

---

# 네이버 카페 SDK 적용 가이드

---

---

# 저작권

---

Copyright © NAVER Corp. All Rights Reserved.

이 문서는 네이버(주)의 지적 재산이므로 어떠한 경우에도 네이버(주)의 공식적인 허가 없이 이 문서의 일부 또는 전체를 복제, 전송, 배포하거나 변경하여 사용할 수 없습니다.

이 문서는 정보 제공의 목적으로만 제공됩니다. 네이버(주)는 이 문서에 수록된 정보의 완전성과 정확성을 검증하기 위해 노력하였으나, 발생할 수 있는 내용상의 오류나 누락에 대해서는 책임지지 않습니다. 따라서 이 문서의 사용이나 사용 결과에 따른 책임은 전적으로 사용자에게 있으며, 네이버(주)는 이에 대해 명시적 혹은 묵시적으로 어떠한 보증도 하지 않습니다.

관련 URL 정보를 포함하여 이 문서에서 언급한 특정 소프트웨어 상품이나 제품은 해당 소유자가 속한 현지 및 국내외 관련법을 따르며, 해당 법률을 준수하지 않음으로 인해 발생하는 모든 결과에 대한 책임은 전적으로 사용자 자신에게 있습니다.

네이버(주)는 이 문서의 내용을 예고 없이 변경할 수 있습니다.

---

# 문서 정보

---

## 문서 개요

이 문서는 네이버 카페 SDK를 애플리케이션에 적용하는 방법을 설명한다.

## 독자

이 문서의 독자는 네이버 카페 SDK를 애플리케이션에 적용하려는 개발자다.

## 문의처

이 문서의 내용에 오류가 있거나 내용과 관련한 의문 사항이 있으면 아래의 연락처로 문의한다.

연락처: [dl\\_gamesdkpartner@navercorp.com](mailto:dl_gamesdkpartner@navercorp.com)

## 문서 버전 및 이력

---

일자	이력
2015. 12. 23.	최종 배포

---

---

# 표기 규칙

---

## 참고 표기

---

### 참고

독자가 참고해야 할 내용을 기술한다.

---

## 주의 표기

---

### 주의

독자가 반드시 알아야 할 사항, 시스템 에러를 유발할 수 있는 사항을 기술한다.

---

## UI 문구, 사용자 입력값 표기

- UI 문구: **메뉴 > 하위메뉴**
- 사용자 고정 입력값: **localhost**를 입력한다.
- 사용자 가변 입력값: `http://www.naver.com/{회사 이름}`

## 소스 코드 표기

이 문서에서 소스 코드는 회색 바탕에 검정색 글씨로 표기한다.

```
COPYDATASTRUCT st;  
st.dwData = PURPLE_OUTBOUND_ENDING;  
st.cbData = sizeof(pp);  
st.lpData = &pp;  
::SendMessage(GetTargetHwnd(), WM_COPYDATA, (LPARAM) this->m_hWnd, (LPARAM) &st);
```

---

# 목차

---

<b>개요</b>	<b>9</b>
네이버 카페 SDK 개요	9
네이버 카페 SDK 기능 및 특징	9
주요 기능	9
사용 환경	9
<b>Android 애플리케이션 적용 가이드</b>	<b>11</b>
<b>개발 환경</b>	<b>11</b>
소프트웨어 요구 사항	11
라이브러리 구성	12
개발 환경 설정	12
<b>네이버 카페 SDK 적용</b>	<b>13</b>
AndroidManifest.xml 설정	13
네이버 카페 SDK 초기화	13
네이버 카페 SDK 화면 시작	13
네이버 카페 SDK 화면 닫기	14
게시글 쓰기	14
App Scheme 처리	14
사용자 게임 아이디 매핑	14
리소스 이미지 변경	15
<b>API 레퍼런스</b>	<b>15</b>
Glink	15
init()	15
isShowGlink()	16
popBackStack()	17
setGameUserId()	17
startEvent()	18
startImageWrite()	18
startHome()	19
startMenu()	19
startNotice()	20

---

startProfile()	20
startVideoWrite()	21
startWrite()	21
stop()	22

## **iOS 애플리케이션 적용 가이드 23**

<b>개발 환경</b>	<b>23</b>
소프트웨어 요구 사항	23
라이브러리 구성	23
개발 환경 설정	24
<b>네이버 카페 SDK 적용</b>	<b>27</b>
네이버 카페 SDK 초기화	27
네이버 아이디로 로그인하기 설정	27
네이버 카페 SDK 화면 시작	27
게시글 쓰기	28
<b>API 레퍼런스</b>	<b>28</b>
NCSDKManager	28
@property (nonatomic, weak) id parentViewController	29
(void)dismissViewController	29
(void)dismissTopViewController	29
(NCSDKManager *)getSharedInstance	30
(id)navercafeRootViewController	30
(void)resetSharedInstance	31
(void)presentArticlePostViewControllerWithMenuId	31
(void)presentArticlePostViewControllerWithType	32
(void)presentMainViewController	33
(void)presentMainViewControllerWithArticleId	33
(void)presentMainViewControllerWithTabIndex	33
(void)presentViewController	34
(void)setGameUserId	34
(void)setNaverLoginClientId	35
NCSDKLoginManager	35
@property (nonatomic, weak) UIViewController *rootViewController	36
(NSSString *)accessToken	36
(NSSString *)accessTokenExpireTime	37
(BOOL)finishNaverLoginWithURL	37
(NCSDKLoginManager *)getSharedInstance	38
(void)isLoginWithFinish	38
(BOOL)isValidAccessTokenExpireTimeNow	38
(void)loginWithFinish	39
(void)logout	39
(void)refreshAccessToken	40

(void)refreshAccessTokenWithFinish	40
(void)requestDeleteToken	41
(void)setIsInAppOAuthEnable	41
(void)setIsNaverAppOAuthEnable	42
<b>Unity 애플리케이션 적용 가이드</b>	<b>43</b>
<b>개발 환경</b>	<b>43</b>
소프트웨어 요구 사항	43
라이브러리 구성	43
개발 환경 설정	43
<b>네이버 카페 SDK 적용</b>	<b>47</b>
실행	47
사용자의 게임 아이디와 네이버 아이디 연동	47
AFNetworking 교체	48
<b>API 레퍼런스</b>	<b>48</b>
GLinkUnity	48
executeArticlePost()	49
executeArticlePostWithImage()	49
executeArticlePostWithVideo()	50
executeMain()	50

---

# 표 및 그림 목록

---

## 표 목록

표 1 Android용 네이버 카페 SDK 사용에 필요한 라이브러리 _____	11
표 2 iOS용 네이버 카페 SDK 사용에 필요한 라이브러리 _____	23

## 그림 목록

그림 1 네이버 카페 SDK _____	9
-----------------------	---

# 개요

## 네이버 카페 SDK 개요

네이버 카페 SDK는 게임 엔진에 관계없이 모바일 게임에 손쉽게 네이버 카페를 통합할 수 있는 라이브러리다. 네이버 카페 SDK를 적용하면 게임 사용자는 게임을 벗어나지 않고 게임 커뮤니티(네이버 카페)와 소통할 수 있다.



그림 1 네이버 카페 SDK

## 네이버 카페 SDK 기능 및 특징

### 주요 기능

네이버 카페 SDK의 주요 기능은 다음과 같다

- iOS와 Android 지원
- Unity 4 이상과 Cocos2d-x 2.1 이상 게임 엔진 지원
- 카페 기능을 사용할 수 있는 화면 제공
- OAuth 2.0 기반의 네이버 아이디로 로그인을 사용한 사용자 인증

### 사용 환경

#### 운영체제

네이버 카페 SDK는 다음과 같은 환경에서 작동한다.

- Android: Android 4.2(Jelly Bean) API Level 17 이상
- iOS: iOS 7.0 이상(ARMv7, ARMv7s, ARM64 지원)

### 네이버 아이디로 로그인 등록

네이버 카페 SDK에서 네이버 아이디로 로그인 기능으로 사용자를 인증하려면 클라이언트 아이디와 클라이언트 시크릿이 있어야 한다.

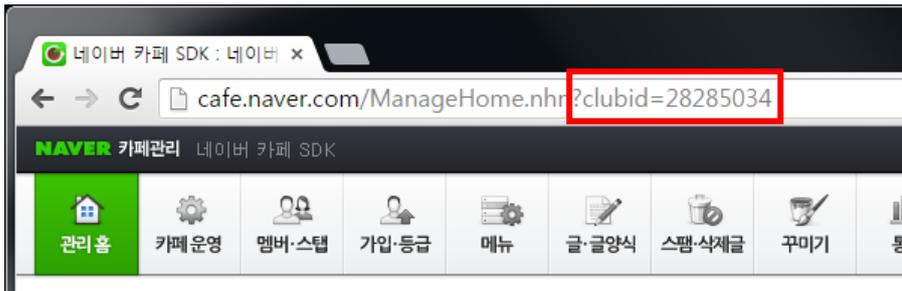
- 클라이언트 아이디: 네이버 아이디로 로그인에 게임을 등록하고 받은 클라이언트 아이디
- 클라이언트 시크릿: 네이버 아이디로 로그인에 게임을 등록하고 받은 클라이언트 시크릿

클라이언트 아이디와 클라이언트 시크릿은 네이버 아이디로 로그인에 애플리케이션을 등록하면 발급받을 수 있다. 네이버 아이디로 로그인에 애플리케이션을 등록하고 클라이언트 아이디와 클라이언트 시크릿을 확인하는 방법은 네이버 아이디로 로그인 개발자 센터의 문서를 참고한다.

- 네이버 아이디로 로그인 개발자 센터: <https://nid.naver.com/devcenter/main.nhn>

### 네이버 카페 ID

네이버 카페 SDK를 사용하려면 적용할 네이버 카페의 카페 ID가 있어야 한다. 카페 ID는 네이버 카페 관리 페이지의 URL 가운데 **clubid** 파라미터의 값이다.



# Android 애플리케이션 적용 가이드

## 개발 환경

### 소프트웨어 요구 사항

#### 개발 도구

- IDE: Android Studio 또는 Eclipse
- Android Support Library v7

#### 라이브러리

네이버 카페 SDK를 사용하려면 다음 라이브러리를 프로젝트에 추가해 함께 빌드한다.

