

Mobile Service > IAP > iOS SDK 사용 가이드

[공지]

구독 결제를 지원하는 신규 IAP SDK가 [TOAST SDK](#)로 출시됐습니다.
기존 IAP SDK는 신규 기능을 개발하지 않을 예정입니다.

개발 환경

- OSX is required
- Xcode 9.0.0 and higher
- IAP SDK Support for iOS 7.x and higher

IAP SDK 사용을 위해서는 어플리케이션에 아래의 Framework를 추가 해야 합니다.

이름	설명
StoreKit.framework	앱스토어의 In App Purchase연동을 위한 framework
libsqlite3.tbd	TOAST Cloud IAP SDK는 로컬데이터관리를 위해 sqlite를 사용합니다
CoreTelephony.framework	사용자의 국가정보를 획득하기 위해 사용합니다

[참고]

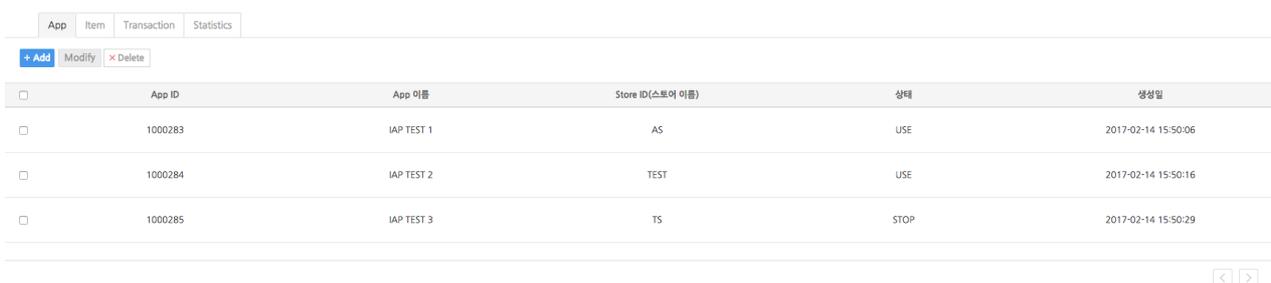
In App Purchase 테스트를 하기 위해 iTunes Connect에 어플리케이션 및 상품등록을 완료했다고 가정합니다.

[iTunes Connect](#)

IAP Console

1. 스토어등록 - APP ID 획득

1. [App] 탭 선택 > [추가] 버튼 클릭
2. [Store ID]에서 AS(Apple Store) 선택
 - 마켓 연동을 위한 정보 입력(Market APP ID : Bundle Id)
3. [APP ID] 확인



The screenshot shows the IAP Console interface with a table of App IDs. The table has columns for App ID, App 이름, Store ID(스토어 이름), 상태, and 생성일. There are three rows of data.

App ID	App 이름	Store ID(스토어 이름)	상태	생성일
1000283	IAP TEST 1	AS	USE	2017-02-14 15:50:06
1000284	IAP TEST 2	TEST	USE	2017-02-14 15:50:16
1000285	IAP TEST 3	TS	STOP	2017-02-14 15:50:29

[그림 1 APP ID 획득]

2. 아이템 등록

1. [Item] 탭 선택 > [추가] 버튼 클릭
2. [Store ID]에서 AS(Apple Store) 선택
3. [아이템 정보 입력]
 - Item Name : 아이템의 이름
 - Store Item ID : iTunes Connect에 등록된 어플리케이션의 아이템의 Product ID
4. [ITEM] 확인

Xcode 프로젝트 설정하기

디렉토리명	설명
/docs	IAP iOS SDK API Reference
/lib	Framework
/samples	Sample Application

[표1 iOS SDK 디렉토리 정보]

1. IAP SDK 및 framework 추가

1. [Xcode] > [Project] > [Targets – Build Phases]
2. [Link Binary With Libraries] 에 아래의 frameworks 추가
 - libsqlite3.tbd
 - TIAPurchase.framework
 - StoreKit.framework
 - CoreTelephony.framework

