

# Kakao Login API

1. 사용자가 카카오계정으로 로그인 버튼 클릭
2. 카카오톡 앱에 연결된 카카오계정의 자격정보(Credentials)를 통해 사용자 인식
3. 자격정보가 올바른 경우 사용자(Resource Owner)로부터 접근 자원에 대한 동의/허가 획득
4. 위 3까지 성공적으로 수행되면, 인증 코드(Authorization Code) 발급

해당 인증 코드는 Redirection URI를 기반으로 Third 앱에 전달.

5. Third 앱에서는 전달받은 인증 코드를 기반으로 사용자 토큰(Access Token, Refresh Token) 요청

카카오 계정 설정

# OPEN | BETA

더 나은 세상을 꿈꾸고 그것을 현실로 만드는 이를 위하여 카카오에서 앱 개발 플랫폼 서비스를 시작합니다.  
이젠 개발자들이 서비스 개발에만 전념할 수 있도록 카카오가 도와드리겠습니다.

Service

Platform



카카오계정 로그인



카카오링크



나에게 보내기



카카오내비



카카오페이



카카오스토리 포스팅



푸시 알림



앱로그 분석



검색



음성



지도/로컬



비전

앱 개발 시작하기

 약관동의

 정보입력

 가입인증

 가입완료

모두 확인, 동의합니다.



- 전체동의는 필수 및 선택정보에 대한 동의도 포함되어 있으며, 개별적으로도 동의를 선택하실 수 있습니다.
- 선택항목에 대한 동의를 거부하시는 경우에도 서비스는 이용이 가능합니다.

카카오 서비스 약관 (필수)

보기



개인정보 수집 및 이용 동의 (필수)

보기



프로필 정보 추가 수집 동의 (선택)

보기



이벤트 및 마케팅 활용 동의 (선택)

보기



다음

# 내 애플리케이션



## 내 애플리케이션

앱 만들기

### 앱 만들기

아이콘

APP

Upload

데스크탑에서는 아이콘 이미지 파일을 끌어서 넣어도 됩니다.

이름

앱 만들기

내 계정

내 애플리케이션

로그아웃

## 내애플리케이션

테스트앱 ▼

개요

통계

API

실시간

활동

앱로그 분석

설정

일반

사용자 관리 U

음성

메시지 템플릿 v2

푸시 알림

팀 관리

고급 U

## 앱 만들기

### 테스트앱

애플리케이션을 생성하였습니다

네이티브 앱 키

[Redacted]

REST API 키

[Redacted]

JavaScript 키

[Redacted]

Admin 키

[Redacted]


카카오 플랫폼 서비스와 함께 멋진 앱을 만들어 보세요.

개발가이드

설정 SDK 다운로드

\* 카카오계정으로 로그인을 사용하려면 사용자 관리를 설정해야 합니다.

# 개요 - 설정

 내애플리케이션

테스트앱

개요

통계

API

실시간

활동

앱로그 분석

설정

일반

사용자 관리

음성

메시지 템플릿 v2

푸시 알림

팀 관리

고급

APP 테스트앱

개요

카카오 계정 로그인 [사용자 관리](#)

사용 불가 - 카카오 계정으로 로그인을 사용하기 위해서는 사용자 관리를 설정해야 합니다.

앱 정보 [설정](#)

앱 키 [? 앱 키 표시](#)

설정된 플랫폼 [?](#)

통계 [자세히보기](#)

API 요청수 [?](#) 0    쿼터 사용량 [?](#) 0

API

카카오링크 [?](#)    앱로그 분석 [?](#)



# 사용자 관리 활성화

## 사용자 관리

개발가이드 : [iOS](#) [Android](#) [JavaScript](#) [REST API](#)

ON

☐

카카오계정으로 로그인 및 API를 사용하려면 사용자 관리를 활성화하세요.

### 동의항목

로그인을 통해 앱(서비스) 사용자에게 보여주는 동의항목을 설정합니다.  
연결 시 최소한의 동의를 받고, 특정 기능 사용 시 추가로 동의를 받을 수 있습니다.  
연결 시 동의항목은 필수항목과 선택항목으로 나뉩니다.  
필수항목은 사용자가 반드시 동의해야만 가입을 완료할 수 있습니다.

