Mobile Service > IAP > Android SDK Guide

> [Notice]
> New IAP SDK suppoting subscription are released as [TOAST SDK](#).
> The existing IAP SDK will not be developing new features.

Development Environment

- Windows
- Eclipse in Android Development Tools / Android Studio IDE 1.2
- Android 5.0.1 (API Level 21)
- Android SDK Version is beyond **2.3.3 (API Level 10)**

The open sources in use are as follows.

Name	Reference	Version	License
okhttp	http://square.github.io/okhttp/	1.5.4	Apache License 2.0
gson	https://code.google.com/p/google-gson/	2.2.4	Apache License 2.0

Using in Android Studio & Gradle

IAP's Android SDK offers development environment for Gradle-based Android Studio IDE. Remote downloading is available from jCenter Maven Repository. Define repository and dependency in build.gradle file of the project as below

Gradle Repository

```
buildscript {
    repositories {
        jcenter()
    }
}
```

```
dependencies {
    compile group: 'com.toast.iap', name: 'iap', version:
project.TOAST_IAP_SDK_VERSION
}
```

> [Reference]
> Specify SDK version in project.TOAST_IAP_SDK_VERSION. Using Gradle Project beyond version 1.1.0 is recommended.

Implementing Sample Application

Import Project

Import project from Android Studio and add /toast-iap-android-sdk-studio to SDK package.

Setting Market Information to AndroidManifest.xml

Set appId / market using information registered to IAP Web Console

Using in Eclipse

IAP Android SDK is distributed in Android Library Project format.

Directory Name	Description
/docs	API reference
/toast-iap-android-sdk-eclipse/iap	IAP Android library project
/toast-iap-android-sdk-eclipse/iap-sample	Sample application

[Table.1 Android SDK Directory Information]

How To Import

```
[Eclipse] > [File] > [Import] select.
[Android] – [Existing Android Code into Workspace] select.
<Import Projects> 창 > [Browse] butoon > [IAP SDK] folder select.
Once IAP SDK & sample application project are displayed in Import list, select
[Finish] while all projects are selected.
```

[Reference]
If error occurs in Project after importing, follow below instructions
- [project] > [clean] select.
- [IAP] right click > [Properties] > [Resource] > [Text File Encoding] set to UTF-8.

Add IAP Android SDK

```
Right Click of [project] > [Properties] > [Android] > [Add] > Add IAP
```
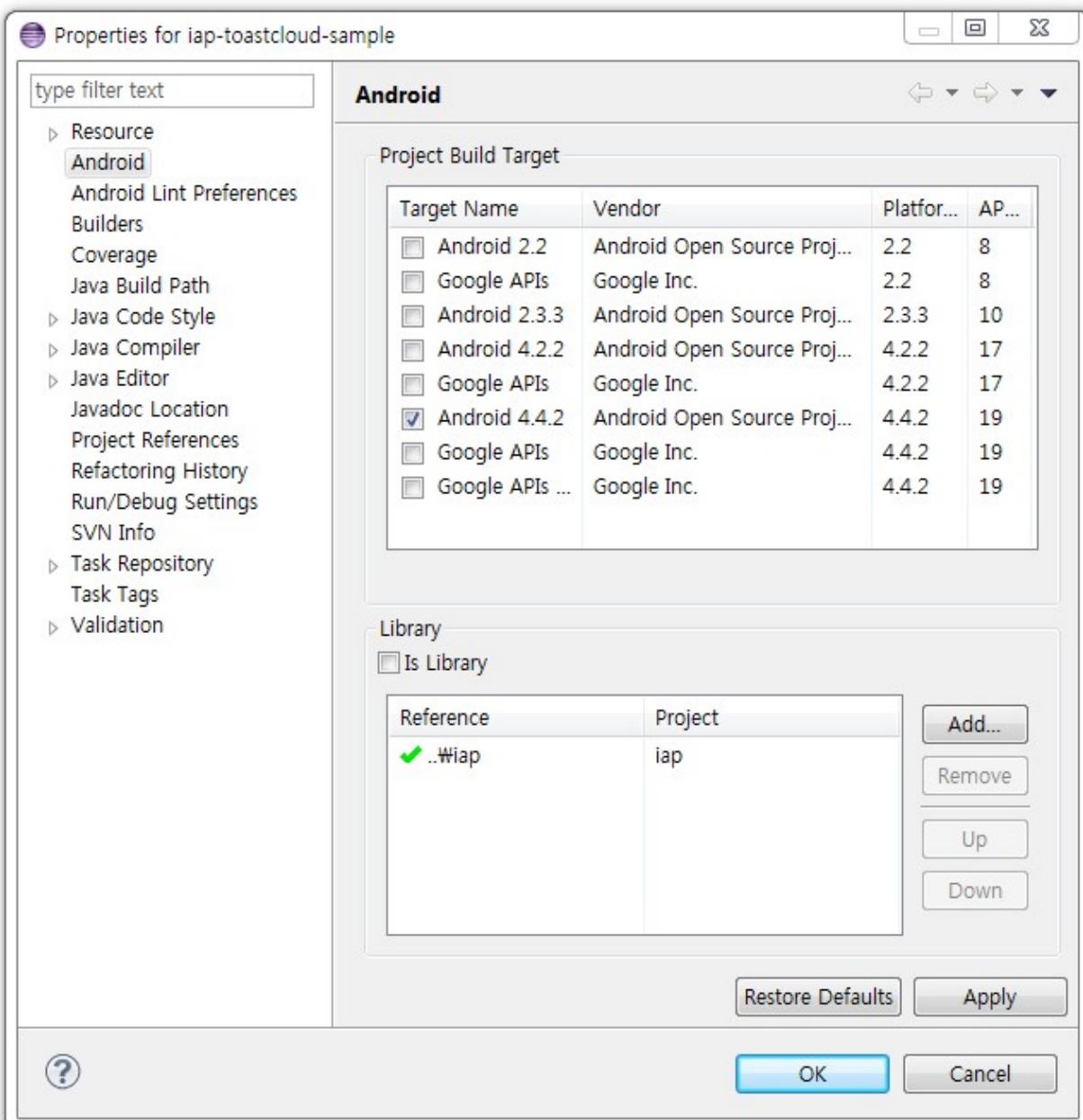
[Figure 1 IAP SDK]