**표 1 Android용 네이버 카페 SDK 사용에 필요한 라이브러리**

라이브러리	다운로드 URL
네이버 아이디로 로그인	<ul style="list-style-type: none"> <li>• 네이버 카페 SDK 라이브러리 파일에 포함되어 있다(4.1.4 버전).</li> <li>• 다운로드 URL: <a href="https://static.nid.naver.com/images/web/devcenter/3rdparty_login_library_android_4.1.4.zip">https://static.nid.naver.com/images/web/devcenter/3rdparty_login_library_android_4.1.4.zip</a></li> </ul>
NAVER Volleyer	<ul style="list-style-type: none"> <li>• 네이버 카페 SDK 라이브러리 파일에 포함되어 있다(2.0.1 버전).</li> <li>• 다운로드 URL: <a href="http://mvnrepository.com/artifact/com.navercorp.volleyextensions/volleyer">http://mvnrepository.com/artifact/com.navercorp.volleyextensions/volleyer</a></li> </ul>
Volley	<ul style="list-style-type: none"> <li>• 네이버 카페 SDK 라이브러리 파일에 포함되어 있다(1.0.2 버전).</li> <li>• 다운로드 URL: <a href="http://mvnrepository.com/artifact/com.mcxiaoke.volley/library">http://mvnrepository.com/artifact/com.mcxiaoke.volley/library</a></li> </ul>
Google Gson	<ul style="list-style-type: none"> <li>• 네이버 카페 SDK 라이브러리 파일에 포함되어 있다(2.3.1 버전).</li> <li>• 다운로드 URL: <a href="http://mvnrepository.com/artifact/com.google.code.gson/gson">http://mvnrepository.com/artifact/com.google.code.gson/gson</a></li> </ul>
Glide	<ul style="list-style-type: none"> <li>• 네이버 카페 SDK 라이브러리 파일에 포함되어 있다(3.6.1 버전).</li> <li>• 다운로드 URL: <a href="http://mvnrepository.com/artifact/com.github.bumptech.glide/glide">http://mvnrepository.com/artifact/com.github.bumptech.glide/glide</a></li> </ul>
Otto	<ul style="list-style-type: none"> <li>• 네이버 카페 SDK 라이브러리 파일에 포함되어 있다(1.3.8 버전).</li> <li>• 다운로드 URL: <a href="http://mvnrepository.com/artifact/com.squareup/otto">http://mvnrepository.com/artifact/com.squareup/otto</a></li> </ul>

## 라이브러리 구성

Android용 네이버 카페 SDK 라이브러리는 다음과 같이 이루어져 있다.

- lib: 네이버 카페 SDK 라이브러리와 필수 라이브러리
  - cafeSdk-x.x.x.aar: Android Studio에서 사용하는 네이버 카페 SDK 라이브러리 파일
  - cafeSdk-x.x.x.zip: Eclipse에서 사용하는 네이버 카페 SDK 라이브러리
  - library: 네이버 카페 SDK 사용에 필요한 라이브러리 폴더
- sample: 네이버 카페 SDK 라이브러리를 사용하는 예제 프로젝트 폴더
  - navercafesdk-sample-android-studio: Android Studio에서 사용할 수 있는 네이버 카페 SDK 예제 프로젝트 폴더
  - navercafesdk-sample-eclipse-master: Eclipse에서 사용할 수 있는 네이버 카페 SDK 예제 프로젝트 폴더

## 개발 환경 설정

네이버 카페 SDK를 사용하려면 IDE에 따라 다음과 같이 개발 환경을 설정한다.

### Android Studio

1. Android 프로젝트의 **libs** 폴더에 네이버 카페 SDK 라이브러리 파일(cafeSdk-x.x.x.aar)을 복사한다.
2. Android 프로젝트의 **libs** 폴더에 네이버 아이디로 로그인 라이브러리 파일(naveroauthlogin-4.x.x.jar)을 복사한다.
3. 메뉴에서 **File > Project Structure**를 클릭하고 **Project Structure** 대화 상자에서 **App > Dependencies**를 클릭한 다음 라이브러리를 추가한다. 또는 다음과 같이 Android 프로젝트의 **build.gradle** 파일에 직접 라이브러리를 추가한다.

```
compile 'com.android.support:support-v13:23.1.0'
compile 'com.navercorp.volleyextensions:volleyer:2.0.1'
compile 'com.google.code.gson:gson:2.3.1'
compile 'com.github.bumptech.glide:glide:3.6.1'
compile 'com.squareup:otto:1.3.8'
```

### Eclipse

1. 네이버 카페 SDK 라이브러리 파일(cafeSdk-x.x.x.zip)의 압축을 해제한다.
2. 압축을 해제한 네이버 카페 SDK 라이브러리 폴더 아래의 **libs** 폴더에 빌드에 필요한 필수 라이브러리 파일을 추가한다.
3. Android 프로젝트의 **AndroidManifest.xml** 파일에 다음과 같이 네이버 아이디로 로그인에서 사용하는 액티비티와 네이버 카페 SDK에서 사용하는 액티비티를 추가한다.

```
<activity
  android:name="com.naver.glink.android.sdk.ui.VideoPlayActivity"
  android:screenOrientation="sensorLandscape"
  android:theme="@android:style/Theme.Translucent.NoTitleBar.Fullscreen" />
<activity
  android:name="com.nhn.android.naverlogin.ui.OAuthLoginActivity"
  android:screenOrientation="sensorLandscape"
  android:theme="@android:style/Theme.Translucent.NoTitleBar" />
<activity
  android:name="com.nhn.android.naverlogin.ui.OAuthLoginInAppBrowserActivity"
```

```
android:label="@Auth2.0 In-app"  
android:screenOrientation="sensorLandscape" />
```

## 네이버 카페 SDK 적용

### AndroidManifest.xml 설정

Android 프로젝트의 **AndroidManifest.xml** 파일에 다음과 같은 내용을 추가한다.

package 속성에는 네이버 아이디로 로그인에 애플리케이션을 등록할 때 **안드로이드 Intent**에 설정한 내용을 입력한다. 네이버 카페 SDK를 제어하는 객체인 Glink 클래스의 접근 권한도 함께 설정한다.

```
<!-- package는 네이버 아이디로 로그인 개발자 센터에 등록된 안드로이드 Intent와 동일해야 한다. -->  
<manifest xmlns:android="http://schemas.android.com/apk/res/android"  
    package="com.naver.glink.sample">  
<!-- Glink에 필요한 접근 권한 -->  
    <uses-permission android:name="android.permission.INTERNET"/>  
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/>  
    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
```

### 네이버 카페 SDK 초기화

네이버 카페 SDK를 초기화하는 기능은 다음과 같이 Glink.init() 메서드로 구현한다.

클라이언트 아이디와 클라이언트 시크릿은 네이버 아이디로 로그인에 애플리케이션을 등록하고 받은 값이다.

```
/**  
 * "네야로 개발자 센터"에서 받은 정보로 SDK를 초기화 합니다.  
 * Glink의 다른 메소드를 호출하기 전에 반드시 초기화를 먼저해야 합니다.  
 * 개발자 센터 주소: https://nid.naver.com/devcenter/main.nhn  
 */  
final int cafeId = 28266581;  
final String consumerKey = "클라이언트 아이디";  
final String consumerSecret = "클라이언트 시크릿";  
Glink.init(consumerKey, consumerSecret, cafeId);
```

### 네이버 카페 SDK 화면 시작

네이버 카페 SDK 화면은 5개의 탭으로 구성된다. 네이버 카페 SDK가 시작할 때 어떤 탭이 선택된 상태로 시작할지 선택한다.

예를 들어 홈 탭이 선택된 상태로 시작하려면 다음과 같이 Glink.startHome() 메서드를 사용한다.

```
// 홈 화면으로 시작한다.  
Glink.startHome(activity);
```

그 외에 공지 사항 탭, 이벤트 탭, 게시판 탭, 프로필 탭으로 시작하게 하려면 다음 메서드를 사용한다.

- Glink.startNotice() 메서드: 공지 사항 탭으로 시작한다.
- Glink.startEvent() 메서드: 이벤트 탭으로 시작한다.
- Glink.startMenu() 메서드: 게시판 탭으로 시작한다.
- Glink.startProfile() 메서드: 프로필 탭으로 시작한다.

## 네이버 카페 SDK 화면 닫기

네이버 카페 SDK 화면을 닫는 기능은 `Glink.stop()` 메서드나 `Glink.popBackStack()` 메서드로 구현한다.

- `Glink.stop()` 메서드: 화면을 바로 닫고 네이버 카페 SDK를 종료한다.
- `Glink.popBackStack()` 메서드: `backStack`으로 쌓여 있는 화면을 하나씩 닫는다. 모든 화면을 닫으면 네이버 카페 SDK를 종료한다. `backStack`으로 쌓일 수 있는 화면은 게시글 보기 화면, 글쓰기 화면, 검색 화면이다.

다음은 `Glink.stop()` 메서드로 네이버 카페 SDK 화면을 닫는 예다.

```
// stop() 메서드로 종료하기.  
Glink.stop(activity);
```

다음은 `Glink.popBackStack()` 메서드로 네이버 카페 SDK 화면을 닫는 예다.

```
// popBackStack으로 종료하기.  
Glink.popBackStack(activity);
```

## 게시글 쓰기

게임 사용자가 카페에 게시글을 쓰는 화면은 다음과 같이 `Glink.startWrite()` 메서드로 구현한다.

```
// 기본 제목, 본문을 넣어서 글쓰기 화면을 시작한다.  
int menuId = 4; // 0이면 메뉴를 선택하지 않는다.  
String text = "기본 제목, 본문을 넣어서 글쓰기 화면을 시작합니다. ";  
Glink.startWrite(MainActivity.this, "subject", text);
```

`Glink.startImageWrite()` 메서드나 `Glink.startVideoWrite()` 메서드를 사용하면 이미지나 동영상이 첨부된 상태로 게시판 글쓰기 화면을 실행할 수 있다.

## App Scheme 처리

App Scheme으로 이동하는 배너 이미지가 있을 때 다음과 같이 `OnClickAppSchemeBannerListener`를 설정하면 터치 이벤트가 일어났을 때 App Scheme을 처리하는 기능을 구현할 수 있다.

```
// 앱스킴 터치 리스너 설정.  
Glink.setOnClickAppSchemeBannerListener(new Glink.OnClickAppSchemeBannerListener() {  
    @Override public void onClickAppSchemeBanner(String appScheme) {  
        // 카페 관리에서 설정한 App Scheme 문자열을 네이버 카페 SDK에서 넘겨준다.  
        // App Scheme을 처리하는 코드를 구현한다.  
    }  
});
```

## 사용자 게임 아이디 매핑

사용자의 게임 아이디와 네이버 아이디를 매핑하는 기능은 다음과 같이 `Glink.setGameUserId()` 메서드로 구현한다.

```
// 사용자의 게임 아이디와 네이버 아이디를 매핑한다.  
// 프로필 화면에서 매핑된 게임 아이디를 확인할 수 있다.  
Glink.setGameUserId(this, "gameUserId", "게임 ID");
```

## 리소스 이미지 변경

네이버 카페 SDK 라이브러리에 포함된 리소스 이미지는 다음과 같이 변경할 수 있다.

1. 네이버 카페 SDK 라이브러리 파일(.aaa 파일)의 압축을 해제한다.
2. 압축을 해제한 폴더의 **/res/drawable-xhdpi** 폴더에 있는 이미지를 원하는 이미지로 바꾼다.
3. 압축을 해제한 폴더를 다시 네이버 카페 SDK 라이브러리 파일(.aaa 파일)로 압축한다.
4. 새로 압축한 네이버 카페 SDK 라이브러리를 적용해 프로젝트를 빌드한다.

---

### 주의

이미지를 변경할 때 이미지의 크기는 기존 이미지와 동일해야 한다.