개인정보 보호항목

접근 권한 관리항목

항목 이름

Id

필수

연결 시 선택

이용중 동의

사용안함

프로필 정보(닉네임/프로필 사진)

profile

☒☐☐☐

관련 API 앱 연결, 사용자 정보 요청, 카카오톡 프로필 요청, 카카오토리 프로필 요청

수집목적

# 플랫폼 추가

## 내애플리케이션

테스트앱

개요

통계

API

실시간

활동

사용자

앱로그 분석

설정

일반

사용자 관리

음성

메시지 템플릿 v2

푸시 알림

팀 관리

고급

## APP 테스트앱

### 기본 정보

#### 앱 키

네이티브 앱 키 ?

REST API 키 ?

JavaScript 키 ?

Admin 키 ?

#### 플랫폼

#### 앱 삭제

× 앱을 영구적으로 삭제합니다

+ 플랫폼 추가

# iOS - Bundle ID

플랫폼 추가

☐ Android ☒ iOS ☐ 웹

번들 ID ?

kr.giftbot.example

플랫폼

🍏 iOS

✓ 번들 ID ? kr.giftbot.example

iPhone 앱 스토어 ID ?

iPhone 마켓 URL ?

iPad 앱 스토어 ID ?

iPad 마켓 URL ?

✓ 필수 입력

✓ 저장되었습니다.

# 프로젝트 설정

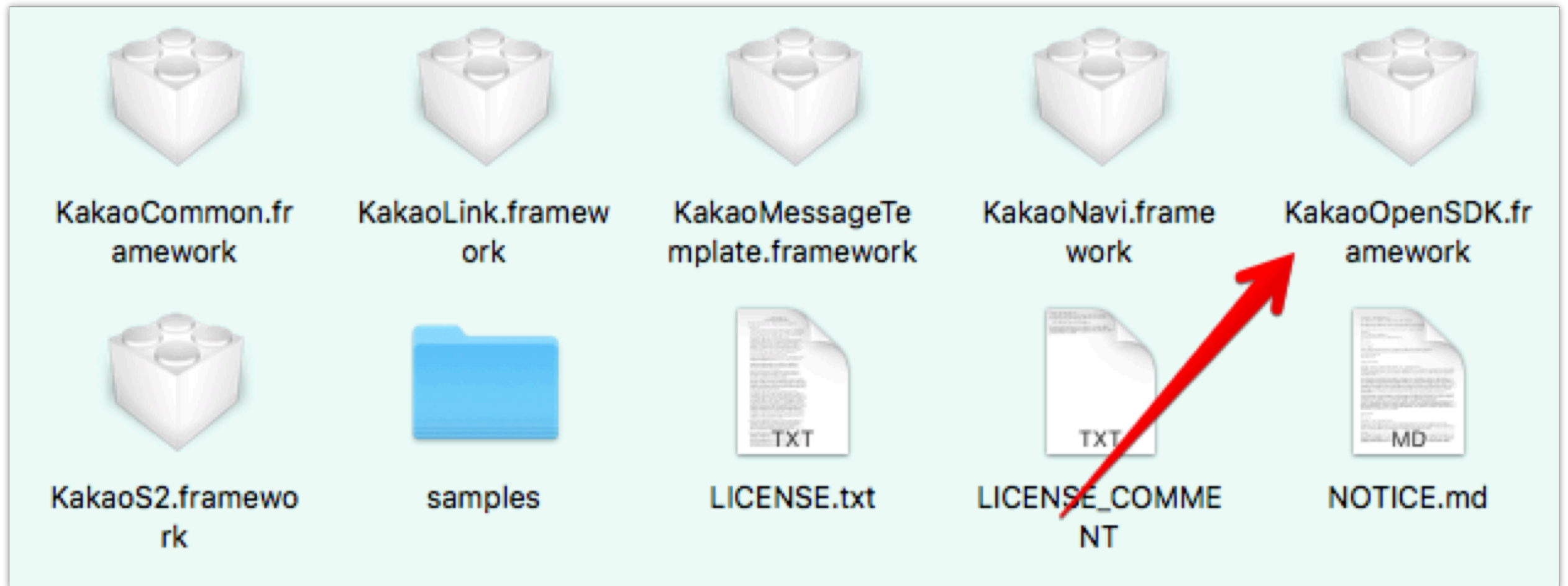
## [ iOS 개발 가이드 ]