▼ Linked Frameworks and Libraries

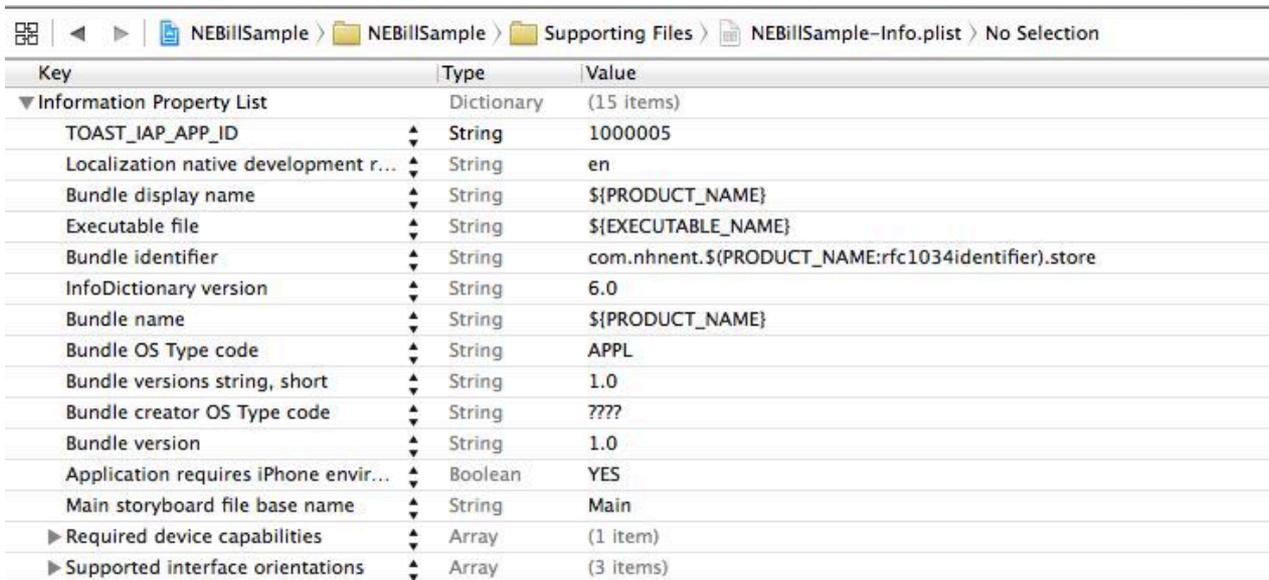
Name	Status
 TIAPurchase.framework	Required ⇅
 libsqlite3.tbd	Required ⇅
 CoreTelephony.framework	Required ⇅
 StoreKit.framework	Required ⇅
 CoreGraphics.framework	Required ⇅
 UIKit.framework	Required ⇅
 Foundation.framework	Required ⇅

+ -

[그림 2 IAP 연동을 위한 라이브러리 추가]

2. plist 설정하기

[plist] 에서 TOAST_IAP_APP_ID 가 KEY인 string value를 생성하고, APP ID를 입력 합니다.
.plist 는 아래와 같은 형태일 것입니다.



Key	Type	Value
▼ Information Property List	Dictionary	(15 items)
TOAST_IAP_APP_ID	String	1000005
Localization native development r...	String	en
Bundle display name	String	\$(PRODUCT_NAME)
Executable file	String	\$(EXECUTABLE_NAME)
Bundle identifier	String	com.nhnet.\$(PRODUCT_NAME:rfc1034identifier).store
InfoDictionary version	String	6.0
Bundle name	String	\$(PRODUCT_NAME)
Bundle OS Type code	String	APPL
Bundle versions string, short	String	1.0
Bundle creator OS Type code	String	????
Bundle version	String	1.0
Application requires iPhone envir...	Boolean	YES
Main storyboard file base name	String	Main
▶ Required device capabilities	Array	(1 item)
▶ Supported interface orientations	Array	(3 items)

[그림 3 plist에 APP ID 설정]

[참고-iOS9 ATS 설정]

iOS9 SDK부터 ATS(App Transport Security)관련 설정을 적용해야 합니다.

XCode7 이상에서 빌드할 경우 아래와 같이 IAP 특정 도메인에 대한 예외처리를 설정해야 합니다.

[Apple Document](#)

```
<key>NSAppTransportSecurity</key>
  <dict>
    <key>NSExceptionDomains</key>
    <dict>
      <key>api-iap.cloud.toast.com</key>
      <dict>
        <key>NSIncludesSubdomains</key>
        <true/>
        <key>NSEnvironmentAllowsInsecureHTTPLoads</key>
        <true/>
        <key>NSEnvironmentRequiresForwardSecrecy</key>
        <false/>
      </dict>
    </dict>
  </dict>
</dict>
```

API Reference

1. 헤더 파일 추가

어플리케이션에 SDK 사용을 위한 준비가 완료되면, IAP SDK의 Header File을 아래와 같이 추가합니다.

```
#import <TIAPurchase/TIAPurchase.h>
```

2. 로그정보 활성화

디버그를 위한 로그 정보에 대한 노출을 활성화 합니다.

[Example Code]

```
[TIAPurchase setDebugMode:YES];
```

3. 유저 등록

애플리케이션에서 사용자에게 대한 인증 이후에 사용자 식별이 가능한 값을 등록합니다.

[Example Code]

```
NSError *purchaseError = nil;
BOOL result = [TIAPurchase registerUserId:@"user001" error:&purchaseError];
If (!result) {
    // An error occurred, we need to handle the error.
    NSLog(@"errorInfo = %@", purchaseError);
}
// register user id succceccfully.
```

4. 결제 요청

인앱 결제 요청을 합니다. 결제가 성공적으로 완료되면 completionHandler 를 통해 결제내역이 전달 됩니다.

[Example Code]

```
[TIAPurchase startPurchaseWithViewController:self itemId:1000004
completionHandler:^(id result, NSError *error) {

    if (error)
    {
        // An error occurred, we need to handle the error
        NSLog(@"purchase error, %@ %d", [error domain], [error code]);
        return;
    }
    else
    {
        /**
         Success! Include your code to handle the results here

         JSON data to 'NSDictionary', This is nil if there was an error.
         [keys]
         - paymentSeq : generated payment id
         - itemSeq : item id
        */
    }
}
```

```

        - purchaseToken : represent to token for consume by
server.
        - currency : represent to item currency
        - price : represent to item price.
    */

    NSDictionary *purchaseResult = result;
    NSLog(@"purchase success, purchase = %@", purchaseResult);
}

}];

```

5. 미소비 결제 내역 조회

결제 내역을 조회 합니다.