---

## API 레퍼런스

### Glink

네이버 카페 SDK를 제어하는 클래스. Glink 클래스의 메서드를 사용해 네이버 카페 SDK를 초기화하거나 시작, 중지하는 기능을 구현한다.

Glink 클래스의 메서드는 다음과 같다.

- `init()`
- `isShowGlink()`
- `popBackStack()`
- `setGameUserId()`
- `startEvent()`
- `startImageWrite()`
- `startHome()`
- `startMenu()`
- `startNotice()`
- `startProfile()`
- `startVideoWrite()`
- `startWrite()`
- `stop()`

### init()

#### 설명

네이버 카페 SDK를 초기화한다.

**구문**

```
public static void init(String clientId, String clientSecret, int cafeId);
```

**파라미터**

파라미터	타입	필수 여부	설명
clientId	String	Y	클라이언트 아이디. 네이버 아이디로 로그인에 애플리케이션을 등록하고 받은 값이다.
clientSecret	String	Y	클라이언트 시크릿. 네이버 아이디로 로그인에 애플리케이션을 등록하고 받은 값이다.
cafeId	int	Y	카페 ID. 네이버 카페 관리 페이지의 URL 가운데 <b>clubid</b> 파라미터의 값이다.

**반환값**

없음

**코드 예**

```
// 네이버 아이디로 로그인 정보와 카페 ID로 네이버 카페 SDK를 초기화한다.
Glink.init("abcd", "aaaa", 33);
```

**isShowGlink()**

**설명**

네이버 카페 SDK 화면이 열려 있는지 확인한다.

**구문**

```
public static boolean isShowGlink(Activity activity);
```

**파라미터**

파라미터	타입	필수 여부	설명
activity	Activity	Y	메서드를 실행하는 액티비티의 Context 객체

**반환값**

- true: 네이버 카페 SDK 화면이 열려 있는 상태
- false: 네이버 카페 SDK 화면이 열려 있지 않은 상태

**코드 예**

```
// 네이버 카페 SDK 화면이 열려 있는지 확인한다.
// 반환값이 true이면 열려 있는 상태고, 반환값이 false이면 열려 있지 않은 상태다.
Glink.isShowGlink(this);
```

## popBackStack()

### 설명

backStack으로 쌓여 있는 화면을 하나씩 닫는다. 모든 화면을 닫으면 네이버 카페 SDK를 종료한다.  
backStack으로 쌓일 수 있는 화면은 게시글 보기 화면, 글쓰기 화면, 검색 화면이다.

### 구문

```
public static void popBackStack(Activity activity);
```

### 파라미터

파라미터	타입	필수 여부	설명
activity	Activity	Y	메서드를 실행하는 액티비티의 Context 객체

### 반환값

없음

### 코드 예

```
// 네이버 카페 SDK 화면을 하나씩 닫는다.  
Glink.popBackStack (activity);
```

## setGameUserId()

### 설명

사용자의 게임 아이디와 카페 아이디를 매핑한다.

### 구문

```
public static void setGameUserId(Activity activity, String gameId, String fieldName);
```

### 파라미터

파라미터	타입	필수 여부	설명
activity	Activity	Y	메서드를 실행하는 액티비티의 Context 객체
gameId	String	N	사용자의 게임 아이디
fieldName	String	N	프로필 화면에 표시할 아이디(기본 값: 사용자의 게임 아이디)

### 반환값

없음

### 코드 예

```
// 사용자의 게임 아이디와 네이버 아이디를 매핑한다.
```

```
// 프로필 화면에서 매핑된 게임 아이디를 확인할 수 있다.
Glink.setGameUserId(this, "gameUserId", "게임 ID");
```

## startEvent()

### 설명

이벤트 탭이 선택된 상태로 네이버 카페 SDK를 시작한다.

### 구문

```
public static void startEvent(Activity activity);
```

### 파라미터

파라미터	타입	필수 여부	설명
activity	Activity	Y	메서드를 실행하는 액티비티의 Context 객체

### 반환값

없음

### 코드 예

```
// 이벤트 탭으로 네이버 카페 SDK를 시작한다.
Glink.startEvent(activity);
```

## startImageWrite()

### 설명

이미지가 첨부된 상태로 게시판 글쓰기 화면을 연다.

### 구문

```
public static void startImageWrite(Activity activity, int menuId, String subject, String text, String imagery);
```

### 파라미터

파라미터	타입	필수 여부	설명
activity	Activity	Y	메서드를 실행하는 액티비티의 Context 객체
menuId	int	N	게시판 아이디(기본값: -1). 게시판 아이디는 네이버 카페 게시판 URL 가운데 <b>menuid</b> 파라미터의 값이다.
subject	String	N	게시글 제목
text	String	N	게시글 본문
imageUri	String	N	이미지 파일의 경로(URI 형식)

**반환값**

없음

**코드 예**

```
// 기본 제목, 본문이 있고 이미지가 첨부된 상태로 글쓰기 화면을 시작한다. 이미지 경로는 URI 형식으로 넣는다.
int menuId = 4; // 0이면 메뉴를 선택하지 않는다.
String text = "기본 제목, 본문이 있고 이미지가 첨부된 상태로 글쓰기 화면을 시작합니다.\n이미지 경로는
URI 형식으로 넣어 주시면 됩니다.";
String path = "your image uri";
Glink.startImageWrite(MainActivity.this, menuId, "subject", text, path);
```

**startHome()****설명**

홈 탭이 선택된 상태로 네이버 카페 SDK 화면을 연다.

**구문**

```
public static void startHome(Activity activity);
```

**파라미터**

파라미터	타입	필수 여부	설명
activity	Activity	Y	메서드를 실행하는 액티비티의 Context 객체

**반환값**

없음

**코드 예**

```
// 홈 탭으로 네이버 카페 SDK 화면을 연다.
Glink.startHome(activity);
```

**startMenu()****설명**

게시판 탭이 선택된 상태로 네이버 카페 SDK 화면을 연다.

**구문**

```
public static void startMenu(Activity activity);
```

**파라미터**

파라미터	타입	필수 여부	설명
activity	Activity	Y	메서드를 실행하는 액티비티의 Context 객체

**반환값**

없음

**코드 예**

```
// 게시판 탭으로 네이버 카페 SDK 화면을 연다.  
Glink.startMenu(activity);
```

**startNotice()**

**설명**

공지 사항 탭이 선택된 상태로 네이버 카페 SDK 화면을 연다.

**구문**

```
public static void startNotice(Activity activity);
```

**파라미터**

파라미터	타입	필수 여부	설명
activity	Activity	Y	메서드를 실행하는 액티비티의 Context 객체

**반환값**

없음

**코드 예**

```
// 공지 사항 탭으로 네이버 카페 SDK 화면을 연다.  
Glink.startNotice(activity);
```

**startProfile()**

**설명**

프로필 탭이 선택된 상태로 네이버 카페 SDK 화면을 연다.

**구문**

```
public static void startProfile(Activity activity);
```

**파라미터**

파라미터	타입	필수 여부	설명
activity	Activity	Y	메서드를 실행하는 액티비티의 Context 객체

**반환값**

없음

**코드 예**

```
// 프로필 탭으로 네이버 카페 SDK 화면을 연다.
Glink.startProfile(activity);
```

**startVideoWrite()**

**설명**

동영상이 첨부된 상태로 게시판 글쓰기 화면을 연다.

**구문**

```
public static void startVideoWrite(Activity activity, int menuId, String subject, String text, String videoUri);
```

**파라미터**

파라미터	타입	필수 여부	설명
activity	Activity	Y	메서드를 실행하는 액티비티의 Context 객체
menuId	int	N	게시판 아이디(기본값: -1). 게시판 아이디는 네이버 카페 게시판 URL 가운데 <b>menuid</b> 파라미터의 값이다.
subject	String	N	게시글 제목
text	String	N	게시글 본문
videoUri	String	N	동영상 파일의 경로(URI 형식)

**반환값**

없음

**코드 예**

```
// 기본 제목, 본문있고 동영상이 첨부된 상태로 글쓰기 화면을 시작한다. 동영상 파일의 경로는 URI 형태로 넣는다.
int menuId = 4; // 0이면 메뉴를 선택하지 않는다.
String text = "기본 제목, 본문이 있고 동영상이 첨부된 상태로 글쓰기 화면을 시작합니다.\n동영상 파일의 경로는 URI 형태로 넣어 주시면 됩니다.";
String path = "your video uri";
Glink.startVideoWrite(MainActivity.this, menuId, "subject", text, path);
```

**startWrite()**

**설명**

게시판 글쓰기 화면을 연다.

**구문**

```
public static void startWrite(Activity activity, int menuId, String subject, String text);
```

### 파라미터

파라미터	타입	필수 여부	설명
activity	Activity	Y	메서드를 실행하는 액티비티의 Context 객체
menuId	int	N	게시판 아이디(기본값: -1). 게시판 아이디는 네이버 카페 게시판 URL 가운데 <b>menuid</b> 파라미터의 값이다.
subject	String	N	게시글 제목
text	String	N	게시글 본문

### 반환값

없음

### 코드 예

```
// 기본 제목, 본문을 넣어서 글쓰기 화면을 시작한다.
int menuId = 4; // 0이면 메뉴를 선택하지 않는다.
String text = "기본 제목, 본문을 넣어서 글쓰기 화면을 시작합니다. ";
Glink.startWrite(MainActivity.this, "subject", text);
```

## stop()

### 설명

네이버 카페 SDK 화면을 바로 닫고 네이버 카페 SDK를 종료한다.

### 구문

```
public static void stop(final Activity activity);
```

### 파라미터

파라미터	타입	필수 여부	설명
activity	Activity	Y	메서드를 실행하는 액티비티의 Context 객체

### 반환값

없음

### 코드 예

```
// stop() 메서드로 종료하기.
Glink.stop(activity);
```

# iOS 애플리케이션 적용 가이드

## 개발 환경

### 소프트웨어 요구 사항

#### 개발 도구

- IDE: Xcode 6.0 이상

---

#### 주의

라이브러리에 ARC(automatic reference counting)가 적용됐다.

---

#### 라이브러리

네이버 카페 SDK를 사용하려면 다음 라이브러리를 프로젝트에 추가해 함께 빌드한다.

#### 표 2 iOS용 네이버 카페 SDK 사용에 필요한 라이브러리

라이브러리	다운로드 URL
네이버 아이디로 로그인	<ul style="list-style-type: none"><li>• 네이버 카페 SDK 라이브러리 파일에 포함되어 있다(4.0.6 버전).</li><li>• 다운로드 URL: <a href="https://static.nid.naver.com/images/web/devcenter/3rdparty_login_library_ios_4.0.6.zip">https://static.nid.naver.com/images/web/devcenter/3rdparty_login_library_ios_4.0.6.zip</a></li></ul>
AFNetworking 1.0 이상	<ul style="list-style-type: none"><li>• 네이버 카페 SDK 라이브러리 파일에 포함되어 있다(2.6.1 버전).</li><li>• 다운로드 URL: <a href="https://github.com/AFNetworking/AFNetworking">https://github.com/AFNetworking/AFNetworking</a></li></ul>

## 라이브러리 구성

iOS용 네이버 카페 SDK 라이브러리는 다음과 같이 이루어져 있다.

