

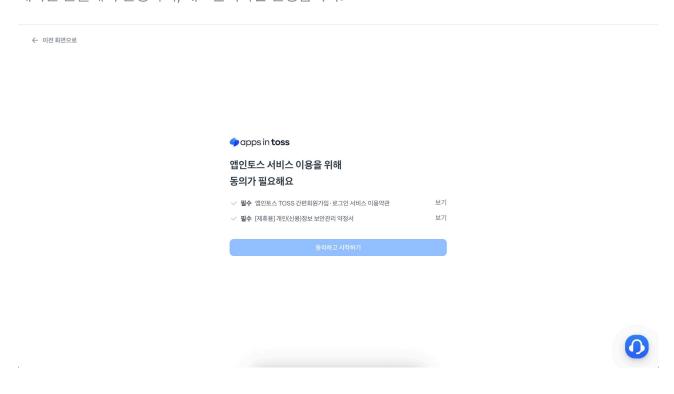
Menu On this page

토스 로그인 연동하기

앱인토스에서 토스 회원을 연동해보세요.

#계약하기

토스 로그인 사용을 위해서는 계약이 필요해요. 계약은 콘솔에서 진행하며, 대표관리자만 진행합니다.



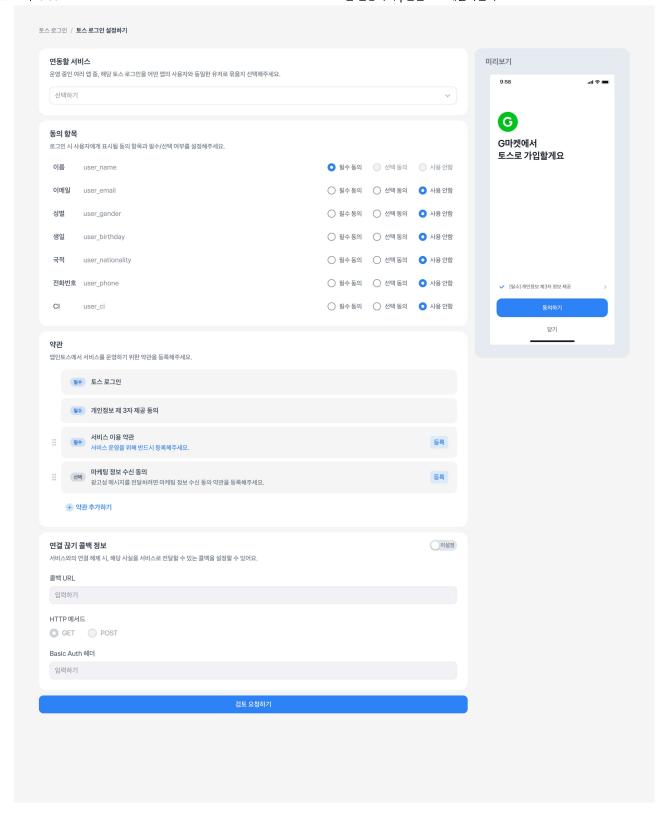
설정하기

로그인 연동을 위해서는 콘솔에 사전 셋팅이 필요해요.

콘솔에 입력한 정보를 바탕으로 사용자가 진행할 로그인 약관 화면을 구성합니다.

우측에 사용자가 진행할 로그인 약관 화면을 볼 수 있어요.

모든 설정을 완료하셨다면 검토를 요청하세요. 영업일 기준 3일 이내 승인해드려요.



기존에 사용하던 로그인 key와 연동하기

파트너사 자체 앱/웹 로그인을 이미 사용하고 있는 경우, 앱인토스 토스 로그인과 연동을 할 수 있어요. 목록에서 Client ID를 선택하면, 앱인토스 토스 로그인 사용 시 파트너사 자체 앱/웹 로그인의 userKey 와 동일하게 내려갑니다.

userKey 는 회원 별로 유니크하게 관리되는 값으로 쉽게 회원을 관리할 수 있어요.

추후 파트너사 자체 앱/웹 로그인을 사용하고 싶어요.

파트너사 자체 앱/웹 로그인 연동하기를 확인하시어 Client ID를 발급해주세요.

파트너사 자체 앱/웹 로그인 Client ID가 발급되었다면, 콘솔에서 연동하여 앱인토스 토스 로그인의 userKey 와 동일하게 사용할 수 있어요.

사용자 권한 범위 설정하기

토스 로그인을 통해 수집하려는 사용자의 권한 범위를 설정해주세요.