[Example Code]

```

[TIAPurchase purchasesWithCompletionHandler:^(id result, NSError *error) {
    if (error)
    {
        // An error occurred, we need to handle the error
        NSLog(@"purchasesWithCompletionHandler occured error, %@ %d", [error
domain], [error code]);
        return;
    }

    /**
    Success! Include your code to handle the results here

    JSON data to 'NSArray', This is nil if there was an error.
    [keys]
    - paymentSeq : generated payment id
    - itemSeq : item id
    - purchaseToken : represent to token for consume by server.
    - currency : represent to item currency.
    - price : represent to item price.
    */

    NSArray *purchases = result;
    NSLog(@"purchasesWithCompletionHandler, size:%d npurchases:%@", [purchases
count], purchases);

}];

```

6. 구매 가능한 아이템 내역 조회

구매 가능한 모든 아이템 내역을 조회합니다.

[Example Code]

```
[TIAPurchase itemListWithCompletionHandler:^(id result, NSError *error) {
    if (error)
    {
        // An error occurred, we need to handle the error
        NSLog(@"itemListWithCompletionHandler ocured error, %@ %d", [error
domain], [error code]);
        return;
    }

    /**
    Success! Include your code to handle the results here

    JSON data to 'NSArray', This is nil if there was an error.
    [keys]
    - itemSeq : item id
    - itemName : item name
    - usingStatus : item status on IAP server
    - regYmdt : item registration date on IAP server
    - appName : app name
    - marketId : market id (AS : APPLE STORE)
    - marketItemId : market item id (product id)
    - currency : represent to item currency
    - price : represent to item price
    - localizedPrice - represent to localized item price
    */

    NSArray *itemList = result;
    NSLog(@"itemListWithCompletionHandler, size:%lu \nitemList:%@", [itemList
count], itemList);
}];
```

7. 미처리 결제건 일괄 재처리

미처리된 결제건(IAP 서버 검증 실패)들에 대해 일괄로 재처리 작업을 진행합니다.

[Example Code]

```
[TIAPurchase processesIncompletePurchasesWithCompletionHandler:^(id result,
NSError *error) {
    if (error)
    {
        // An error occurred, we need to handle the error
        NSLog(@"processesIncompletePurchasesWithCompletionHandler ocured
error, %@ %d", [error domain], [error code]);
        return;
    }
}
```

```

/**
 Include your code to handle the results here

 JSON data to 'NSDictionary'.
 [keys]
 - successList : success data list (NSArray)
     [keys]
     - paymentSeq : generated payment id
     - itemSeq : represent item id
     - purchaseToken : represent token for validation
     - marketItemId : market item id (product id)
     - currency : represent to item currency
     - price : represent to item price

 - failList : fail data list (NSArray)
     [keys]
     - paymentSeq : generated payment id
     - itemSeq : represent item id
     - purchaseToken : represent token for validation
     - marketItemId : market item id (product id)
     - currency : represent to item currency
     - price : represent to item price
 */
NSDictionary *data = result;
NSLog(@"processesIncompletePurchasesWithCompletionHandler data:%@",
data);
}];

```

8. 결제 소비

사용자 애플리케이션 서버는 아이템을 지급하기 전 IAP 서버에게 결제를 소비할 것을 알려야 합니다. 결제 소비를 위한 API는 아래를 참고 해주세요.

[참고]

[Payment Consume API](#)

9. AppStore 프로모션 결제

AppStore 프로모션 결제를 사용한다면 Delegate 를 등록해 결제 내역을 전달 받으세요.

초기화 이전에 Delegate를 등록해야 정상적으로 결제 내역을 전달 받을 수 있습니다.

[Example Code]

TIAPurchaseDelegate

```

@interface ViewController () <TIAPurchaseDelegate>
...
@end

// Delegate 등록

```

```

[TIAPurchase setDelegate:self];

// Delegate 구현
- (void)didFinishPurchasingItemFromStore:(TIAPItem *)item
    paymentSeq:(NSString *)paymentSeq
    purchaseToken:(NSString *)purchaseToken
    withError:(NSError *)error {

    if (error == nil) {
        NSLog(@"AppStore 프로모션 결제 성공");
    } else {
        NSLog(@"AppStore 프로모션 결제 실패");
    }
}

```

TIAPItem

```

@interface TIAPItem : NSObject

@property (nonatomic, copy, readonly) NSString *itemSeq;
@property (nonatomic, copy, readonly) NSString *storeId;
@property (nonatomic, copy, readonly) NSString *marketItemId;
@property (nonatomic, copy, readonly) NSString *currency;
@property (nonatomic, copy, readonly) NSNumber *price;
@property (nonatomic, copy, readonly) NSString *localizedPrice;

@end

```