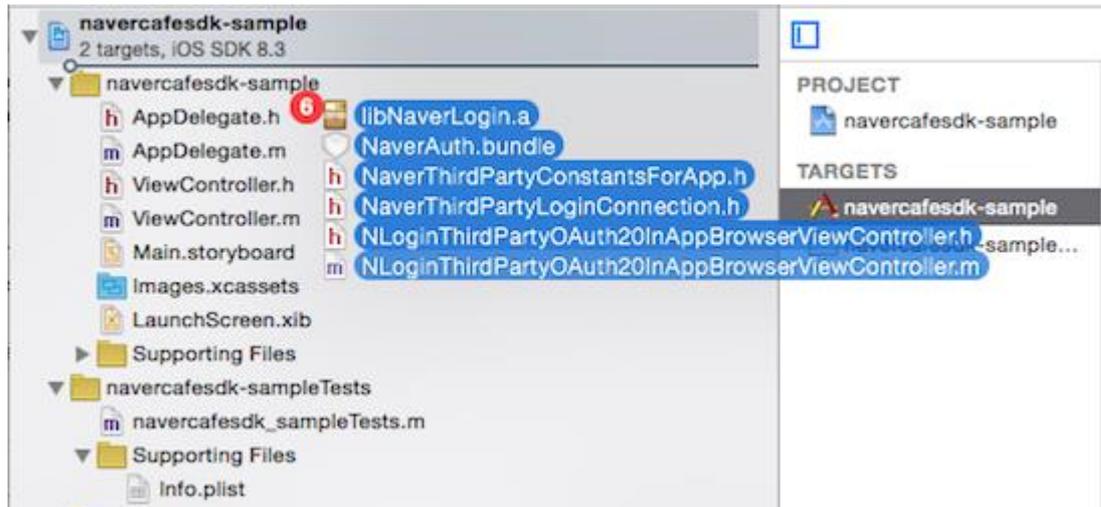
- lib: 네이버 카페 SDK 라이브러리 폴더
  - NaverCafeSDK.bundle
  - NaverCafeSDK.framework
- sample: 네이버 카페 SDK 라이브러리를 사용하는 예제 프로젝트와 필수 라이브러리 폴더
  - external-lib: 네이버 아이디로 로그인 라이브러리와 AFNetworking 라이브러리 폴더
  - navercafesdk-sample-ios: 네이버 카페 SDK 예제 프로젝트 폴더

## 개발 환경 설정

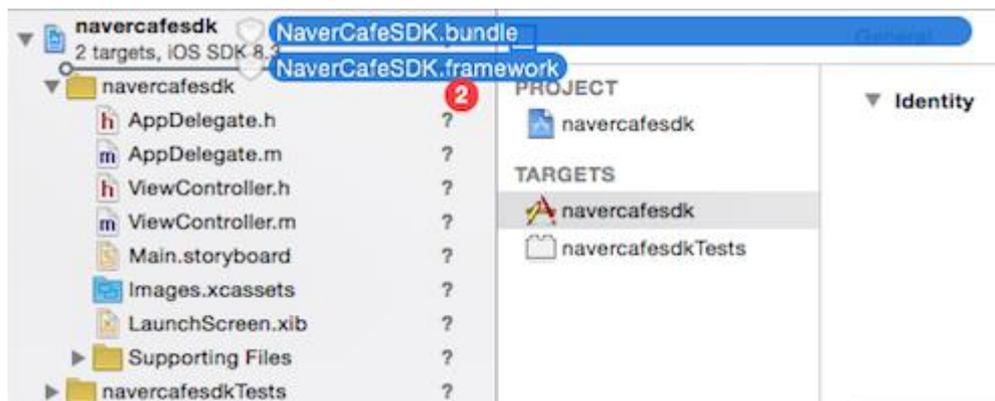
### Xcode 설정

네이버 카페 SDK를 사용하려면 Xcode에서 다음과 같이 개발 환경을 설정한다.

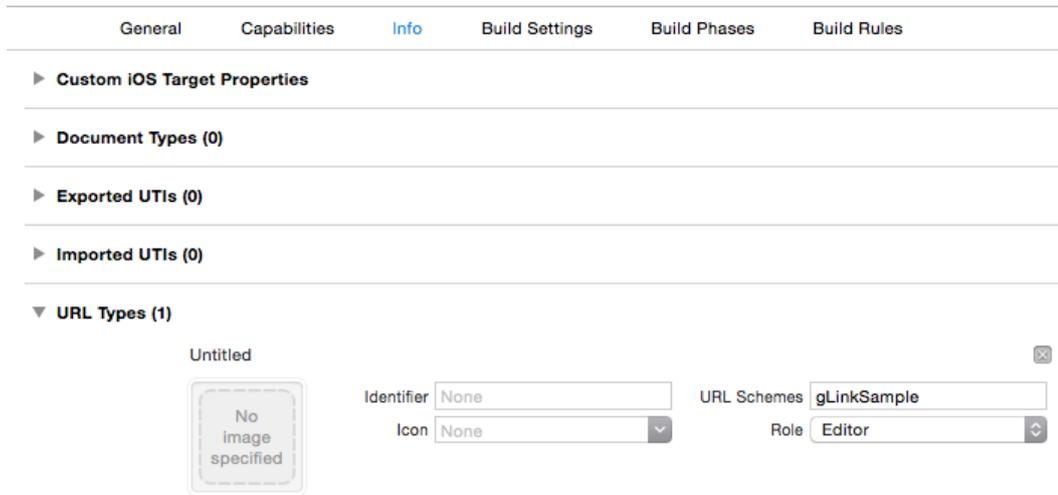
1. 네이버 아이디로 로그인 라이브러리의 압축을 해제한다.
2. 네이버 아이디로 로그인 라이브러리를 Xcode에서 프로젝트에 추가한다.



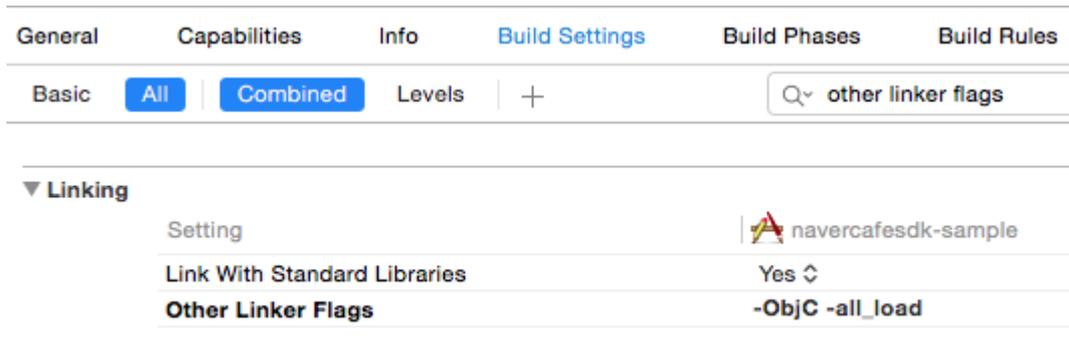
3. **AFNetworking** 라이브러리를 Xcode에서 프로젝트에 추가한다.
4. 네이버 카페 SDK 라이브러리의 압축을 해제한다.
5. **NaverCafeSDK.framework**와 **NaverCafeSDK.bundle**을 Xcode에서 프로젝트에 추가한다.



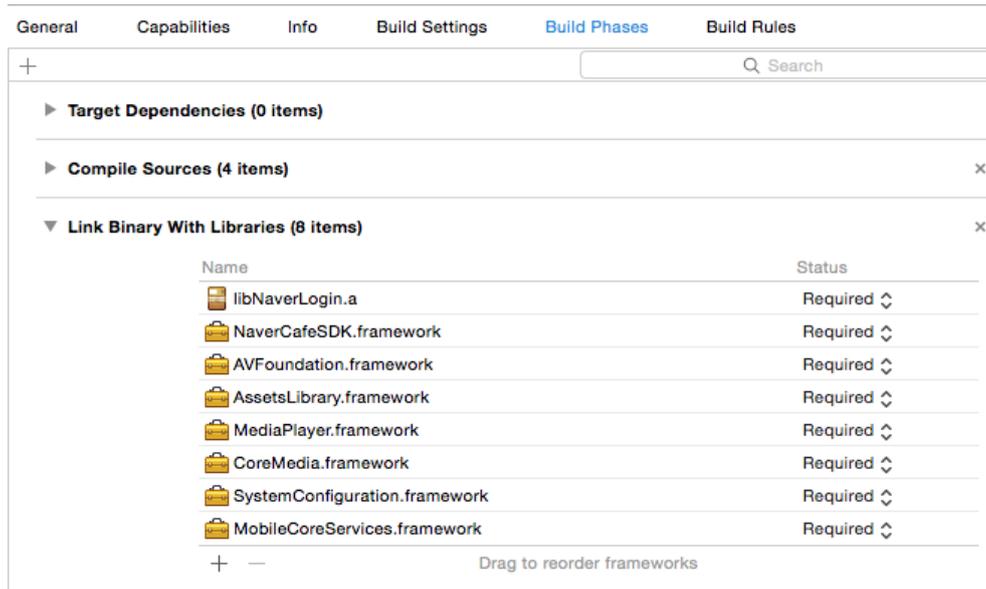
6. 네이버 아이디로 로그인에 애플리케이션을 등록할 때 입력한 URL Scheme을 프로젝트에 등록한다.



7. 정적 라이브러리를 사용할 수 있게 **Build Settings**의 **Other Linker Flags**에 **-ObjC -all\_load** 옵션을 설정한다.



8. **Build Phases**의 **Link Binary With Libraries**에 다음 라이브러리를 추가한다.
  - MobileCoreServices.framework
  - SystemConfiguration.framework
  - MediaPlayer.framework
  - AVFoundation.framework
  - CoreMedia.framework



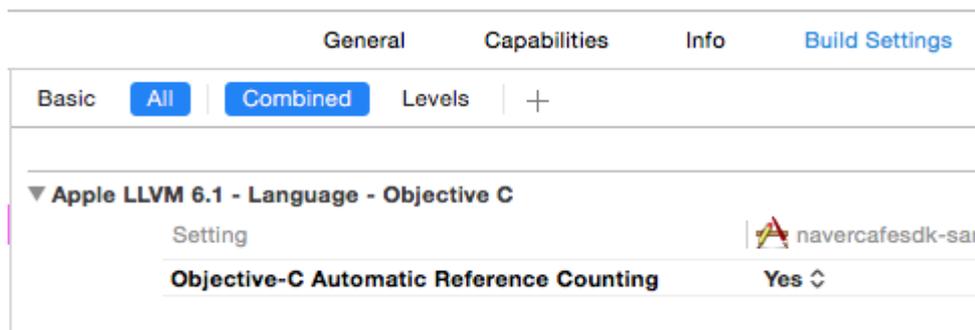
Cocos2d-x 엔진을 사용한다면 다음 라이브러리를 추가한다.