- 이름(USER_NAME): 사용자의 이름
- 이메일(USER_EMAIL): 사용자의 이메일로 토스 가입 시 필수사항이 아니기 때문에 값이 존재하지 않을 수 있어요. 그럴 경우 null로 전달드려요.
- 성별(USER GENDER): 사용자의 성별
- 생일(USER BIRTHDAY): 사용자의 생년월일
- 국적(USER NATIONALITY): 사용자의 국적
- 전화번호(USER_PHONE): 사용자의 전화번호
- CI(USER CI): 사용자를 식별하는 KEY값인 CI(Connection Information)

약관 설정하기

앱인토스에서 서비스를 운영하기 위한 약관을 등록해주세요.

토스 로그인 서비스의 약관과 개인정보 제 3자 제공동의는 자동으로 등록됩니다.

다만, 파트너사의 서비스 약관 및 개인정보 수집약관 그리고 필요시 마케팅 정보 수신 동의는 추가로 등록 해주셔야 합니다.

활용하고자 하는 목적에 따라 정확한 동의문 링크를 첨부해 주세요.



간편 로그인 검토 가이드라인을 참고해주세요.

가이드라인에 부합하지 않을 경우 반려될 수 있어요.

연결 끊기 콜백 정보 설정하기

사용자가 토스앱에서 로그인을 끊었을 경우 콜백을 받을 수 있어요.



사용자가 토스앱에서 로그인 끊기를 할 경우, 토스에서는 사용자의 동의 약관 정보나 로그인 정보 등을 모두 삭제합니다.

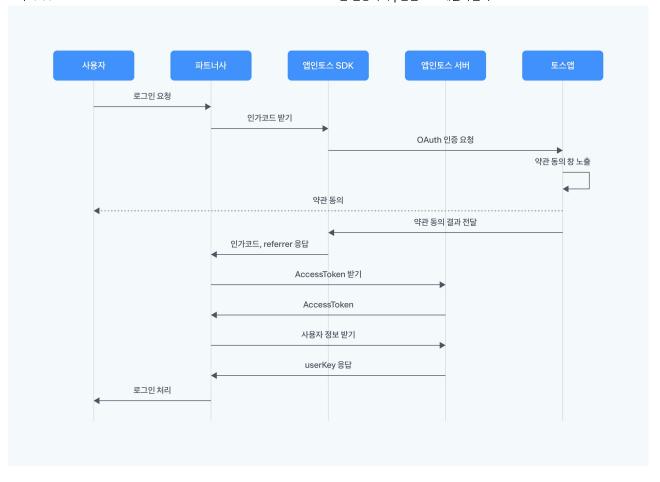
개발을 위한 키 확인하기



토스 로그인 연동이 승인되었다면 개발을 위한 키 확인을 할 수 있어요. 복호화 키는 **이메일로 키 받기** 버튼을 통해 이메일로 받을 수 있어요.

- Client ID
- 복호화 키

개발하기



1. 인가 코드 받기

SDK를 통해 연동해주세요.

사용자의 인증을 요청하고, 사용자가 인증에 성공하면 인가 코드를 메소드 응답으로 전달드려요. appLogin 함수를 사용해서 인가 코드(authorizationCode)와 referrer 를 받을 수 있어요. appLogin를 확인해 주세요.

- 샌드박스앱에서는 referrer 가 sandbox 가 반환돼요
- 토스앱에서는 referrer 가 DEFAULT 가반환돼요

🕍 잠시만요

인가코드의 유효시간은 10분입니다.

토스 로그인을 처음 진행할 때

appLogin 함수를 호출하면 토스 로그인 창이 열리고, 앱인토스 콘솔에서 등록한 약관 동의 화면이 노출 돼요.

사용자가 필수 약관에 동의하면 인가 코드가 반환돼요.

토스 로그인을 이미 진행했을 때

appLogin 함수를 호출하면 별도의 로그인 창 없이 바로 인가 코드가 반환돼요.

2. AccessToken 받기

사용자 정보 조회 API 를 사용하기 위한 접근 토큰을 발급합니다.

• Content-type: application/json

• Method: POST

• URL: /api-partner/v1/apps-in-toss/user/oauth2/generate-token

🕍 잠시만요

AccessToken 유효시간은 1시간이에요.

