

# REMON Android SDK GUIDE

(SDK Version 1.3.1)

## 문서 개정 내역

변경일	버전	변경 내용	비고
2014.06.30	1.0.0	최초 작성	
2014.09.30	1.1.0	개인정보 항목 변경, 개인정보 이용약관 기능 추가	
2014.12.01	1.2.0	Proguard 추가	
2014.12.16	1.2.0	Android Studio 기준 샘플 및 가이드 추가	
2015.01.29	1.3.0	광고 사이즈 추가 및 변경	
2015.02.06	1.3.1	일부 Permission를 옵션으로 변경	

## < 목 차 >

<b>1. RemoN SDK 소개</b>	<b>4</b>
<b>2. 변경 이력</b>	<b>4</b>
<b>3. RemoN 광고 삽입 방법</b>	<b>4</b>
3.1 RemoN SDK 구성	4
3.1.1 cId 발급 받기	5
3.1.2 라이브러리 Import	5
3.1.3 AndroidManifest.xml 설정	9
3.1.4 광고 요청을 위한 UI 구성 및 설정	11
3.1.5 ProGuard 설정(옵션)	19
<b>4. Class reference</b>	<b>19</b>
4.1 Package kr.co.dnasoft.remonsdk	19
4.1.1 Class Summary	19
4.1.2 Interface Summary	19
4.1.3 Enum Summary	19
4.2 Class AdView	20
4.2.1 Constructor Summary	20
4.2.2 Method Summary	20
4.2.3 Interface AdView.OnAdClickedListener	21
- Method Summary	21
4.2.4 Interface AdView.OnAdFailedListener	21
- Method Summary	21
4.2.5 Interface AdView.OnAdClosedListener	21
- Method Summary	21
4.2.6 Interface AdView.OnAdLoadedListener	21
- Method Summary	21
4.2.7 Interface AdView.OnAdWillLoadListener	22
- Method Summary	22
4.3 Class AdInfo	23
4.3.1 Constructor Summary	23
4.3.2 Method Summary	23
4.4 Class AdInterstitial	23
4.4.1 Constructor Summary	23
4.4.2 Method Summary	23
<b>5. Enum ADErrorCode</b>	<b>24</b>

## 1. RemoN SDK 소개

이 가이드는 Android Application에 모바일 광고를 노출하기 위한 광고 데이터 요청과 처리 방법을 설명 한다.

어플리케이션 운영 정책에 어긋나는 경우 적립금 지급이 거절 될 수 있으니 유의하시기 바랍니다.

### - 광고 종류 및 사이즈

광고 종류	사이즈	지원 버전
Inline	320 * 50	9 이상(2.3)
Square	150 * 150	9 이상(2.3)
Rectangle	300 * 250	9 이상(2.3)
Floating(확장형)	104 * 77 -> 300 * 250	11 이상(3.0)
Interstitial	320 * 480	9 이상(2.3)

## 2. 변경 이력

- Version 1.3.1
  - 일부 Permission를 옵션으로 변경
- Version 1.3.0
  - 광고 사이즈 추가 및 변경
- Version 1.2.0
  - Google Ad Id 사용을 위한 google-play-services 라이브러리 및 사용 코드 추가
  - 갱신 오류 수정
- Version 1.1.0
  - 개인정보 항목 변경
  - 위치 정보 관련 항목 변경
  - 개인정보 이용약관 관련 기능 추가
- Version 1.0.0
  - 최초 작성

### 3. RemoN 광고 삽입 방법

#### 3.1 RemoN SDK 구성

- RemoNSDK.jar : RemoN 광고를 삽입해주는 라이브러리
- RemonTest : Eclipse용 광고 샘플
- RemonTest\_Android\_Studio : Android Studio용 광고 샘플

