

JWT 토큰 생성 및 검사

# Springboot JWT

참조 <https://github.com/dron512/mhkakaoJWT>

## 1.1 카카오 로그인

### 카카오 개발자 등록 및 애플리케이션 생성:

- 카카오 개발자 사이트(<https://developers.kakao.com/>)에 가입하고 로그인합니다.
- 내 애플리케이션 > 애플리케이션 추가하기를 통해 새로운 애플리케이션을 생성합니다.
- 생성한 애플리케이션의 앱 키(Client ID)와 리다이렉트 URI를 확인합니다.

← → ↺

developers.kakao.com/console/app/675279/d

앱

dron512/git231229

dron512/AI\_305\_7...

codingspecialist/S...

코딩의신스프링부트

스프링시큐리티

혼자공부하는머신러닝

머신러닝완벽가이드

AI\_501\_4/pmh/na...

모든 북마크

kakao developers

내 애플리케이션

제품

문서

도구

포럼

mungsunsang@kakao.com

KOR ENG

내 애플리케이션 > 앱 설정 > 플랫폼

앱 설정

요약 정보

일반

비즈니스

앱 키

플랫폼

앱 권한 신청

팀 관리

제품 설정

iOS

[가이드 확인하기](#)

iOS 플랫폼 등록

Web

삭제

수정

|         |                       |
|---------|-----------------------|
| 사이트 도메인 | http://localhost:8080 |
|---------|-----------------------|

• 카카오 로그인 사용 시 Redirect URI를 등록해야 합니다. [등록하러 가기](#)

## 1.1 카카오 로그인

리다이렉트 url 등록합니다.

### Redirect URI

#### Redirect URI

카카오 로그인에서 사용할 OAuth Redirect URI를 설정합니다.

여러개의 URI를 줄바꿈으로 추가해주세요. (최대 10개)

REST API로 개발하는 경우 필수로 설정해야 합니다.

예시: (O) https://example.com/oauth (X) https://www.example.com/oauth

http://localhost:8080/oauth/kakao/callback

### Redirect URI

삭제

수정

Redirect URI

http://localhost:8080/oauth/kakao/callback

- 카카오 로그인에서 사용할 OAuth Redirect URI를 설정합니다. (최대 10개)
- REST API로 개발하는 경우 필수로 설정해야 합니다.

REST API 키

38b99bb2df5447bcb6b585cfccbe97dc

취소

저장

# 1.1 카카오 로그인

보안 -> 클라이언트 시크릿 생성

개인정보 국외이전

연결 끊기

사용자 프로퍼티

보안

보안 이벤트

고급

메시지

카카오톡 채널

푸시 알림

고급 설정

허용 IP 주소

통계



명손상



ID 675279

OWNER

Web

카카오 로그인

ON

## Client Secret

삭제

토큰 발급 시, 보안을 강화하기 위해 Client Secret을 사용할 수 있습니다. (REST API인 경우에 해당)

코드

iHlwOUaEh408I4Tvi5ZMntOZJ2JRcZzl

재발급

활성화 상태

사용안함

설정

# 1.1 카카오 로그인

동의항목에서 동의

제품 설정

카카오 로그인

동의항목

간편가입

카카오톡 채널

개인정보 국외이전

연결 끊기

사용자 프로퍼티

보안

보안 이벤트

고급

메시지

카카오톡 채널

푸시 알림

로그인 설정

카카오 로그인 ON

[동의 화면 미리보기](#)

## 동의항목

카카오 로그인으로 서비스를 시작할 때 동의받는 항목을 설정합니다. 미리보기를 통해 사용자에게 보여질 화면을 확인할 수 있습니다. 사업자 정보를 등록하고 비즈니스 인증을 완료하면, 권한이 필요한 동의항목에 대한 심사를 신청할 수 있습니다.

개인정보 동의항목 심사 신청

## 개인정보

| 항목 이름       | ID               | 상태                         |
|-------------|------------------|----------------------------|
| 닉네임         | profile_nickname | ● 필수 동의 <a href="#">설정</a> |
| 프로필 사진      | profile_image    | ● 필수 동의 <a href="#">설정</a> |
| 카카오 계정(이메일) | account_email    | ● 선택 동의 <a href="#">설정</a> |

# 1.1 카카오 로그인

동의항목에서 동의

카카오 로그인

동의항목

간편가입

카카오톡 채널

개인정보 국외이전

연결 끊기

사용자 프로퍼티

보안

보안 이벤트

고급

메시지

카카오톡 채널

푸시 알림

고급 설정

허용 IP 주소



명순상

ID 675279

OWNER

Web

## 허용 서버 IP 주소

IP 주소를 지정한 후에는 REST API 키 혹은 Admin 키로 API를 호출할 경우 아래에 지정한 IP 주소로만 호출이 가능합니다.