Setting Android Project

AndroidManifest.xml Modification

This is for Google Play development.

```xml
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
<!-- google iab permission -->
<uses-permission android:name="com.android.vending.BILLING" />

<application>
...
        <activity android:name="com.nhnent.mobill.api.core.IAPActivity"

 android:configChanges="keyboardHidden|orientation|screenSize|locale|layoutDir
ection"
            android:theme="@android:style/Theme.Translucent.NoTitleBar"
            android:windowSoftInputMode="adjustResize|stateHidden" />
        <meta-data android:name="com.toast.iap.config.appId"
android:value="1000000" />
        <meta-data android:name="com.toast.iap.config.market"
android:value="GG" />
</application>
```

Setting information for AndroidManifest.xml is as follows.

Property Name	Description
android.permission	Common Android authority for IAP use * Duplicate permission can be omitted.
com.android.vending.BILLING	Authority for Google Play. * Names and/or contents may differ depending on store.
activity	Register activity offered by IAP
com.toast.iap.config.appId	App identifier issued by IAP web console
com.toast.iap.config.market	Store GG : Google Play AS : Apple App Store TS : One Store TEST : Test

[Table.2 AndroidManifest.xml Information]

IAP Payment Flow Diagram

In-app purchase is proceeded in two stages: payment request and payment consume.
If payment consume is complete, issue item in user's application.

Registering User Identifier

Register authorized user ID.
This is user identifier defined by developer, which qualifies for item issuance.

[Request Example]

```
InAppPurchases.InAppPurchase.registerUserId(userId); // userId : String value
```

Inquiry All Purchasable Items

Inquire All Purchasable Items

[Request Example]

```
InAppPurchases.InAppPurchase.queryItems(activity, new
InAppPurchase.ItemListCallback() {
    @Override
    public void onCallback(JSONArray result, InAppPurchaseException exception)
{
        if (exception != null) {
            // An error occurred, we need to handle the error
            return;
        }
        // Success! Include your code to handle the results here
    }
});
```

[Method]

Word	Description
Syntax	public void queryItems(Activity activity, ItemListCallback callback)
Parameters	activity [in] of current application.
Parameter	callback [in] which returns response of API.
Return Value	void

[Response Example]

```
[
    {
        "itemSeq" : 1000208,
        "itemName" : "Test item 01",
        "marketItemId": "item01",
        "price": 1000,
        "currency": "KRW",
        "localizedPrice":"₩1,000"
    },
    {
        "itemSeq" : 1000209,
        "itemName" : "Test item 02",
        "marketItemId": "item02",
        "price": 7.99,
        "currency": "USD",
        "localizedPrice":"$7.99"
}]
```

Request Payment

Request item purchase from client. Response for payment request will be received via PurchaseCallback. Once payment is successfully done, send result to server to consume payment details.