- <https://developers.kakao.com/docs/ios/getting-started>

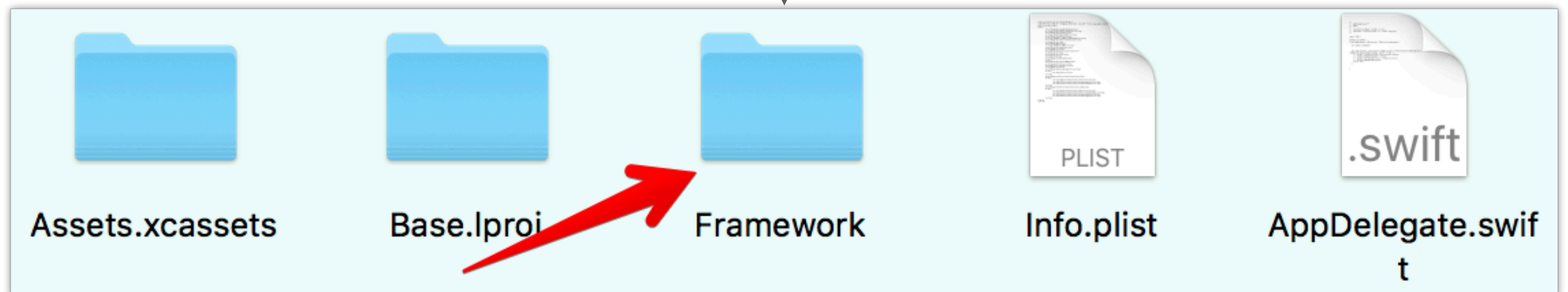
## [ 최신 SDK 다운로드 ]

- <https://developers.kakao.com/sdk/latest-ios-sdk>

# Copy Framework



Project 경로에 가서 Framework polder에 붙여넣기



# Link Binary With Libraries

General

Capabilities

Resource Tags

Info

Build Settings

Build Phases

Build Rules

+

Filter


▶ Target Dependencies (0 items)

▶ Compile Sources (2 items)

×

▼ Link Binary With Libraries (1 item)

×

Name	Status
 KakaoOpenSDK.framework	Required ↕
+ 누르고 Add Others로 경로의 Framework 추가	
<div><div>+</div><div>—</div></div>	Drag to reorder frameworks

▶ Copy Bundle Resources (3 items)

×

- all\_load - **Loads** all members of **static** archive libraries.
- ObjC - **Loads** all members of **static** archive libraries that implement an **Objective-C** class or category.

GeneralCapabilitiesResource TagsInfoBuild SettingsBuild PhasesBuild Rules

BasicCustomizedAllCombinedLevels+

Q other linker

▼ Linking

SettingOAuth

► Other Linker Flags

Quote Linker Arguments

Yes

-all\_load



# URL Schemes

## kakao + Native App Key

### Info

#### ▼ URL Types (1)

Untitled



Identifier

Icon

URL Schemes

Role

▶ Additional url type properties (0)



▼ URL types	⌵	Array	(1 item)
▼ Item 0 (Editor)		Dictionary	(2 items)
Document Role	⌵	String	Editor
▼ URL Schemes	⌵	Array	(1 item)
Item 0		String	kakaocac9ecd83e01243b8b4c12a229a1b0ba

# KAKAO\_APP\_KEY

Key	Type	Value
▼ Information Property List	Dictionary	(16 items)
KAKAO_APP_KEY ⌵ ⬆ ⬇	String	⌵ cac9ecd83e01243b8b4c12a229a1b0ba
Localization native development re... ⌵	String	\$(DEVELOPMENT_LANGUAGE) ⌵
Executable file ⌵	String	\$(EXECUTABLE_NAME)
Bundle identifier ⌵	String	\$(PRODUCT_BUNDLE_IDENTIFIER)
InfoDictionary version ⌵	String	6.0
Bundle name ⌵	String	\$(PRODUCT_NAME)
Bundle OS Type code ⌵	String	APPL
Bundle versions string, short ⌵	String	1.0
▶ URL types ⌵	Array	(1 item)
Bundle version ⌵	String	1
Application requires iPhone enviro... ⌵	Boolean	YES ⌵
Launch screen interface file base... ⌵	String	LaunchScreen
Main storyboard file base name ⌵	String	Main
▶ Required device capabilities ⌵	Array	(1 item)
▶ Supported interface orientations ⌵	Array	(3 items)
▶ Supported interface orientations (i... ⌵	Array	(4 items)