- MobileCoreServices.framework
- SystemConfiguration.framework
- MediaPlayer.framework
- AVFoundation.framework
- CoreMedia.framework
- GameController.framework
- AssetsLibrary.framework
- Security.framework

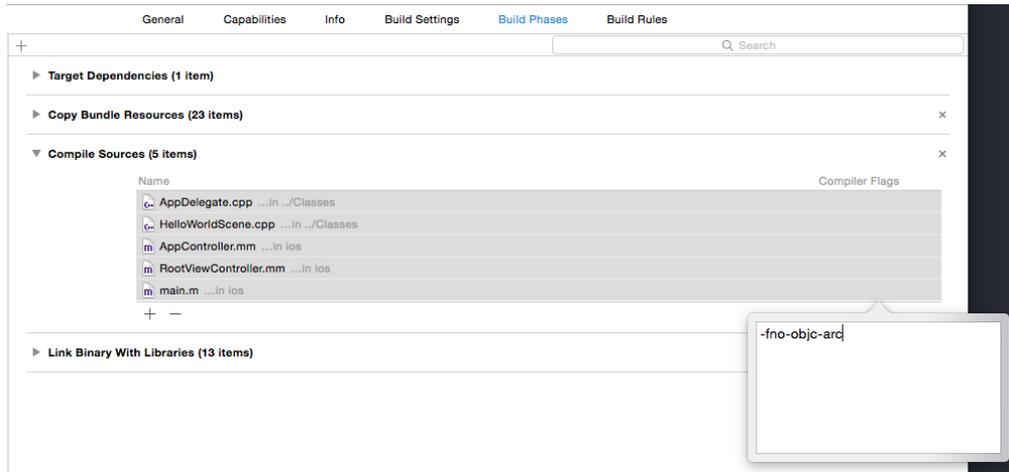
### MRC 환경을 ARC 환경으로 전환

Cocos2d-x 엔진을 사용하는 프로젝트가 MRC(manual reference counting) 환경이면 다음과 같은 방법으로 ARC(automatic reference counting) 환경으로 전환한다.

1. Build Settings의 Objective-C Automatic Reference Counting을 YES로 설정한다.



2. Build Phases의 Compile Sources에서 컴파일할 파일의 Compile Flags를 -fno-objc-arc로 설정한다.



## 네이버 카페 SDK 적용

### 네이버 카페 SDK 초기화

다음과 같은 형식으로 네이버 카페 SDK를 초기화한다. 클라이언트 아이디와 클라이언트 시크릿은 네이버 아이디로 로그인에 애플리케이션을 등록하고 받은 값이다.

```
//ViewController
#import <NaverCafeSDK/NCSDKManager.h>

- (void)viewDidLoad {
//네이버 아이디로 로그인 등록 정보, 카페 ID로 네이버 카페 SDK를 초기화한다.
[[NCSDKManager sharedInstance] setNaverLoginConsumerKey:@"클라이언트 아이디"
                                naverLoginConsumerSecret:@"클라이언트 시크릿"
                                cafeId:00000000];
}
```

### 네이버 아이디로 로그인하기 설정

다음과 같이 앱 델리게이트를 설정해 네이버 아이디로 로그인이 완료되면 네이버 카페 SDK에 로그인 정보가 설정되게 한다.

```
//AppDelegate
#import <NaverCafeSDK/NCSDKLoginManager.h>

- (BOOL)application:(UIApplication *)application openURL:(NSURL *)url
sourceApplication:(NSString *)sourceApplication annotation:(id)annotation {
//로그인 정보를 네이버 아이디로 로그인 객체에 설정한다.
return [[NCSDKLoginManager sharedInstance] finishNaverLoginWithURL:url];
}
```

### 네이버 카페 SDK 화면 시작

네이버 카페 SDK 화면을 시작하려면 네이버 카페 SDK 뷰 컨트롤러를 설정하고 presentMainViewController 메서드를 실행한다.

```
#import <NaverCafeSDK/NCSDKManager.h>
```

```
//카페 홈을 실행합니다
[[NCSDKManager sharedInstance] setParentViewController:self];
[[NCSDKManager sharedInstance] presentMainViewController];
```

presentMainViewController 메서드 대신 presentMainViewControllerWithTabIndex 메서드를 사용하면 네이버 카페 SDK 화면의 특정한 탭을 선택한 상태로 네이버 카페 SDK 화면을 시작할 수 있다. presentMainViewControllerWithArticleId 메서드를 사용하면 특정한 게시판 글이 열린 상태로 네이버 카페 SDK 화면을 시작할 수 있다.

## 게시글 쓰기

카페 게시판에 글을 쓰는 화면을 열려면 다음과 같이 presentArticlePostViewControllerWithMenuId 메서드를 사용한다.

```
//글쓰기 화면을 실행합니다.
[[NCSDKManager sharedInstance] setParentViewController:self];
[[NCSDKManager sharedInstance] presentArticlePostViewControllerWithMenuId:0
subject:@"글 제목" content:@"글 내용" filePath:@"document"];
```

## API 레퍼런스

### NCSDKManager

네이버 카페 SDK를 제어하는 클래스. NCSDKManager 클래스의 메서드를 사용해 네이버 카페 SDK를 초기화하거나 시작, 중지하는 기능을 구현한다. 네이버 카페 SDK를 실행하는 최상위 뷰 컨트롤러인 parentViewController 클래스를 가지고 있다.

NCSDKManager 클래스의 메서드는 다음과 같다.

- dismissViewController
- dismissTopViewController
- sharedInstance
- navercafeRootViewController
- resetSharedInstance
- presentArticlePostViewControllerWithMenuId
- presentArticlePostViewControllerWithType
- presentMainViewController
- presentMainViewControllerWithArticleId
- presentMainViewControllerWithTabIndex
- presentViewController
- setGameUserId
- setNaverLoginClientId

## @property (nonatomic, weak) id parentViewController

### 설명

네이버 카페 SDK를 실행하는 뷰 컨트롤러. 네이버 카페 SDK가 실행되는 부모 클래스다.

### 구문

```
@property (nonatomic, weak) id parentViewController;
```

### 파라미터

없음

### 반환값

UIViewController 객체

### 코드 예

```
[[NCSDKManager sharedInstance] setParentViewController:self]
```

## (void)dismissViewController

### 설명

네이버 카페 SDK를 실행하는 뷰 컨트롤러 위에 실행된 다른 뷰 컨트롤러를 제거한다.

### 구문

```
- (void)dismissViewController:(id)viewController;
```

### 파라미터

파라미터	타입	필수 여부	설명
viewController	id	Y	네이버 카페 SDK 뷰 컨트롤러 위에 실행된 뷰 컨트롤러

### 반환값

없음

### 코드 예

```
[[NCSDKManager sharedInstance] dismissViewController:self];
```

## (void)dismissTopViewController

### 설명

네이버 카페 SDK를 실행하는 뷰 컨트롤러 위에 실행된 다른 뷰 컨트롤러 가운데 제일 위에 있는 뷰 컨트롤러를 제거한다.

**구문**

```
- (void)dismissTopViewController;
```

**파라미터**

없음

**반환값**

없음

**코드 예**

```
[NCSDKManager sharedInstance] dismissTopViewController];
```

**(NCSDKManager \*)getSharedInstance**

**설명**

네이버 카페 SDK 인스턴스(싱글턴 인스턴스)를 얻는다.

**구문**

```
+ (NCSDKManager *)getSharedInstance;
```

**파라미터**

없음

**반환값**

NCSDKManager 객체

**코드 예**

```
[NCSDKManager sharedInstance]
```

**(id)navercafeRootViewController**

**설명**

네이버 카페 SDK의 최상위 뷰 컨트롤러 객체를 얻는다.

**구문**

```
- (id)navercafeRootViewController;
```

**파라미터**

없음

**반환값**

UIViewController 객체

**코드 예**

```
[[NCSDKManager sharedInstance] navercafeRootViewController]
```

**(void)resetSharedInstance**

**설명**

NCSDKManager 객체를 삭제한다.

**구문**

```
+ (void)resetSharedInstance;
```

**파라미터**

없음

**반환값**

없음

**코드 예**

```
[NCSDKManager resetSharedInstance]
```

**(void)presentArticlePostViewControllerWithMenuId**

**설명**

게시판 글쓰기 화면을 연다.

**구문**

```
- (void)presentArticlePostViewControllerWithMenuId:(NSInteger)menuId
        subject:(NSString *)subject
        content:(NSString *)content;
```

**파라미터**

파라미터	타입	필수 여부	설명
menuId	NSInteger	Y	게시판 아이디(기본값: 0). 게시판 아이디는 네이버 카페 게시판 URL 가운데 <b>menuId</b> 파라미터의 값이다.
subject	NSString	N	게시글 제목
content	NSString	N	게시글 본문

**반환값**

없음

**코드 예**

```
[[NCSDKManager sharedInstance] presentArticlePostViewControllerWithMenuId:1
                                subject:@"제목"
                                content:@"내용"];
```

**(void)presentArticlePostViewControllerWithType**

**설명**

파일이 첨부된 상태로 게시판 글쓰기 화면을 연다.

**구문**

```
- (void)presentArticlePostViewControllerWithType:(GLArticlePostType) type
        menuId:(NSInteger)menuId
        subject:(NSString *)subject
        content:(NSString *)content
        filePath:(NSString *)filePath;
```

**파라미터**

파라미터	타입	필수 여부	설명
type	GLArticlePostType Y		첨부 파일의 유형 <ul style="list-style-type: none"> <li>1: 이미지 파일</li> <li>2: 동영상 파일</li> </ul>
menuId	NSInteger	Y	게시판 아이디(기본값: 0). 게시판 아이디는 네이버 카페 게시판 URL 가운데 <b>menuid</b> 파라미터의 값이다.
subject	NSString	N	게시글 제목
content	NSString	N	게시글 본문
filePath	NSString	Y	첨부파일 경로

**반환값**

없음

**코드 예**

```
[[NCSDKManager sharedInstance] presentArticlePostViewControllerWithType:kGLArticlePostTypeVideo
                                menuId:1
                                subject:@"제목"
                                content:@"내용"
                                filePath:@"private/var/mobile/Applications/0D1657F9-EACF-4D64-BC8A-4E01EB4FF247/tmp/trim.2CC623C7-78C3-4597-BA75-9BA12BFEF333.MOV"];
```

### **(void)presentMainViewController**

**설명**

네이버 카페 SDK 화면을 연다.

**구문**

```
- (void)presentMainViewController;
```

**파라미터**

없음

**반환값**

없음

**코드 예**

```
[[NCSDKManager sharedInstance] presentMainViewController];
```

### **(void)presentMainViewControllerWithArticleId**

**설명**

게시판 글이 열린 상태로 네이버 카페 SDK 화면을 연다.