[Request Example]

```
InAppPurchases.InAppPurchase.requestPurchase(this, 1000001, new
PurchaseCallback() {

    @Override
    public void onCallback(JSONObject result, InAppPurchaseException exception)
{
        if (!result.isSuccess()) {
            // An error occurred, we need to handle the error
            return;
        }
        // Success! Include your code to handle the results here
    }
});
```

[Method]

Word	Description
Syntax	public void requestPurchase(Activity activity, long itemId, String currency, float price, PurchaseCallback callback)
Parameters	activity [in] of current application.
Parameters	itemId [in] generated in Web Console [Item]
Parameters	callback [in] which returns response of API.
Return Value	void

[Response Example]

```
{
    "paymentSeq": "2014082210002092",
    "purchaseToken":
"5PYSHgisiCU8BditHnDbPhmlS/0DSt4JDs2UMyg1/EY8oC6Q8qkuw5VBo7GNrBYLNUy656GCAh7h9e1
BtXeoBA==",
    "itemSeq": 1000001,
    "currency": "KRW",
    "price": 1000.0
}
```

[Reference]
IAP Android SDK payment confirmation pop-up
There will be no payment confirmation pop-up appearing from toast-iap-android-sdk-1.2.3;
it will be directly processed to market purchase.
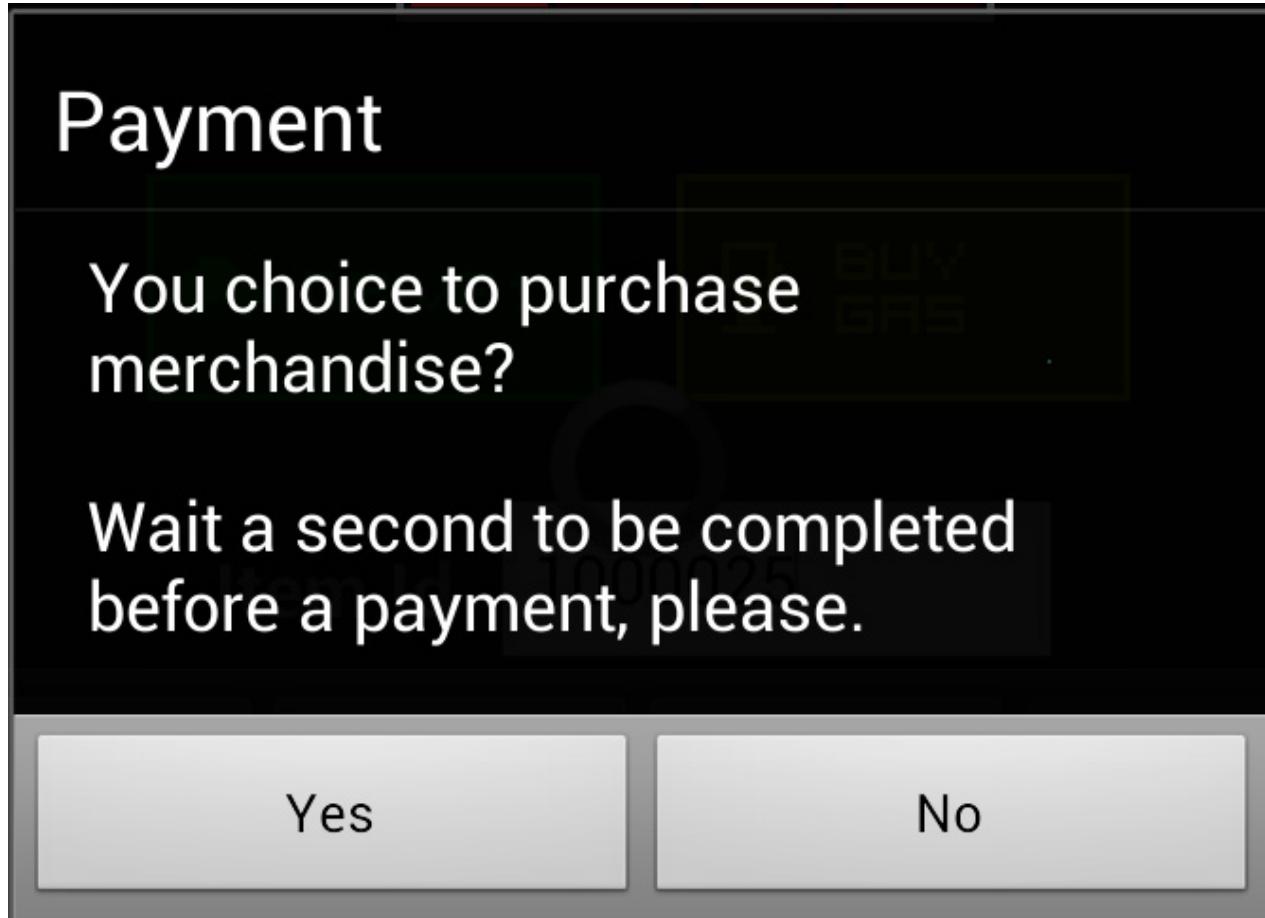
[Figure 2 Payment confirmation pop-up supported in old version]