IP(IPv4, IPv6), CIDR(IP/CIDR), Subnet Mask(IP/Mask)를 제공합니다.

허용 IP 주소 등록

## 1.1 카카오 로그인

### 1. application.yml 파일 수정

```
kakao:
  client:
    id: 38b99bb2df5447bcb6b585cfccbe97dc #rest api 키
    secret: iHIw0UaEh408I4TvI5ZMnt0ZJ2JRcZz1 # 클라이언트 시크릿 키
```

# 1.1 카카오 로그인

카카오 로그인  
문서 읽어보기

kakao developers

내 애플리케이션

제품

문서 > 카카오 로그인 > REST API

카카오 로그인

이해하기

설정하기

REST API

JavaScript

Android

iOS

Flutter

보안 이벤트 구독

콜백

디자인 가이드

FAQ

문제 해결

카카오싱크

커뮤니케이션

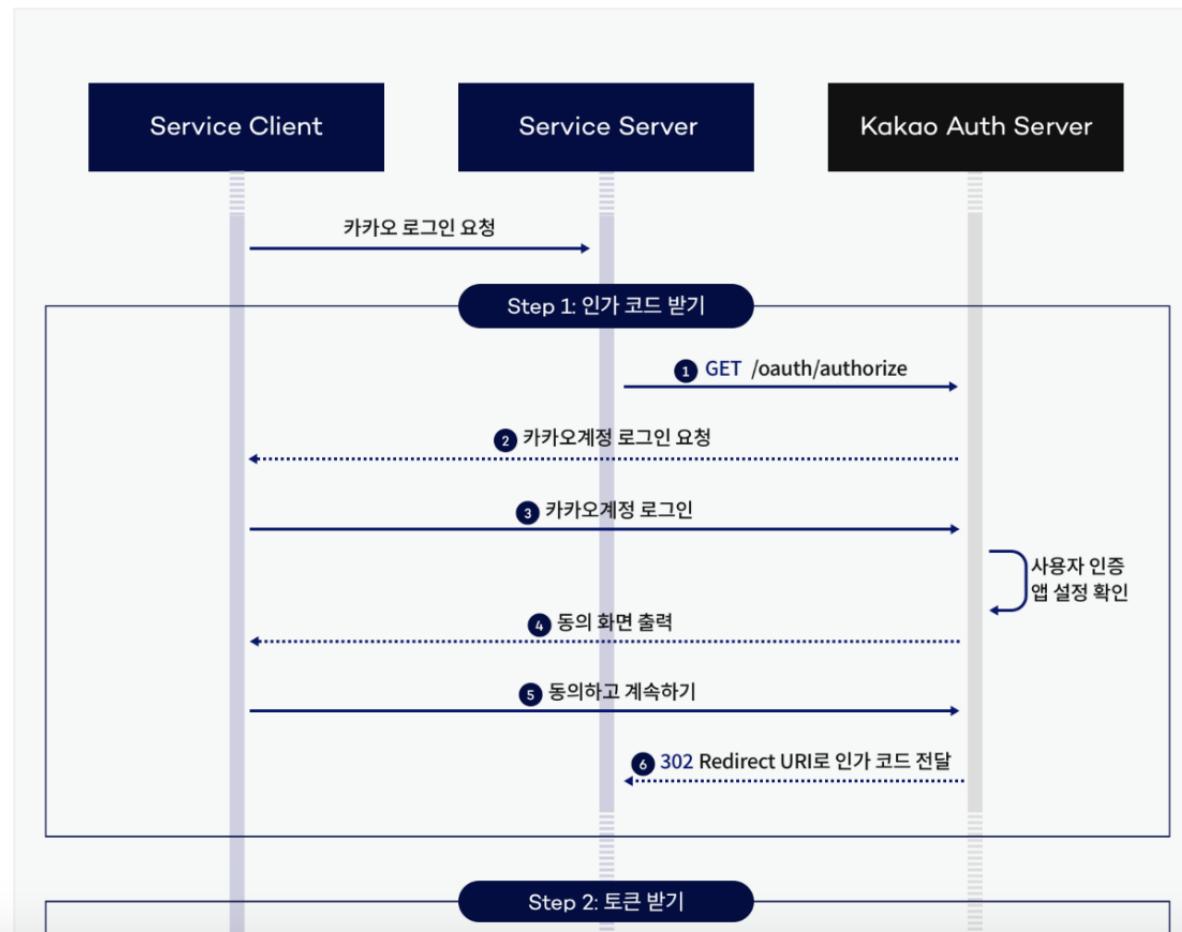
메시지

카카오톡 소셜

카카오톡 채널

## 카카오 로그인 과정

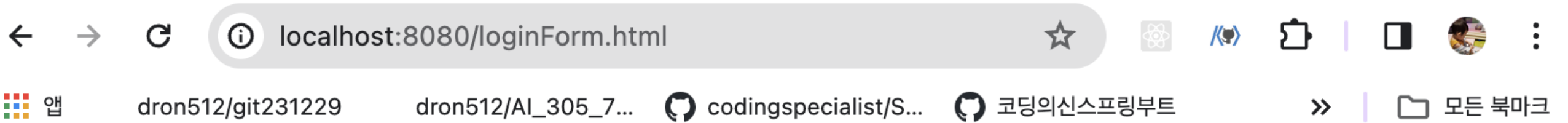
REST API를 사용한 카카오 login은 PC 및 모바일 웹에서 카카오 로그인 구현 시 적합한 방식입니다. 다음은 REST API를 사용한 카카오 로그인 과정을 나타낸 시퀀스 다이어그램(Sequence diagram)입니다. 단계별 안내를 함께 참고합니다.





# 1.1 카카오 로그인

1. 프론트 페이지 만들기