**구문**

```
- (void)presentMainViewControllerWithArticleId:(NSInteger)articleId;
```

**파라미터**

파라미터	타입	필수 여부	설명
articleId	NSInteger	Y	게시글 아이디(기본값: 0)

**반환값**

없음

**코드 예**

```
[[NCSDKManager sharedInstance] presentMainViewControllerWithArticleId:10];
```

### **(void)presentMainViewControllerWithTabIndex**

**설명**

지정한 탭이 선택된 상태로 네이버 카페 SDK 화면을 연다.

**구문**

```
- (void)presentMainViewControllerWithTabIndex:(NSInteger)tabIndex;
```

---

### 파라미터

파라미터	타입	필수 여부	설명
tabIndex	NSUInteger	Y	선택할 탭의 인덱스값(기본값: 0) <ul style="list-style-type: none"> <li>• 0: 홈 탭</li> <li>• 1: 공지 사항 탭</li> <li>• 2: 이벤트 탭</li> <li>• 3: 통합 게시판 탭</li> <li>• 4: 프로필 탭</li> </ul>

### 반환값

없음

### 코드 예

```
[[NCSDKManager sharedInstance] presentMainViewControllerWithTabIndex:1];
```

## (void)presentViewController

### 설명

네이버 카페 SDK의 최상위 뷰 컨트롤러에서 다른 뷰 컨트롤러를 실행한다.

### 구문

```
- (void)presentViewController:(id)viewController;
```

### 파라미터

파라미터	타입	필수 여부	설명
viewController	id	Y	뷰 컨트롤러 객체의 ID

### 반환값

없음

### 코드 예

```
[[NCSDKManager sharedInstance] presentViewController:self];
```

## (void)setGameUserId

### 설명

사용자의 게임 아이디와 네이버 아이디를 연동하고 프로필 화면에 아이디를 표시한다.

### 구문

```
- (void)setGameUserId:(NSString *)gameUserId fieldName:(NSString *)fieldName;
```

**파라미터**

파라미터	타입	필수 여부	설명
gameUserId	NSString	Y	사용자의 게임 아이디
fieldName	NSString	N	프로필 화면에 표시할 아이디(기본값: 게임 아이디)

**반환값**

없음

**코드 예**

```
[[NCSDKManager sharedInstance] setGameUserId:@"abc3251235" fieldName:@"게임ID"];
```

**(void)setNaverLoginClientId**

**설명**

네이버 아이디로 로그인 객체를 설정한다.

**구문**

```
- (void)setNaverLoginClientId:(NSString *)naverLoginClientId
      naverLoginClientSecret:(NSString *)naverLoginClientSecret
      cafeId:(NSInteger)cafeId;
```

**파라미터**

없음

**반환값**

없음

**코드 예**

```
[[NCSDKManager sharedInstance] setNaverLoginConsumerKey:@"Consumer ID"
      naverLoginConsumerSecret:@"Secret ID"
      cafeId:00000000];
```

**NCSDKLoginManager**

네이버 카페 SDK에서 네이버 아이디로 로그인 기능을 제어하는 클래스.

NCSDKLoginManager 클래스의 메서드는 다음과 같다.

- accessToken
- accessTokenExpireTime
- finishNaverLoginWithURL
- sharedInstance

- isLoginWithFinish
- isValidAccessTokenExpireTimeNow
- loginWithFinish
- logout
- refreshAccessToken
- refreshAccessTokenWithFinish
- requestDeleteToken
- setIsInAppOAuthEnable
- setIsNaverAppOAuthEnable

### **@property (nonatomic, weak) UIViewController \*rootViewController**

#### **설명**

로그인 웹뷰를 실행하는 뷰 컨트롤러

#### **구문**

```
@property (nonatomic, weak) UIViewController *rootViewController;
```

#### **파라미터**

없음

#### **반환값**

UIViewController 객체

#### **코드 예**

```
[[NCSDKLoginManager sharedInstance] setRootViewController:self];
```

### **(NSString \*)accessToken**

#### **설명**

네이버 로그인 인증 후 네이버 서버에서 받은 접근 토큰(access token)을 반환받는다.

#### **구문**

```
- (NSString *)accessToken;
```

#### **파라미터**

없음

#### **반환값**

접근 토큰

**코드 예**

```
[[NCSDKLoginManager sharedInstance] accessToken];
```

**(NSString \*)accessTokenExpireTime**

**설명**

접근 토큰(access token)의 만료 시간을 확인한다.

**구문**

```
- (NSString *)accessTokenExpireTime;
```

**파라미터**

없음

**반환값**

접근 토큰 만료 시간

**코드 예**

```
[[NCSDKLoginManager sharedInstance] accessTokenExpireTime];
```

**(BOOL)finishNaverLoginWithURL**

**설명**

네이버 아이디로 로그인 완료 후 앱 델리게이트를 실행한다.

**구문**

```
- (BOOL)finishNaverLoginWithURL:(NSURL *)url;
```

**파라미터**

파라미터	타입	필수 여부	설명
url	NSURL	Y	네이버 아이디로 로그인 완료 후 앱 델리게이트로 콜백 되는 URL Scheme

**반환값**

- true: 로그인 성공
- false: 로그인 실패

**코드 예**

```
- (BOOL)application:(UIApplication *)application openURL:(NSURL *)url
sourceApplication:(NSString *)sourceApplication annotation:(id)annotation {
    return [[NCSDKLoginManager sharedInstance] finishNaverLoginWithURL:url];
}
```

## (NCSDKLoginManager \*)getSharedInstance

### 설명

네이버 아이디로 로그인 인스턴스(싱글턴 인스턴스)를 얻는다.

### 구문

```
+ (NCSDKLoginManager *)getSharedInstance;
```

### 파라미터

없음

### 반환값

NCSDKLoginManager 객체

### 코드 예

```
[NCSDKLoginManager getSharedInstance]
```

## (void)isLoginWithFinish

### 설명

네이버 로그인 인증 후 네이버 서버에서 받은 접근 토큰(access token)이 유효한지 확인한다.

### 구문

```
- (void)isLoginWithFinish:(void (^)(BOOL successAccessToken))finish;
```

### 파라미터

없음

### 반환값

- true: 유효한 접근 토큰이다.
- false: 접근 토큰이 만료되어 유효하지 않은 접근 토큰이다.

### 코드 예

```
[[NCSDKLoginManager getSharedInstance] isLoginWithFinish:^(BOOL successAccessToken) {  
    if (successAccessToken) {  
    }  
}];
```

## (BOOL)isValidAccessTokenExpireTimeNow

### 설명

접근 토큰(access token)이 있는지, 유효 기간이 남아 있는지 확인한다. 단, 서버에서 유효 기간이 만료된 경우에는 확인할 수 없다.

#### 구문

```
- (BOOL)isValidAccessTokenExpireTimeNow;
```

#### 파라미터

없음

#### 반환값

- true: 유효한 접근 토큰이다.
- false: 접근 토큰이 만료되어 유효하지 않은 접근 토큰이다.

#### 코드 예

```
[[NCSDKLoginManager sharedInstance] isValidAccessTokenExpireTimeNow];
```

### (void)loginWithFinish

#### 설명

네이버 아이디로 로그인하기 인증 과정을 시작하고 접근 토큰(access token)을 반환받는다.

#### 구문

```
- (void)loginWithFinish:(void (^)(BOOL successAccessToken))finish;
```

#### 파라미터

없음

#### 반환값

- true: 유효한 접근 토큰이다.
- false: 접근 토큰이 만료되어 유효하지 않은 접근 토큰이다.

#### 코드 예

```
[[NCSDKLoginManager sharedInstance] isLoginWithFinish:^(BOOL successAccessToken) {  
    if (successAccessToken) {  
    }  
}];
```

### (void)logout

#### 설명

클라이언트에 저장된 접근 토큰(access token)과 갱신 토큰(refresh token)을 삭제하고 로그아웃한다.

#### 구문

```
- (void)logout;//delete local accesstoken
```

**파라미터**

없음

**반환값**

없음

**코드 예**

```
[[NCSDKLoginManager sharedInstance] logout];
```

**(void)refreshAccessToken**

**설명**

네이버 로그인 인증 후 네이버 서버에서 받은 갱신 토큰(refresh token)으로 접근 토큰(access token)을 갱신한다.

**구문**

```
- (void)refreshAccessToken;
```

**파라미터**

없음

**반환값**

없음

**코드 예**

```
[[NCSDKLoginManager sharedInstance]refreshAccessToken];
```

**(void)refreshAccessTokenWithFinish**

**설명**

네이버 로그인 인증 후 네이버 서버에서 갱신 토큰(refresh token)을 반환받는다.

**구문**

```
- (void)refreshAccessTokenWithFinish:(void (^)(BOOL successAccessToken)) finish;
```

**파라미터**

없음

**반환값**

- true: 접근 토큰 갱신 성공
- false: 접근 토큰 갱신 실패

**코드 예**

```
[[NCSDKLoginManager sharedInstance] refreshAccessTokenWithFinish:^(BOOL successAccessToken)
{
    if (successAccessToken) {
    }
}];
```

**(void)requestDeleteToken**

**설명**

클라이언트와 서버에 저장된 접근 토큰(access token)과 갱신 토큰(referesh token)을 삭제한다.

**구문**

```
- (void)requestDeleteToken;//delete server authorization
```

**파라미터**

없음

**반환값**

없음

**코드 예**

```
[[NCSDKLoginManager sharedInstance] requestDeleteToken];
```

**(void)setIsInAppOauthEnable**

**설명**

인앱 브라우저를 실행해 로그인 절차를 진행한다.

**구문**

```
- (void)setIsInAppOauthEnable:(BOOL)enable;
```

**파라미터**

파라미터	타입	필수 여부	설명
enable	BOOL	Y	인앱 브라우저 실행 여부

**반환값**

없음

**코드 예**

```
[[NCSDKLoginManager sharedInstance] setIsInAppOauthEnable:YES];
```

## (void)setIsNaverAppOauthEnable

### 설명

네이버 앱을 실행해 로그인 절차를 진행한다.