Payment Consume

User application server should notify IAP server for payment consume before issuing item. Use Payment Purchase Token to check validity of the payment between user server and IAP server.

> [Reference]
> [Server Payment Consume API](#)

[HTTP Request Example]

```
POST
https://api-iap.cloud.toast.com/inapp/v3/consume/{paymentSeq}/items/{itemSeq}

RequestBody
{
 "purchaseToken":string
}
```

[Response Example]

```json
{
    "header": {
        "resultCode": 0,
        "resultMessage": "request is successful",
        "isSuccessful": true
    },
    "result": {
        "price":1000.0,
        "currency":"KRW"
    }
}
```

Inquiry unconsumed user payment history.

Inquire unconsumed user payment history.

[Request Example]

```
InAppPurchases.InAppPurchase.queryPurchases(this, new PurchaseListCallback() {

    @Override
    public void onCallback(JSONArray result, InAppPurchaseException exception) {
            if (!result.isSuccess()) {
                // An error occurred, we need to handle the error
                return;
            }
            // Success! Include your code to handle the results here }
});
```

[Method]

Word	Description
Syntax	public void queryPurchases(Activity activity, PurchaseListCallback callback)
Parameters	activity [in] of current application.
Parameter	callback [in] which returns response of API.
Return Value	void

[Response Example]

```
[{
    "paymentSeq": "2014082210002092",
    "purchaseToken":
"5PYSHgisiCU8BditHnDbPhmlS/0DSt4JDs2UMyg1/EY8oC6Q8qkuw5VBo7GNrBYLNUy656GCAh7h9e1
BtXeoBA==",
    "itemSeq": 1000208,
```

```
    "currency": "KRW",
    "price": 1000.0
}, {
    "paymentSeq": "2014082210002093",
    "purchaseToken":
"Q+os4dDsYaGiEEqkLeXQfhmlS/0DSt4JDs2UMyg1/EY8oC6Q8qkuw5VBo7GNrBYLNUy656GCAh7h9e1
BtXeoBA==",
    "itemSeq": 1000208,
    "currency": "KRW",
    "price": 1000.0
}, {
    "paymentSeq": "2014082210002094",
    "purchaseToken":
"GMBcODtMnX306wVlFGIcDRmlS/0DSt4JDs2UMyg1/EY8oC6Q8qkuw5VBo7GNrBYLNUy656GCAh7h9e1
BtXeoBA==",
    "itemSeq": 1000208,
    "currency": "KRW",
    "price": 1000.0
}]
```

Batch process of unconsumed payment

processes whole of unconsumed(cause of verification failure or network loss) payments.