# LSApplicationQueriesSchemes

사용하고자 하는 URL scheme들을 등록

```
<key>LSApplicationQueriesSchemes</key>
<array>
  <string>kakao + Native App Key</string>
  <string>kakaokompassauth</string>
  <string>storykompassauth</string>
  <string>kakaolink</string>
  <string>storylink</string>
</array>
```

Key	Type	Value
▼ Information Property List	Dictionary	(17 items)
▼ LSApplicationQueriesSchemes ⌵ ╕ Ⓜ	Array	(5 items)
Item 0	String	kakaocac9ecd83e01243b8b4c12a229a1b0ba
Item 1	String	kakaokompassauth
Item 2	String	storykompassauth
Item 3	String	kakaolink
Item 4	String	storylink

# ATS Settings

http에 접근 가능하도록 설정



```
<key>NSAppTransportSecurity</key>
<dict>
  <key>NSAllowsArbitraryLoads</key>
  <true/>
</dict>
```

Key	Type	Value
▼ Information Property List	Dictionary	(18 items)
▼ App Transport Security Settin... ⌵ + -	Dictionary ⌵	(1 item)
Allow Arbitrary Loads ⌵	Boolean	YES ⌵

# Create Objective-C File

Choose a template for your new file:

iOS

watchOS

tvOS

macOS

Filter

## Source



Cocoa Touch  
Class



UI Test  
Case Class



Unit Test  
Case Class



Swift File



Objective-C File



Header File



C File



C++ File



Metal File

## User Interface



Storyboard



View



Empty



Launch Screen

# Create Bridging Header

Choose op



## Would you like to configure an Objective-C bridging header?

Adding this file to OAuth will create a mixed Swift and Objective-C target. Would you like Xcode to automatically configure a bridging header to enable classes to be accessed by both languages?