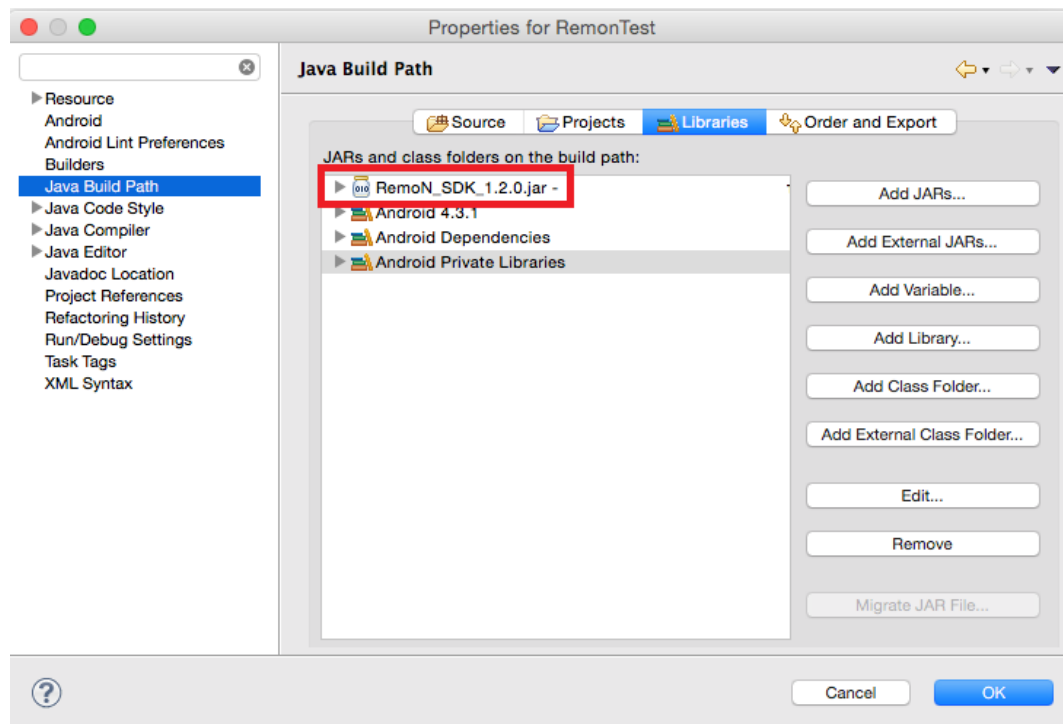
##### 3.1.1 cId 발급 받기

- 실제 광고를 다운로드 받고 수익창출을 위해서 <http://www.realclick.co.kr/product/?sidx=rm>에서 회원 가입후 앱 등록 절차를 거쳐 cId를 발급 받아야 한다.

##### 3.1.2 라이브러리 Import

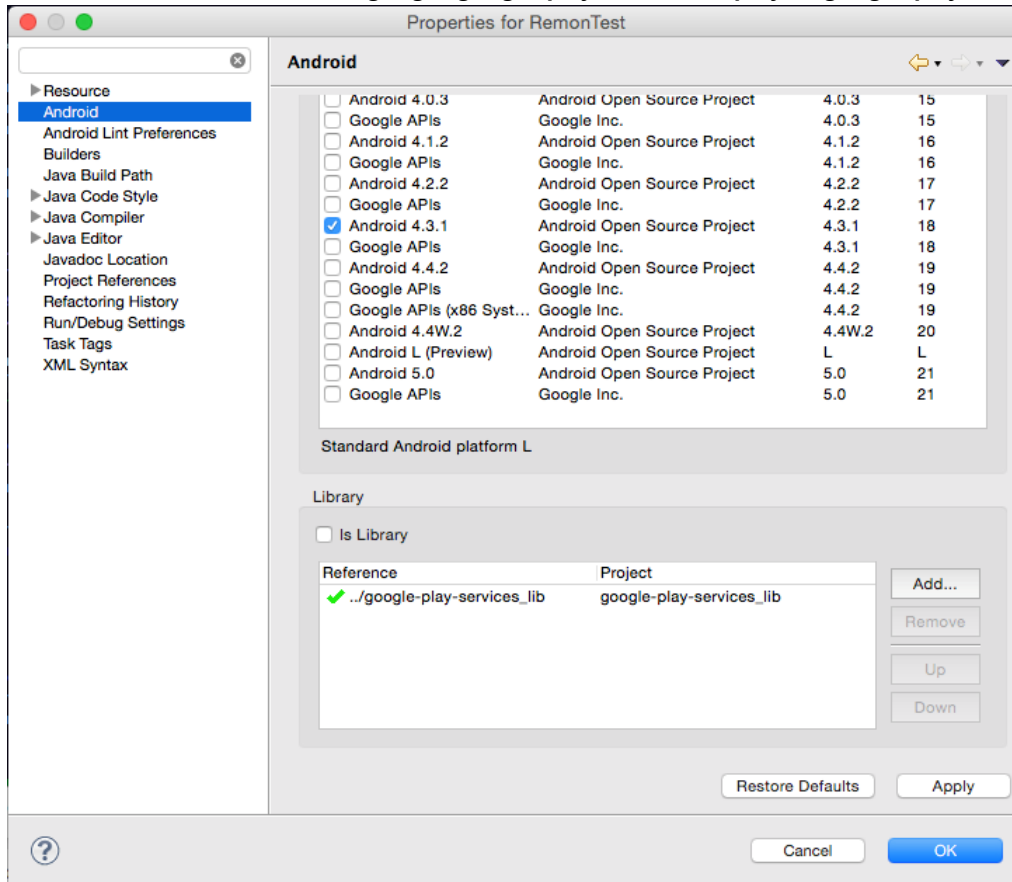
###### 3.1.2.1 ADT를 사용한 경우

- RemoN SDK를 프로젝트 내에 라이브러리로 Import 한다.