[Request Example]

```
InAppPurchases.InAppPurchase.processesIncompletePurchases(activity, new
InAppPurchase.IncompletePurchasesCallback() {

    @Override
    public void onCallback(JSONObject result, InAppPurchaseException exception)
{
        if (exception != null) {
            // An error occurred, we need to handle the error
            return;
        }
        // Success! Include your code to handle the results here }
});
```

[Method]

Word	Description
Syntax	public void processesIncompletePurchases(Activity activity, IncompletePurchasesCallback callback)
Parameters	activity [in] of current application.
Parameter	callback [in] which returns response of API.
Return Value	void

[Response Example]

```
{
    "successList": [
        {
                "paymentSeq" : "2014082510002163",
                "purchaseToken" :
"8nkx3SzHKlI74vmgQLzHExmlS/0DSt4JDs2UMyg1/EY8oC6Q8qkuw5VBo7GNrBYLNUy656GCAh7h9e1
BtXeoB-AB",
                "itemSeq" : 1000208,
                "marketItemId"  : "item01",
                "currency" : "KRW",
                "price" : 1000.0
        },
        {
                "paymentSeq" : "2014082510002164",
                "purchaseToken" :
"8nkx3SzATKlI74vmgQLzHExmlS/0DSt4JDs2UMyg1/EY8oC6Q8qkuw5VBo7GNrBYLNUy656GCAh7h9e
1BtXeoBaAC",
                "itemSeq" : 1000209,
            "marketItemId"      : "item02",
                "currency" : "KRW",
                "price" : 1000.0
        }
    ],
    "failList": [
        {
                "paymentSeq" : "2014082510002165",
                "purchaseToken" : null,
                "itemSeq" : 1000210,
                "marketItemId"  : "item03",
                "currency" : "KRW",
                "price" : 1000.0
        }
    ]
}
```

Processing Error after Calling API

InAppPurchaseException class delivers error information of API call.

```
InAppPurchases.InAppPurchase.queryPurchases(activity, new PurchaseListCallback()
{

    @Override
    public void onCallback(JSONArray result, InAppPurchaseException exception) {
            if (exception != null) {
                int errorCode = exception.getErrorCode();
                String errorMessage = exception.getMessage();
                // TODO : do something when error occurs.
                ....
                return;
            }
    }
});
```

- errorCode - error code
- errorMessage - error detail message

> [Reference]
> [Error Code Guide](#)

Android Sample Application

Sample application can be imported in Eclipse + ADT development environment.

```
[Eclipse] > [File] > select [Import]
[Android – Existing Android Code into Workspace] select > select [Next]
<Import Projects> window > click [Browse] button > select [IAP Android SDK]
folder
you can see [IAP Android SDK] and sample application project in [Import] list.
check projects to be imported
check [Copy Project into Workspace]
click [Finish] button and Import completed
```
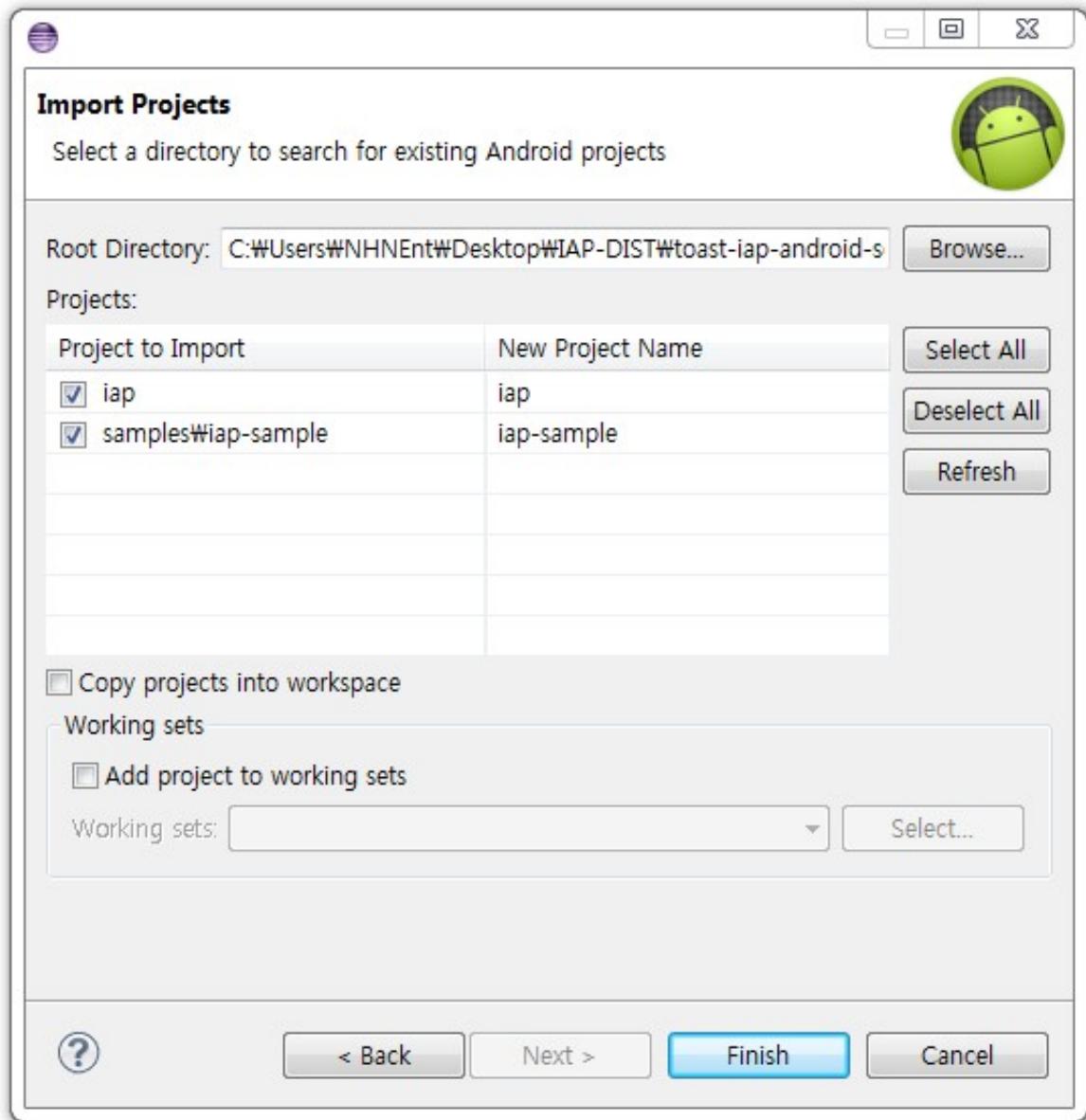
[Figure 3 Add sample application]