카카오 로그인

## 1.1 카카오 클래스 설계

### 1. OAuthLoginDto

프론트에서 Kakao 로그인을 위한 토큰 보내올시

### 1. KakaoController

인가코드를 이용한 카카오토큰 발행을 위한 리다이렉트 설정

### 2. KakaoLoginApiService

카카오토큰을 이용한 카카오 정보를 요청

### 3. KakaoTokenClient

카카오토큰 발행을 위한 오픈페인 인터페이스

### 4. KakaoTokenDto

인가코드 Request 설정

토큰코드 Response 설정

### 5. KakaoUserInfoClient

카카오토큰으로 카카오 정보를 가져오는 오픈페인 인터페이스

### 6. KakaoUserInfoResponseDto

카카오 정보를 저장하는 클래스

OAuthAttributes

#### ▼ oauth

© OAuthLoginDto

© OAuthLoginService

© KakaoController

© KakaoLoginApiService

① KakaoTokenClient

© KakaoTokenDto

① KakaoUserInfoClient

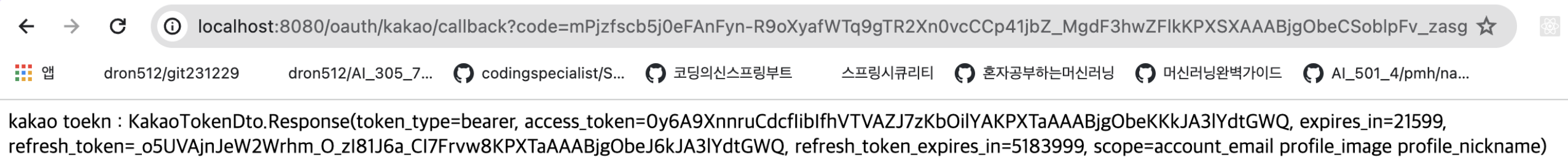
© KakaoUserInfoResponseDto

© OAuthAttributes

## 1.1 카카오 로그인

로그인 성공시

AccessToken RefreshToken 받아올수있다.



오픈 페인 클라이언트를 이용해서 카카오에 요청 하도록 한다.

