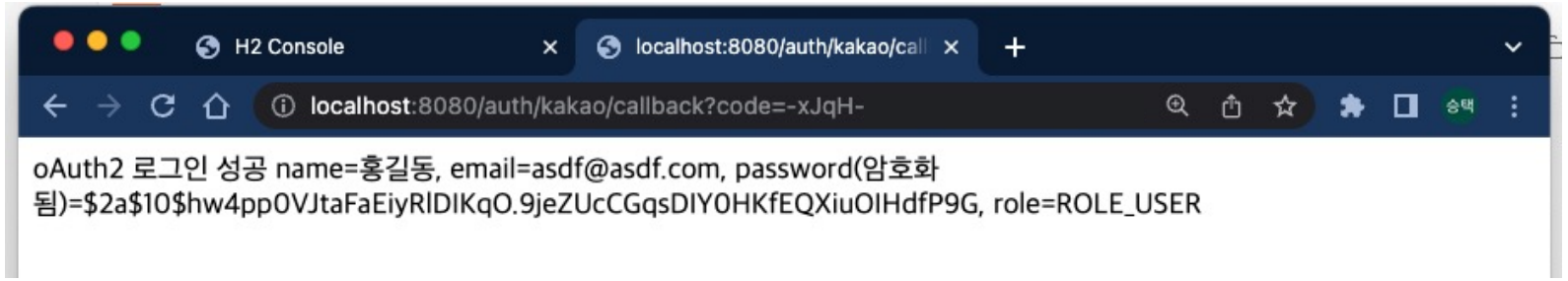
oAuth2 Api 예시

1. 로그인 서비스

1-1 계정 없음==해당 oAuth와 연동된 계정 없음

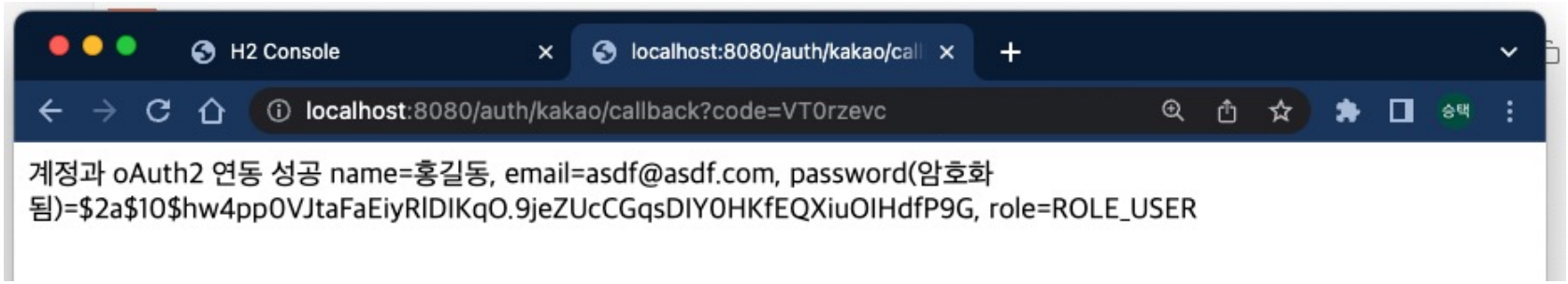


1-2 계정 존재==계정과 oAuth 연동된 상태. Response header에 JWT 확인



oAuth2 Api 예시

2. 계정과 연동 서비스 (header에 JWT가 포함되어 있어야 합니다.)



API 테스트 순서(종합)

1. oAuth 로그인 시도 -> 로그인 실패
2. 회원가입 (http://localhost:8080/signup)
3. 일반 로그인 (http://localhost:8080/login)
4. 접근권한이 필요한 서비스 이용 (http://localhost:8080/user 또는 /admin)
5. oAuth를 계정과 연동
6. oAuth 로그인 시도 -> 로그인 성공