Using Test Store

Set market code to TEST to test API provided by IAP Android SDK without actual market sync information.

```
<meta-data android:name="com.toast.iap.config.market" android:value="TEST" />
```

Sample Code

Refer to IAPServiceProvider class of com.nhnent.iap.sample package.

Android Reference

Package: com.toast.android.iap

public interface InAppPurchase

Interface for In app purchase payment.

[Method Summary]

name	Return Value	Parameter
setDebugMode	void	boolean isDebuggable
registerUserId	void	String userId
requestPuchase	void	Activity activity, long itemId, PurchaseCallback callback
queryPurchases	void	Activity activity, PurchaseListCallback callback

[setDebugMode]

Word	Description
Description	Set IAP SDK log information activation.
Syntax	public void setDebugMode(boolean isDebuggable)
Parameters	isDebuggable [in] Expose log information when true.

[Example Code]

```
InAppPurchases.InAppPurchase.setDebugMode(true);
```

[registerUserId]

Word	Description
Description	Register user identifier after user verification in application. It is not store account.
Syntax	public void registerUserId(String userId)
Parameters	userId [in] userId shall be unchanging unique value as user identifier.
Return Value	void

[Example Code]

```
InAppPurchases.InAppPurchase.registerUserId("guest0001");
```

[requestPurchase]

Word	Description
Description	Request in-app purchase. Response for payment request is delivered via PurchaseCallback interface. Register item information via Web Console.
Syntax	public void requestPurchase(Activity activity, long itemId, PurchaseCallback callback)
Parameters	activity [in] of current application.
Parameters	itemId [in] generated in Web Console
Parameter	callback [in] which returns response of API.
Return Value	void

[Response (JSON)]

Attribute	Value	Description
paymentSeq	String	unique payment identifier for completed payment.
itemSeq	Long	item identifier

| purchaseToken | String | Token information required for requesting payment consume in IAP server from application server. | currency | String | currency | | price | Float | price |

[Response Example]

```
{
    "paymentSeq": "2014082210002092",
    "purchaseToken":
"5PYSHgisiCU8BditHnDbPhmlS/0DSt4JDs2UMyg1/EY8oC6Q8qkuw5VBo7GNrBYLNUy656GCAh7h9e1
BtXeoBA==",
    "itemSeq": 1000001,
    "currency": "KRW",
    "price" : 1000.0
}
```

[queryPurchases]