카카오 로그인 페이지 요청

1. <https://github.com/dron512/mhkakaoJWT/blob/main/src/main/resources/static/loginForm.html>
2. 인가코드를 이용한 액세스 토큰 요청
3. 액세스 토큰을 이용한 카카오 정보 가져오기

## 1.1 카카오 로그인

```
@GetMapping("/oauth/kakao/callback")
public @ResponseBody String loginCallback(String code) {
    String contentType = "application/x-www-form-urlencoded;charset=utf-8";
    KakaoTokenDto.Request kakaoTokenRequestDto = KakaoTokenDto.Request.builder()
        .client_id(clientId)
        .client_secret(clientSecret)
        .grant_type("authorization_code")
        .code(code)
        .redirect_uri("http://localhost:8080/oauth/kakao/callback")
        .build();
    KakaoTokenDto.Response kakaoToken = kakaoTokenClient.requestKakaoToken(contentType, kakaoTokenRequestDto);
    return "kakao token : " + kakaoToken;
}
```

## 1.1 카카오 로그인

회원 테이블 설계

| member_id             | 회원아이디    |
|-----------------------|----------|
| member_type           | 회원타입     |
| email                 | 이메일      |
| password              | 비밀번호     |
| member_name           | 회원이름     |
| profile               | 프로필      |
| role                  | 역할       |
| refresh_token         | 리프레시 토큰  |
| token_expiration_time | 토큰 만료 시간 |
| create_by             | 생성자      |
| create_time           | 생성시간     |
| modified_by           | 수정자      |
| update_time           | 수정시간     |

## 1.1 카카오 로그인

Hibernate:

```
drop table if exists member cascade
```

Hibernate:

```
create table member (  
    create_time timestamp(6),  
    member_id bigint generated by default as identity,  
    token_expiration_time timestamp(6),  
    update_time timestamp(6),  
    member_type varchar(10) not null check (member_type in ('KAKAO')),  
    role varchar(10) not null check (role in ('USER','ADMIN')),  
    member_name varchar(20) not null,  
    email varchar(50) not null unique,  
    password varchar(200),  
    profile varchar(200),  
    refresh_token varchar(250),  
    created_by varchar(255),  
    modified_by varchar(255),  
    primary key (member_id)  
)
```

## 1.1 카카오 로그인

← → ↻ ⓘ localhost:8080/api/token-test/create

앱 dron512/git231229 dron512/AI\_305\_7... AI\_501\_4/pmh/na... jwt최근방식 JWT생성 박해선

```
1 // 20240309124909
2 // http://localhost:8080/api/token-test/create
3
4 {
5   "grantType": "Bearer",
6   "accessToken":
7     "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJzdWIiOiJBQ0NFU1MiLCJpYXQiOiE3MDk5
8     1JTIJ9.52rvo2-xpN18TyX_0S8K6l9a5xNskAgQ7GiUk53ktdE",
9   "accessTokenExpireTime": "2024-03-09 13:04:09",
10  "refreshToken":
11    "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJzdWIiOiJSRUZSRVNIIiwiaWF0IjoxNzA5
12    MIZUE5WQgFCjaWuRow8Qk7ffzeZdDqC4",
13  "refreshTokenExpireTime": "2024-03-23 12:49:09"
14 }
```

## 1.1 TokenManager

Refresh Token, accessToken

액세스 토큰(access token)과 리프레시 토큰(refresh token)은 인증 및 권한 부여를 위해 사용되는 보안 토큰입니다. 이 두 가지 토큰은 주로 OAuth 2.0 프로토콜에서 사용되며, 사용자의 인증 및 권한 부여를 관리하는 데 중요한 역할을 합니다.

액세스 토큰(Access Token):

액세스 토큰은 사용자가 리소스에 접근할 때 사용되는 토큰입니다.

주로 OAuth 2.0 인증 서버에 의해 발급됩니다.

액세스 토큰은 일반적으로 짧은 유효 기간을 가지며(예: 몇 분 또는 몇 시간), 발행 후 일정 시간이 지나면 만료됩니다.

만료된 경우, 사용자는 새로운 액세스 토큰을 받기 위해 인증 서버에 다시 인증해야 합니다.

리프레시 토큰(Refresh Token):

리프레시 토큰은 액세스 토큰을 갱신하기 위해 사용되는 토큰입니다.

주로 OAuth 2.0 인증 서버에 의해 발급됩니다.

리프레시 토큰은 액세스 토큰보다 긴 유효 기간을 가집니다(예: 일반적으로 몇 일 또는 몇 주).

액세스 토큰이 만료되었을 때, 사용자는 리프레시 토큰을 사용하여 새로운 액세스 토큰을 요청할 수 있습니다.