요청

이름	타입	필수값 여부	설명
authorizationCode	string	Υ	인가코드
referrer	string	Υ	referrer

성공 응답

이름	타입	필수값 여부	설명
tokenType	string	Υ	bearer 로 고정
accessToken	string	Υ	accessToken
refreshToken	string	Υ	refreshToken

이름	타입	필수값 여부	설명
expiresIn	string	Υ	만료시간(초)
scope	string	Υ	인가된 scope(구분)

```
interval try to the state of the state
```

실패 응답

인가 코드가 만료되었거나 동일한 인가 코드로 AccessToken 을 중복으로 요청할 경우

```
json
{
    "error":"invalid_grant"
}

    "resultType":"FAIL",
    "error":{
        "errorCode":"INTERNAL_ERROR",
        "reason":"요청을 처리하는 도중에 문제가 발생했습니다."
    }
}
```

3. AccessToken 재발급 받기

사용자 정보 조회 API를 사용하기 위한 접근 토큰을 재발급합니다.

• Content-type: application/json

• Method: POST

• URL: /api-partner/v1/apps-in-toss/user/oauth2/refresh-token

🕍 잠시만요

refreshToken 유효시간은 14일이에요.

요청

이름	타입	필수값 여부	설명
refreshToken	string	Υ	획득한 RefreshToken

성공 응답

이름	타입	필수값 여부	설명
tokenType	string	Υ	bearer 로 고정
accessToken	string	Υ	accessToken
refreshToken	string	Υ	refreshToken
expiresIn	string	Υ	만료시간(초)
scope	string	Υ	인가된 scope(구분)

실패 응답

이름	타입	필수값 여부	설명
errorCode	string	Υ	에러 코드
reason	string	Υ	에러 메시지

4. 사용자 정보 받기

사용자 정보를 조회합니다.

DI는 null 로 내려가며 횟수 제한없이 호출이 가능해요.

개인정보 보호를 위해 암호화된 형태로 제공합니다.

• Content-type: application/json

• Method: GET

• URL: /api-partner/v1/apps-in-toss/user/oauth2/login-me

요청 헤더

이름	타입	필수값 여부	설명
Authorization	string	Υ	AccessToken으로 인증 요청 Authorization: Bearer \${AccessToken}

성공 응답

이름	타입	필수값 여부	암호화 여부	설명
userKey	number	Υ	N	사용자 식별자
scope	string	Υ	N	인가된 scope(구분)
agreedTerms	list	Υ	Ν	사용자가 동의한 약관 목록
policy	string	Υ	N	OAuth 로그인 연동 동선 구분 = Login 으로 값 고정 필요
ci	string	N	Υ	사용자 CI
name	string	N	Υ	사용자 이름
phone	string	N	Υ	사용자 휴대전화번호
gender	string	N	Υ	사용자 성별(MALE/FEMALE)
nationality	string	N	Υ	사용자 내/외국인 여부 (LOCAL/FOREIGNER)
birthday	string	N	Υ	사용자 생년월일(yyyyMMdd)
email	string	N	Υ	사용자 이메일(점유인증 하지 않은 이메일 정 보)

json

```
{
    "resultType":"SUCCESS",
    "success":{
        "userKey":443731104,
        "scope": "user_ci, user_birthday, user_nationality, user_name, user_phone, use
        "agreedTerms":["terms_tag1","terms_tag2"],
        "policy": "AUTO SELECT",
        "certTxId": "ad052b57-dc8f-4cdb-a6e2-7b494e28b5ec",
        "name": "ENCRYPTED VALUE",
        "phone": "ENCRYPTED_VALUE",
        "birthday":"ENCRYPTED_VALUE",
        "ci": "ENCRYPTED VALUE",
        "di":null,
        "gender":"ENCRYPTED_VALUE",
        "nationality": "ENCRYPTED VALUE",
        "email":null
    }
```

실패 응답

유효하지 않은 토큰을 사용할 경우, 현재 사용 중인 access_token의 유효시간을 확인하고 재발급을 진행 해주세요.

```
json
{
    "error":"invalid_grant"
}
```

서버 에러 응답 예시

errorCode	설명
INTERNAL_ERROR	내부 서버 에러
USER_KEY_NOT_FOUND	로그인 서비스에 접속한 유저 키 값 을 찾을 수 없음
USER_NOT_FOUND	토스 유저 정보를 찾을 수 없음
BAD_REQUEST_RETRIEVE_CERT_RESULT_EXCEEDED_LIMIT	조회 가능 횟수 초과 동일한 토큰으로 /api/login/user/me/without-

errorCode	설명
	di API 조회하면 정상적으로 조회 되나, di 필드는 null 값으로 내려감

```
json {
    "resultType":"FAIL",
    "error":{
        "errorCode":"INTERNAL_ERROR",
        "reason":"요청을 처리하는 도중에 문제가 발생했습니다."
    }
}
```

5. 사용자 정보 복호화하기

콘솔을 통해 이메일로 받은 복호화 키 와 AAD(Additional Authenticated DATA) 로 진행해주세요.