Word	Description
Description	Inquire unconsumed payment history.
Syntax	public void queryPurchases(Activity activity, PurchaseListCallback callback)
Parameters	activity [in] of current application.
Parameter	callback [in] which returns response of API.
Return Value	void

[Response (JSON)]

Attribute	Value	Description
paymentSeq	String	unique payment identifier for completed payment.
purchaseToken	String	Token information required for requesting payment consume in IAP server from application server.
itemSeq	Long	item identifier
currency	String	currency
price	Float	price

[Response Example]

```
[{
    "paymentSeq": "2014082210002092",
    "purchaseToken":
"5PYSHgisiCU8BditHnDbPhmlS/0DSt4JDs2UMyg1/EY8oC6Q8qkuw5VBo7GNrBYLNUy656GCAh7h9e1
BtXeoBA==",
    "itemSeq": 1000208,
    "currency": "KRW",
    "price" : 1000.0

}, {
    "paymentSeq": "2014082210002093",
    "purchaseToken":
"Q+os4dDsYaGiEEqkLeXQfhmlS/0DSt4JDs2UMyg1/EY8oC6Q8qkuw5VBo7GNrBYLNUy656GCAh7h9e1
BtXeoBA==",
    "itemSeq": 1000208,
    "currency": "KRW",
    "price" : 1000.0

}, {
    "paymentSeq": "2014082210002094",
    "purchaseToken":
```

```
"GMBcODtMnX306wVlFGIcDRmlS/0DSt4JDs2UMyg1/EY8oC6Q8qkuw5VBo7GNrBYLNUy656GCAh7h9e1
BtXeoBA==",
    "itemSeq": 1000208,
    "currency": "KRW",
    "price" : 1000.0

}]
```

[queryItems]

Word	Description
Description	Inquire all purchasable items.
Syntax	public void queryItems(Activity activity, ItemListCallback callback)
Parameters	activity [in] of current application.
Parameter	callback [in] which returns response of API.
Return Value	void

[Response (JSON)]

Attribute	Value	Description
itemSeq	Long	item identifier
itemName	String	item name
marketItemId	String	item ID shown in store console.
currency	String	currency
price	Float	price

[Response Example]

```
[{
    "itemSeq" : 1000208,
    "itemName" : "Test item 01",
    "marketItemId": "item01",
    "price": 1000,
    "currency": "KRW"
}, {
    "itemSeq" : 1000209,
    "itemName" : "Test item 02",
    "marketItemId": "item02",
    "price": 7.99,
    "currency": "USD"
}]
```

public interface InAppPurchase.PurchaseCallback

callback interface for receiving response after payment request

[Method Summary]

Name	Return Value	Parameter
onCallback	void	JSONObject result, InAppPurchaseException exception

[onCallback]

Word	Description
Description	Send API request result.
Syntax	public abstract void onCallback(JSONObject result, InAppPurchaseException exception)
Parameters	result [in] return result code and message.
Parameters	exception [in] return detail error. if null, request is successful.
Return Value	void

public interface InAppPurchase.PurchaseListCallback

callback interface for receiving response after payment request

[Method Summary]

Name	Return Value	Parameter
onCallback	void	JSONArray result, InAppPurchaseException exception

[onCallback]

Word	Description
Description	Send API request result.
Syntax	public abstract void onCallback(JSONArray result, InAppPurchaseException exception)
Parameters	result [in] return result code and message.
Parameters	exception [in] return detail error. if null, request is successful.
Return Value	void

> [Reference]
> 1. Call unsynchronized API from UI Thread (main Thread)
> 2. When calling unsynchronized API, send response result via parameter Callback interface.

public final class InAppPurchases

Entry Point for IAP payment interface

[Field]

Type	Variable	Description
public static final InAppPurchase	InAppPurchases	interface for iap payment

[getSdkVersion]

Word	Description
Description	returns SDK version.
Syntax	public static String getSdkVersion()
Return Value	String SDK의 Version

[getAppId]

Word	Description
Description	returns App ID.
Syntax	public static long getAppId()
Return Value	String App ID registered in SDK

Package: com.toast.android.iap.exception

public class InAppPurchaseException extends Exception

Returns error of api request.

[getErrorCode]

Word	Description
Description	returns error code.
Syntax	public int getErrorCode()
Return Value	int error code

[getMessage]

Word	Description
Description	returns detail error message.
Syntax	public String getMessage()
Return Value	String detail error message