리프레시 토큰은 보안 상의 이유로 액세스 토큰보다 안전한 저장소에 보관되어야 합니다.



## 1.1 Token 발급

TokenManager 만들기

<https://github.com/dron512/mhkakaoJWT/blob/main/src/main/java/org/mh/studyspringboots/security/iwt/TokenManager.java>

1 usage · dron512

```
public JwtTokenDto createJwtTokenDto(Long memberId, Role role) {  
    Date accessExpireTime = null;  
    Date refreshExpireTime = null;  
    String accessToken = null;  
    String refreshToken = null;  
    accessToken = createAccessToken(memberId, role, accessTokenExpireTime);  
    refreshToken = createRefreshToken(memberId, refreshTokenExpireTime);  
  
    return JwtTokenDto.builder()  
        .grantType(GrantType.BEARER.getType())  
        .accessToken(accessToken)  
        .accessTokenExpireTime(accessTokenExpireTime)  
        .refreshToken(refreshToken)  
        .refreshTokenExpireTime(refreshTokenExpireTime)  
        .build();  
}
```

# 1.1 Token 발급

토큰 생성

<https://github.com/dron512/mhkakaoJWT/blob/main/src/main/java/org/mh/studyspringboots/ecurity/jwt/TokenTestController.java>

GET

http://localhost:8080/api/token-test/create

Send

Params

Authorization

Headers (7)

Body

Pre-request Script

Tests

Settings

Cookies

Query Params

| <input type="checkbox"/> | Key | Value | Description | ... | Bulk Edit |
|--------------------------|-----|-------|-------------|-----|-----------|
| <input type="checkbox"/> | id  |       |             |     |           |
|                          | Key | Value | Description |     |           |

Body

Cookies

Headers (14)

Test Results

200 OK 15 ms 931 B

Save as example

...

Pretty

Raw

Preview

Visualize

JSON

```
1 {
2   "grantType": "Bearer",
3   "accessToken": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJzdWIiOiJBQ0NFU1MiLCJpYXQiOiE3MDk5NTYzMTIsImV4cCI6MTcwOTk1NzIxMiwibWVtYmVySWQiOiEsInJvbGUiOiJBRE1JTjI9.1kvs4CpRPC6BLfSb4FPtfGXe7o009s9bBmagE7RUa78",
4   "accessTokenExpiresTime": "2024-03-09 13:06:52",
5   "refreshToken": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJzdWIiOiJSRUZSRVNIiwiiaWF0IjoxNzA5OTU2MzEyLCJleHAiOiE3MTE5NjU5MTIsIm1lbWJlcklkIjoxfQ.uqVSFyHTfM5GFD80XvIfYaUdPu6XEMCbE_72i0wrhk",
6   "refreshTokenExpiresTime": "2024-03-23 12:51:52"
7 }
```

## 1.1 Token 발급

토큰 검증

<https://github.com/dron512/mhkakaoJWT/blob/main/src/main/java/org/mh/studyspringboots/ecurity/jwt/TokenTestController.java>

GET

▼

http://localhost:8080/api/token-test/valid?token=eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJzdWliOi...

Send

▼

Params ● Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

Query Params

| <input checked="" type="checkbox"/> | Key   | Value                                   | Description | ⋮ Bulk Edit |
|-------------------------------------|-------|---|-------------|-------------|
| <input checked="" type="checkbox"/> | token | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9... |             |             |
|                                     | Key   | Value                                   | Description |             |

Body Cookies Headers (14) Test Results

200 OK 71 ms 435 B Save as example ⋮

Pretty Raw Preview Visualize Text ▼

1 success



## 1.1 Token 발급

토큰 검증

<https://github.com/dron512/mhkakaoJWT/blob/main/src/main/java/org/mh/studyspringboots/ecurity/jwt/TokenTestController.java>

# 1.1 카카오 로그인

토큰 검증

<https://github.com/dron512/mhkakaoJWT/blob/main/src/main/java/org/mh/studyspringboots/ecurity/jwt/TokenTestController.java>

HTTP macapi / http://localhost:8080/login

Save

POST

http://localhost:8080/login

Send

Params

Authorization

Headers (10)

Body

Pre-request Script

Tests

Settings

Cookies

Type

Bearer Tok...

The authorization header will be automatically generated when you send the request. Learn more about [Bearer Token](#) authorization.