### 구문

```
- (void)setIsNaverAppOauthEnable:(BOOL)enable;
```

### 파라미터

파라미터	타입	필수 여부	설명
enable	BOOL	Y	네이버 앱 실행 여부

### 반환값

없음

### 코드 예

```
[[NCSDKLoginManager sharedInstance] setIsNaverAppOauthEnable:YES];
```

# Unity 애플리케이션 적용 가이드

## 개발 환경

### 소프트웨어 요구 사항

#### 개발 도구

- 게임 엔진: Unity 4 이상

### 라이브러리 구성

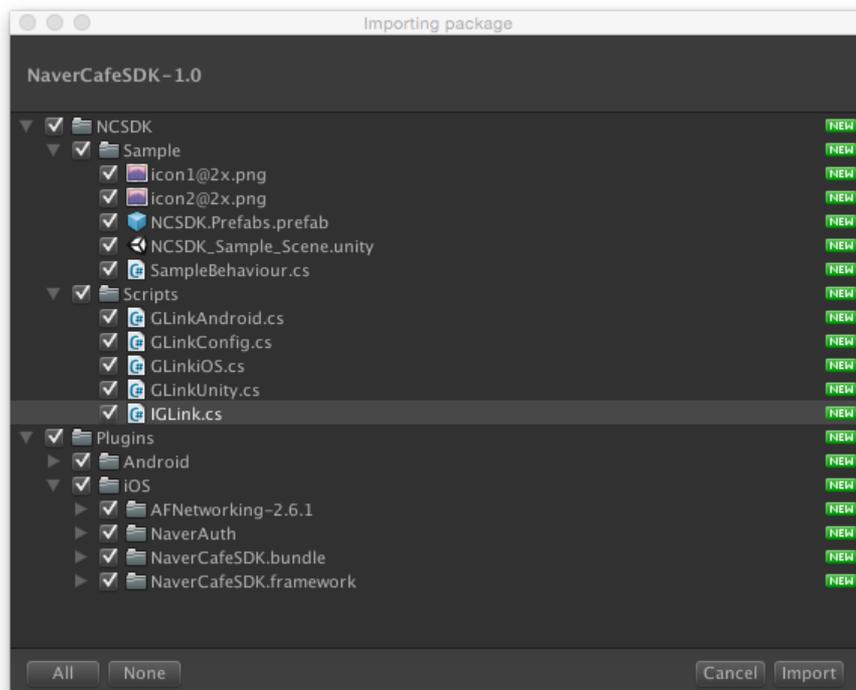
- NaverCafeSDK-1.0.unitypackage: Unity용 네이버 카페 SDK 라이브러리

### 개발 환경 설정

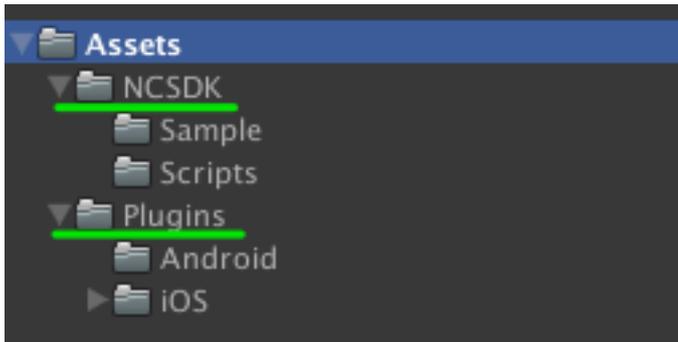
네이버 카페 SDK를 사용하려면 다음과 같이 개발 환경을 설정한다.

#### Unity 설정

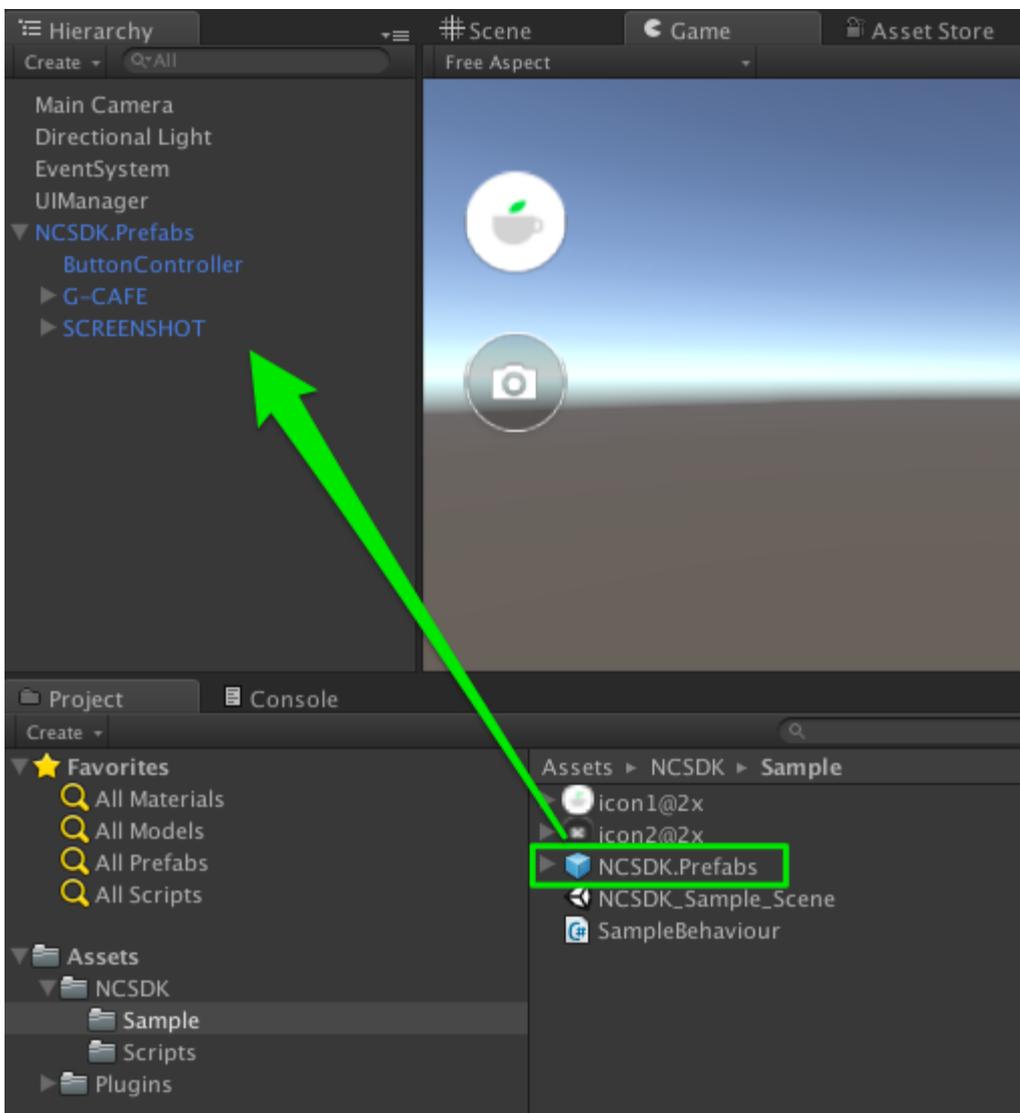
1. Unity의 메뉴에서 **Asset > Import package**를 클릭한 다음 **Custom Package** 대화 상자에서 다운로드한 **NaverCafeSDK-1.0.unityPackage** 파일을 선택한다.



2. **Custom Package** 대화 상자에서 **Import**를 클릭하면 **NCSDK** 폴더와 **Plugins** 폴더가 생성된다.



3. NCSDK 폴더의 **Sample** 폴더에 있는 **NCSDK.Prefabs**를 원하는 신으로 끌어다 놓는다. 네이버 카페 SDK에서 기본으로 제공하는 버튼이 생성된다.



4. NCSDK 폴더의 **Scripts** 폴더에 있는 **GLinkConfig** 파일에 카페 ID와 클라이언트 아이디, 클라이언트 시크릿을 입력한다.

```
//NCSDK/Scripts/GLinkConfig
static class GLinkConfig
{
```

```
public const string NaverLoginConsumerKey =
    "Consumer ID";

public const string NaverLoginConsumerSecret =
    "Secret ID";

public const int CafeId =
    00000000;
}
```

### Android 설정

Unity 5 엔진을 사용할 때는 다음과 같이 네이버 카페 SDK 라이브러리를 추가한다.

- 네이버 카페 SDK 라이브러리 파일(cafeSdk-x.x.x.aar 파일)과 필수 라이브러리 파일을 **Assets/Plugins/Android** 폴더에 추가한다.

Unity 4 엔진을 사용할 때는 Eclipse에서 다음과 같은 추가 설정이 필요하다.

1. 제공받은 네이버 카페 SDK 라이브러리 파일 가운데 cafeSdk-x.x.x.zip 파일의 압축을 해제한다.
2. 압축을 해제한 네이버 카페 SDK 라이브러리 폴더 아래의 **libs** 폴더에 빌드에 필요한 필수 라이브러리 파일을 추가한다.
3. Eclipse에서 압축을 해제한 폴더를 프로젝트로 불러온다.
4. 프로젝트 설정 대화 상자에서 **Android**를 클릭하고 **Is Library**를 선택한다.
5. **AndroidManifest.xml** 파일에 다음과 같이 네이버 카페 SDK에서 사용하는 액티비티를 추가한다.

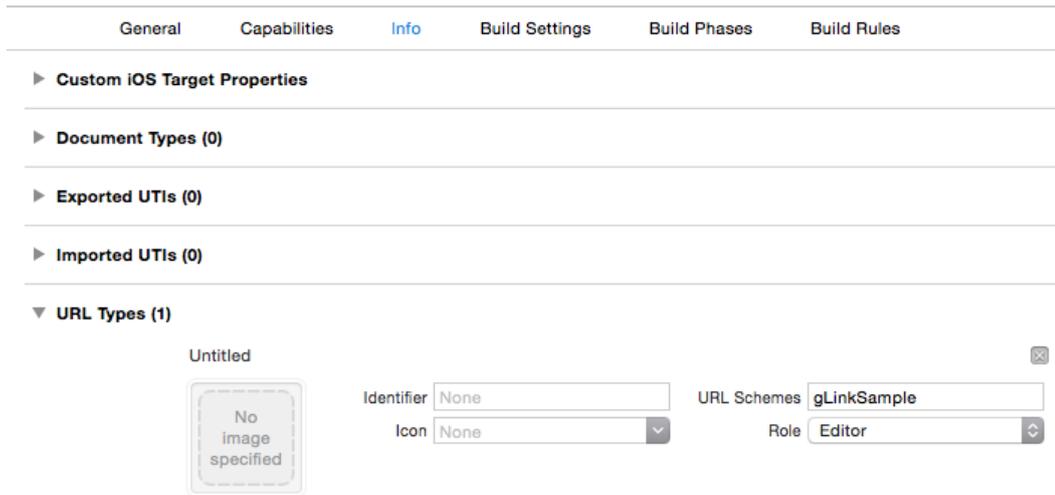
```
<activity
    android:name="com.nhn.android.naverlogin.ui.OAuthLoginActivity"
    android:screenOrientation="sensorLandscape"
    android:theme="@android:style/Theme.Translucent.NoTitleBar"/>
<activity
    android:name="com.nhn.android.naverlogin.ui.OAuthLoginInAppBrowserActivity"
    android:screenOrientation="sensorLandscape"
    android:label="OAuth2.0 In-app"/>
<activity
    android:name=".ui.VideoPlayActivity"
    android:screenOrientation="sensorLandscape"
    android:theme="@android:style/Theme.Translucent.NoTitleBar.Fullscreen"/>
```

6. 해당 프로젝트를 빌드한다.
7. Eclipse로 빌드한 파일을 Unity 프로젝트의 **Assets/Plugins/Android** 폴더에 추가한다.
8. Unity 프로젝트를 빌드한다.