암호화 알고리즘

- AES 대칭키 암호화
- 키길이: 256비트
- 모드:GCM
- AAD : 복호화 키와 함께 이메일로 전달드립니다.

데이터 교환방식

- 암호화된 데이터의 앞 부분에 IV/NONCE 값을 첨부하여 드립니다.
- 복호화할 때는 암호화된 데이터의 앞 부분에서 IV/NONCE 값을 추출하여 사용하시길 바랍니다.

복호화 샘플 코드

- ▶ Kotlin 예제
- ▶ PHP 예제

▶ JAVA 예제

6. 로그인 끊기

발급받은 AccessToken을 더 이상 사용하지 않거나 사용자의 요구에 의해 만료시킬 경우 토큰을 삭제(만료)해주세요.

• Content-type: application/json

Method: Post

• URI:

- accessToken 으로 연결 끊기: /api-partner/v1/apps-in-toss/user/oauth2/access/remove-by-access-token
- userKey 로 연결 끊기: /api-partner/v1/apps-in-toss/user/oauth2/access/removeby-user-key

AccessToken 으로 로그인 연결 끊기

```
// 포맷

curl --request POST 'https://apps-in-toss-api.toss.im/api-partner/v1/apps-in-tos
--header 'Content-Type: application/json' \
--header 'Authorization: Bearer $access_token'

// 예시

curl --request POST 'https://apps-in-toss-api.toss.im/api-partner/v1/apps-in-tos
--header 'Content-Type: application/json' \
--header 'Authorization: Bearer eyJraWQiOiJjZXJ0IizzYWxnIjoiUlMyNTYifQ.eyJzdWIiO
```

userKey 로 로그인 연결 끊기



userKey에 맵핑된 AccessToken이 많을 경우 readTimeout 이 발생할 수 있어요 readTimeout 은 3초 입니다

```
// 포맷

curl --request POST 'https://apps-in-toss-api.toss.im/api-partner/v1/apps-in-tos
--header 'Content-Type: application/json' \
--header 'Authorization: Bearer $access_token' \
--data '{"userKey": $user_key}'

// 예시

curl --request POST 'https://apps-in-toss-api.toss.im/api-partner/v1/apps-in-tos
--header 'Content-Type: application/json' \
--data '{"userKey": 443731103}'
```

```
json
{
    "resultType": "SUCCESS",
    "success": {
        "userKey": 443731103
    }
}
```

7. 콜백을 통해 로그인 끊기

사용자가 토스앱 내에서 서비스와의 연결을 해제한 경우 가맹점 서버로 알려드려요.

서비스에서 연결이 끊긴 사용자에 대한 처리가 필요한 경우 활용할 수 있어요. 콜백을 받을 URL과 basic Auth 헤더는 콘솔에서 입력할 수 있어요.

서비스에서 직접 로그인 연결 끊기 요청을 호출한 경우 콜백이 호출되지 않아요.

GET 방식

• 요청 requestParam에 userKey 와 referrer 을 포함합니다.

```
// 포맷

curl --request GET '$callback_url?userKey=$userKey&referrer=$referrer'

// 예시

curl --request GET '$callback_url?userKey=443731103&referrer=UNLINK'
```

POST 방식

• 요청 body에 userKey 와 referrer 을 포함합니다.

```
// 포맷

curl --request POST '$callback_url' \
--header 'Content-Type: application/json' \
--data '{"userKey": $user_key, "referrer": $referrer}'

// 예시

curl --request POST '$callback_url' \
--header 'Content-Type: application/json' \
--data '{"userKey": 443731103, "referrer": "UNLINK"}'
```

referrer 은 연결 끊기 요청 경로에요.

referrer	설명
UNLINK	사용자가 앱에서 연결 끊기
WITHDRAWAL_TERMS	로그인 서비스 약관 철회
WITHDRAWAL_TOSS	토스 회원 탈퇴

트러블슈팅

로컬 개발 중 인증 에러가 발생할 때

로컬에서 개발할 때 인증 에러가 발생하는 원인은 주로 두가지예요.

- 1. 인증 토큰이 만료됨 기존에 발급받은 인증 토큰이 만료되었을 수 있어요. 새로운 토큰을 발급받아 다시 시도해보세요.
- 2. 개발자 로그인이 되지 않음 샌드박스 환경에서 개발자 계정으로 로그인하지 않은 상태일 수 있어요. <u>샌</u> <u>드박스 앱 다운로드</u>를 참고해 로그인을 진행한 뒤 다시 시도해보세요.

Previous page 간편 로그인 연동하기 Next page 파트너사 로그인 연동하기