ⓘ Heads up! These parameters hold sensitive data. To keep this data secure while working in a collaborative environment, we recommend using variables. Learn more about [variables](#).

Token

bpK71i-9XpckWdryYviuuXBrMVc4loy3IG0KI...

# 1.1 카카오 로그인

토큰 검증

<https://github.com/dron512/mhkakaoJWT/blob/main/src/main/java/org/mh/studyspringboots/ecurity/jwt/TokenTestController.java>

GET

▼

http://localhost:8080/api/token-test/valid?token=eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJzdWliOi...

Send

▼

Params ● Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

Query Params

| <input checked="" type="checkbox"/> | Key   | Value                                   | Description | ⋮ Bulk Edit |
|-------------------------------------|-------|---|-------------|-------------|
| <input checked="" type="checkbox"/> | token | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9... |             |             |
|                                     | Key   | Value                                   | Description |             |

Body Cookies Headers (14) Test Results

200 OK 71 ms 435 B Save as example ⋮

Pretty Raw Preview Visualize Text ▼

1 success

# 1.1 카카오 로그인

토큰 검증

<https://github.com/dron512/mhkakaoJWT/blob/main/src/main/java/org/mh/studyspringboots/ecurity/jwt/TokenTestController.java>

GET

▼

http://localhost:8080/api/token-test/valid?token=eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJzdWliOi...

Send

▼

Params ● Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

Query Params

| <input checked="" type="checkbox"/> | Key   | Value                                   | Description | ⋮ Bulk Edit |
|-------------------------------------|-------|---|-------------|-------------|
| <input checked="" type="checkbox"/> | token | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9... |             |             |
|                                     | Key   | Value                                   | Description |             |

Body Cookies Headers (14) Test Results

200 OK 71 ms 435 B Save as example ⋮

Pretty Raw Preview Visualize Text ▼

1 success

# 1.1 카카오 로그인

토큰 검증

<https://github.com/dron512/mhkakaoJWT/blob/main/src/main/java/org/mh/studyspringboots/ecurity/jwt/TokenTestController.java>

GET

▼

http://localhost:8080/api/token-test/valid?token=eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJzdWliOi...

Send

▼

Params ● Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

Query Params

| <input checked="" type="checkbox"/> | Key   | Value                                   | Description | ⋮ Bulk Edit |
|-------------------------------------|-------|---|-------------|-------------|
| <input checked="" type="checkbox"/> | token | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9... |             |             |
|                                     | Key   | Value                                   | Description |             |

Body Cookies Headers (14) Test Results

200 OK 71 ms 435 B Save as example ⋮

Pretty Raw Preview Visualize Text ▼

1 success



# 1.1 카카오 로그인

토큰 검증

<https://github.com/dron512/mhkakaoJWT/blob/main/src/main/java/org/mh/studyspringboots/ecurity/jwt/TokenTestController.java>

GET

▼

http://localhost:8080/api/token-test/valid?token=eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJzdWliOi...

Send

▼

Params ● Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

Query Params

|                                     |       |   |             |             |
|-------------------------------------|-------|---|-------------|-------------|
| <input checked="" type="checkbox"/> | Key   | Value                                   | Description | ⋮ Bulk Edit |
| <input checked="" type="checkbox"/> | token | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9... |             |             |
|                                     | Key   | Value                                   | Description |             |

body Cookies Headers (14) Test Results

200 OK 71 ms 435 B Save as example

Pretty Raw Preview Visualize Text ▼

1 success

## 1.1 카카오 로그인

토큰 검증

<https://github.com/dron512/mhkakaoJWT/blob/main/src/main/java/org/mh/studyspringboots/ecurity/jwt/TokenTestController.java>

[http://k.kakaocdn.net/dn/dhtMw8/btsDWsZxQIW/uQ3rsKSzTVdHE1mTsFTvkK/img\\_110x110.jpg](http://k.kakaocdn.net/dn/dhtMw8/btsDWsZxQIW/uQ3rsKSzTVdHE1mTsFTvkK/img_110x110.jpg)

```
userInfo : OAuthAttributes(name=헬로우, email=mungsunsang@kakao.com, profile=http://k.kakaocdn.net/dn/dhtMw8/btsDWsZxQIW/uQ3rsKSzTVdHE1mTsFTvkK/img\_110x110.jpg,
```

고맙습니다.