Cancel

Don't Create

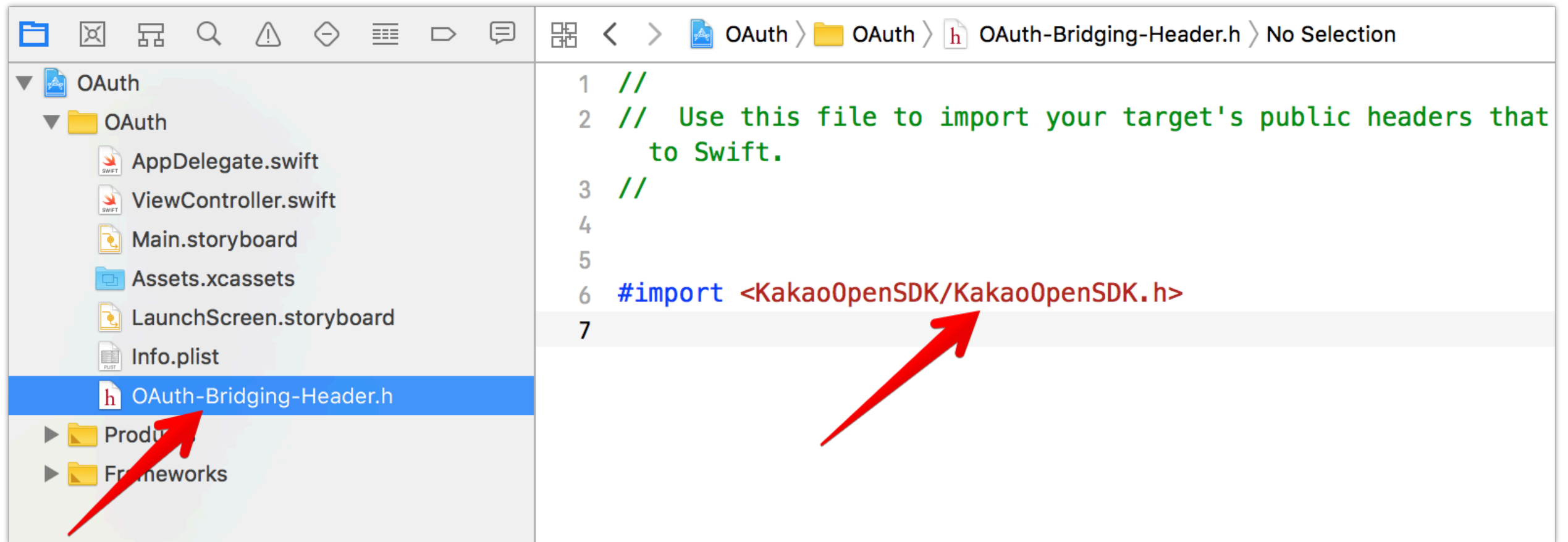
Create Bridging Header

File: temp

File Type: Empty File

Class: NSObject

# import KakaoOpenSDK



# Objective-C Bridging Header

GeneralCapabilitiesResource TagsInfoBuild SettingsBuild PhasesBuild Rules

BasicCustomizedAllCombinedLevels+

Q bridg

▼ Apple LLVM 9.0 - Warnings - Objective C and ARC

SettingOAuth

Using \_\_bridge Casts Outside of ARCYes

▼ Swift Compiler - General

SettingOAuth

Install Objective-C Compatibility HeaderYes

► Objective-C Bridging HeaderOAuth/OAuth-Bridging-Header.h

Precompile Bridging HeaderYes



카카오 개발자 웹사이트에 등록한 번들 ID와 실행하려는 앱에 설정된 번들 ID가 다를 경우에는 다음과 같은 에러가 출력되면서 인증이 되지 않습니다.

**invalid android\_key\_hash or ios\_bundle\_id**

- 로그인 시도시 '+[NSBundle ko\_appKey]: unrecognized selector sent to class 0x111111'의 에러가 발생하면서 앱이 크래쉬 되는 문제

현재 작업중인 프로젝트에서 Other Linker Flags에 "-all\_load"가 추가되었는지 확인해 봅니다. 자세한 내용은 [Kakao SDK import](#)를 참고합니다.

- Other Linker Flags에 "-all\_load"를 추가하였을 경우, 타 라이브러리와 충돌로 빌드가 실패하는 경우

"-all\_load"대신에 "-force\_load \$(SRCROOT)/KakaoOpenSDK.framework/KakaoOpenSDK"를 사용하여 Kakao SDK를 강제 로드합니다. <!--

- dyld: Symbol not found: **OBJC\_CLASS\$\_NSJSONSerialization**

Kakao SDK는 iOS 5.1.1 버전 이상이 필요합니다. 해당 os의 버전 확인이 필요합니다. -->

- 웹뷰 등 취소 후 다시 로그인 시도시 "The operation is cancelled."가 반복적으로 나오는 현상

openWithCompletionHandler 호출 전에 `[[KOSession sharedSession] close]`를 해주는 로직이 빠지지 않았는지 확인합니다. 로그인 예제를 통해 구현을 다시 한번 확인합니다.

로그인 구현

# AppDelegate

---

```
func applicationDidEnterBackground(_ application: UIApplication) {  
    K0Session.handleDidEnterBackground()  
}
```

```
func applicationDidBecomeActive(_ application: UIApplication) {  
    K0Session.handleDidBecomeActive()  
}
```

```
func application(_ app: UIApplication, open url: URL, options:  
[UIApplicationOpenURLOptionsKey : Any] = [:]) -> Bool {  
    if K0Session.isKakaoAccountLoginCallback(url) {  
        return K0Session.handleOpen(url)  
    }  
    return false  
}
```

```
@IBAction private func kakaoLogin(_ sender: UIButton) {  
    guard let session = K0Session.shared() else { return }  
  
    // Close Old Session  
    session.isOpen() ? session.close() : ()  
  
    session.open { (error) in  
        guard session.isOpen() else {  
            // Open Session Failed  
            return  
        }  
        // Login Success  
    }  
}
```

# KOSessionDidChange

---

```
func sessionDidChangeNotification() {  
    NotificationCenter.default.addObserver(  
        forName: Notification.Name.KOSessionDidChange,  
        object: nil,  
        queue: .main  
    ) { noti in  
        guard let session = noti.object as? KOSession else { return }  
        // Code...  
    }  
}
```