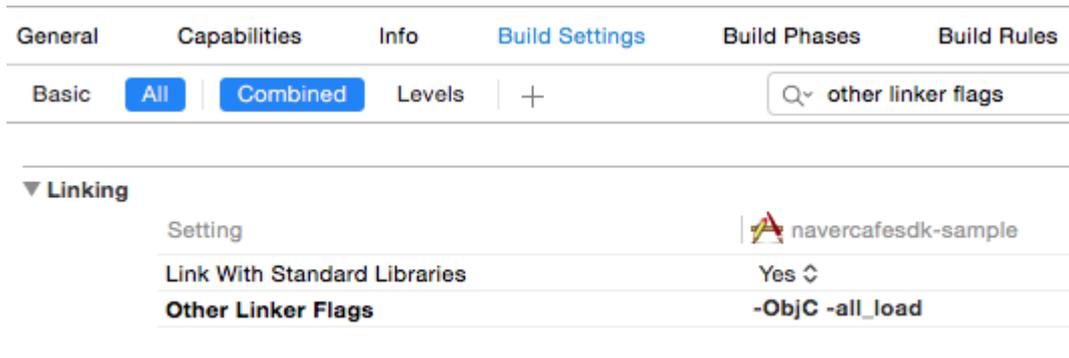
### iOS 설정

iOS 애플리케이션을 개발할 때는 다음과 같이 Xcode를 설정한다.

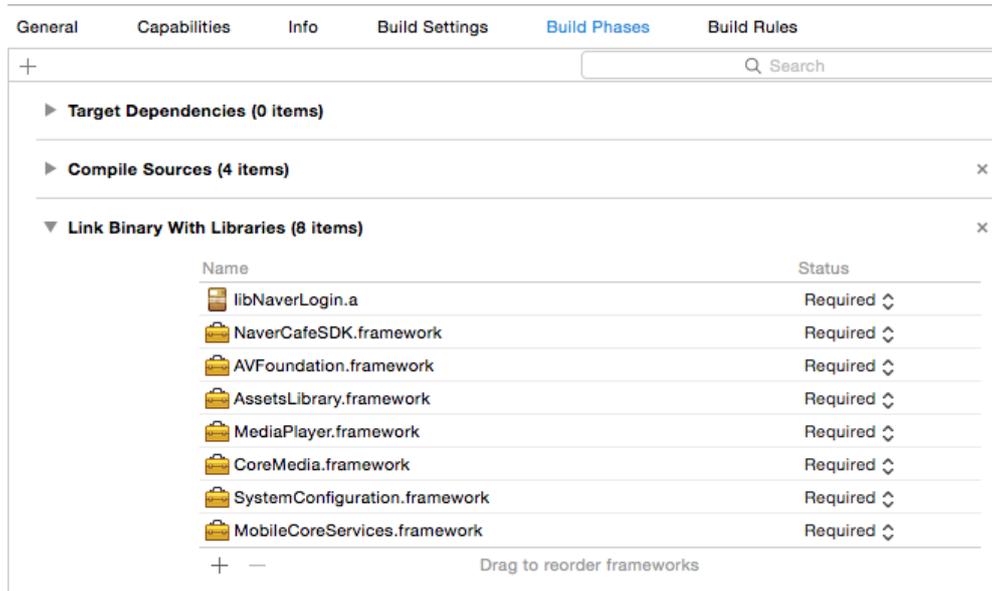
1. 네이버 아이디로 로그인에 애플리케이션을 등록할 때 입력한 URL Scheme을 Xcode 프로젝트에 등록한다.



2. 정적 라이브러리를 사용할 수 있게 **Build Settings**의 **Other Linker Flags**에 **-ObjC -all\_load** 옵션을 설정한다.



3. **Build Phases**의 **Link Binary With Libraries**에 다음 라이브러리를 추가한다.
  - MobileCoreServices.framework
  - SystemConfiguration.framework
  - MediaPlayer.framework
  - AVFoundation.framework
  - CoreMedia.framework
  - Security.framework
  - AssetsLibrary.framework



4. 다음과 같이 앱 델리게이트를 설정해 네이버 아이디로 로그인 이 완료되면 앱 델리게이트가 호출되게 한다.

```
//AppDelegate
# import <NaverCafeSDK/NCSDKLoginManager.h>

- (BOOL)application:(UIApplication *)application openURL:(NSURL *)url
sourceApplication:(NSString *)sourceApplication annotation:(id)annotation {
    return [[NCSDKLoginManager getSharedInstance] finishNaverLoginWithURL:url];
}
```

## 네이버 카페 SDK 적용

### 실행

네이버 카페 SDK 화면을 시작하려면 다음과 같이 `GlinkUnity.executeMain()` 메서드를 호출한다.

```
// 카페 홈
GlinkUnity.executeMain ();
```

게임 사용자가 카페에 게시글을 쓰는 화면은 다음과 같이 `GlinkUnity.executeArticlePostWithImgae()` 메서드로 구현한다.

```
//스크린샷
GlinkUnity.executeArticlePostWithImgae(menuId, "글 제목", "글 내용", image path);
```

`GlinkUnity.executeArticlePostWithImage()` 메서드나 `GlinkUnity.executeArticlePostWithVideo()` 메서드를 사용하면 이미지나 동영상이 첨부된 상태로 게시판 글쓰기 화면을 열 수 있다.

### 사용자의 게임 아이디와 네이버 아이디 연동

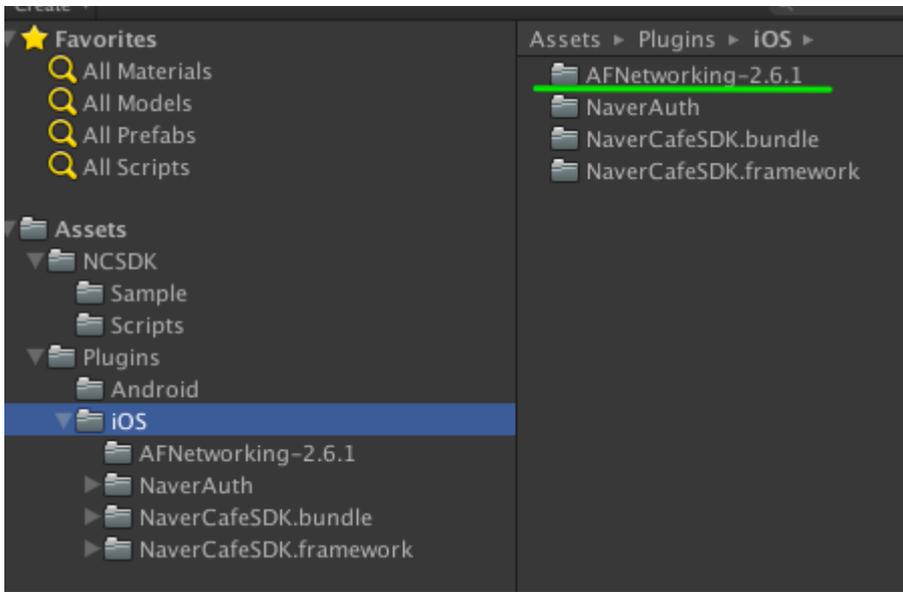
네이버 카페 SDK는 사용자의 게임 아이디와 네이버 아이디를 매핑하는 기능을 제공한다. PC에서 카페 관리자 페이지에 접속하면 매핑된 게임 아이디와 네이버 아이디의 목록을 확인할 수 있다.

다음은 iOS 애플리케이션에서 게임 아이디와 네이버 아이디를 매핑하는 기능을 구현한 예다.

```
/*  
네이버 아이디와 게임 아이디 연동을 위해 사용자의 게임 아이디를 연동한다.  
fieldName은 단순히 프로필 영역에 표시할 문자열이다(기본값: 게임 아이디)  
*/  
- (void)setGameUserId:(NSString *)gameUserId fieldName:(NSString *)fieldName;
```

## AFNetworking 교체

**NaverCafeSDK.unitypackage**에는 네트워크 통신을 위한 라이브러리인 AFNetworking 2.6.1이 기본으로 포함되어 있다. 프로젝트의 상황에 따라 AFNetworking 라이브러리를 삭제하거나 다른 버전으로 바꿔서 빌드해도 된다. 단, 네이버 카페 SDK는 AFNetworking 1.x 이상을 지원한다.



## API 레퍼런스

### GLinkUnity

네이버 카페 SDK를 제어하는 클래스. GLinkUnity 클래스의 메서드를 사용해 네이버 카페 SDK를 시작하거나 글쓰기 화면을 여는 기능을 구현한다.

GLinkUnity 클래스의 메서드는 다음과 같다.

- executeArticlePost()
- executeArticlePostWithImage()
- executeArticlePostWithVideo()
- executeMain()

## executeArticlePost()

### 설명

게시판 글쓰기 화면을 연다.

### 구문

```
public static void executeArticlePost(int menuId, string subject, string content) {
    sharedInstance().executeArticlePost (menuId, subject, content);
}
```

### 파라미터

파라미터	타입	필수 여부	설명
menuId	int	Y	게시판 아이디(기본값: -1). 게시판 아이디는 네이버 카페 게시판 URL 가운데 <b>menuid</b> 파라미터의 값이다.
subject	string	Y	게시글 제목
content	string	Y	게시글 본문

### 반환값

없음

### 코드 예

```
GlinkUnity.executeArticlePost(28290504, "글 제목", "글 본문" );
```

## executeArticlePostWithImage()

### 설명

이미지가 첨부된 상태로 게시판 글쓰기 화면을 연다.

### 구문

```
public static void executeArticlePostWithImage(int menuId, string subject, string content, string filePath) {
    sharedInstance().executeArticlePostWithImage (menuId, subject, content, filePath);
}
```

### 파라미터

파라미터	타입	필수 여부	설명
menuId	int	Y	게시판 아이디(기본값: -1). 게시판 아이디는 네이버 카페 게시판 URL 가운데 <b>menuid</b> 파라미터의 값이다.
subject	string	Y	게시글 제목
content	string	Y	게시글 본문
filePath	string	Y	이미지 파일의 경로(URI 형식)

**반환값**

없음

**코드 예**

```
{
GlinkUnity.executeArticlePostWithImgae (28290504, "글 제목", "글
내용", "/navercafesdk/glink.png" );
}
```

**executeArticlePostWithVideo()**

**설명**

동영상이 첨부된 상태로 게시판 글쓰기 화면을 연다.

**구문**

```
public static void executeArticlePostWithVideo(int menuId, string subject, string
content, string filePath) {
    sharedInstance().executeArticlePostWithVideo (menuId, subject, content, filePath);
}
```

**파라미터**

파라미터	타입	필수 여부	설명
menuId	int	Y	게시판 아이디(기본값: -1). 게시판 아이디는 네이버 카페 게시판 URL 가운데 <b>menuid</b> 파라미터의 값이다.
subject	string	Y	게시글 제목
content	string	Y	게시글 본문
filePath	string	Y	동영상 파일의 경로(URI 형식)

**반환값**

없음

**코드 예**

```
{
GlinkUnity.executeArticlePostWithImgae (28290504, "글 제목", "글
내용", "/navercafesdk/glink.avi" );
}
```

**executeMain()**

**설명**

네이버 카페 SDK 화면을 연다.

**구문**

```
public static void executeMain() {
```

```
sharedInstance().executeMain ();  
}
```

**파라미터**

없음

**반환값**

없음

**코드 예**

```
{  
  GlinkUnity.executeMain ();  
}
```