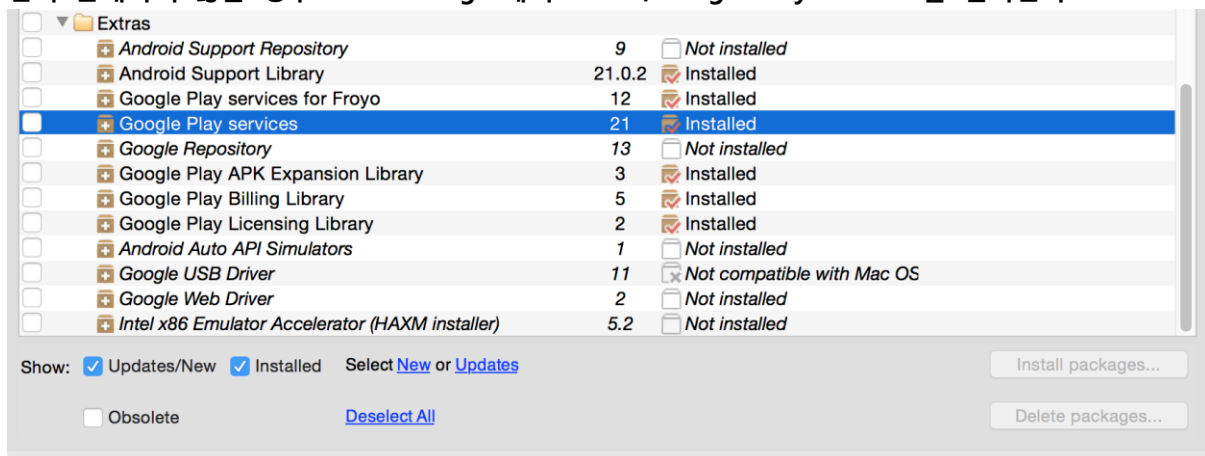


- Google play service 라이브러리 추가 (항상 최신버전을 유지하도록 한다)

(기본 위치 : {SDK Root}/extras/google/google\_play\_services/libproject/google-play-services\_lib)



만약 존재하지 않을 경우 SDK Manager에서 Extras/Google Play Services를 설치한다.

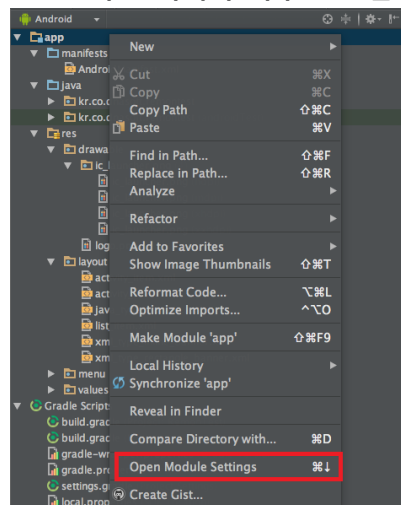


### 3.1.2.2 Android Studio를 사용할 경우

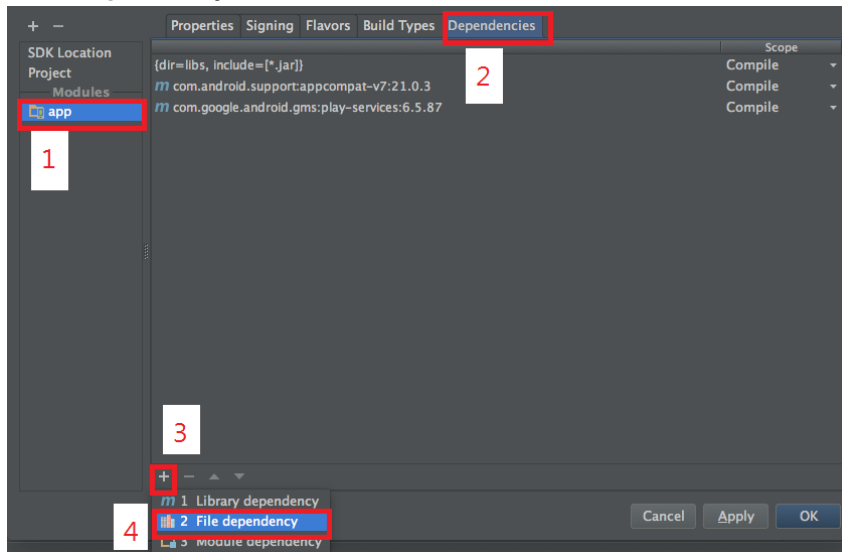
- SDK를 해당 모듈 내의 libs폴더로 복사한다.

▼ app	오늘 오전 11:20
app.iml	오늘 오전 11:15
▶ build	오늘 오전 11:20
build.gradle	오늘 오전 11:14
▼ libs	오늘 오전 11:09
RemoN_SDK_1.2.0.jar	2014년 12월 2일 오전 9:28
proguard-rules.pro	오늘 오전 11:02
▶ src	오늘 오전 11:03
▶ build	오늘 오전 11:21
build.gradle	오늘 오전 11:02
▶ gradle	오늘 오전 11:02
gradle.properties	오늘 오전 11:02
gradlew	오늘 오전 11:02
gradlew.bat	오늘 오전 11:02
local.properties	오늘 오전 11:02
RemonTest.iml	오늘 오전 11:04
settings.gradle	오늘 오전 11:02

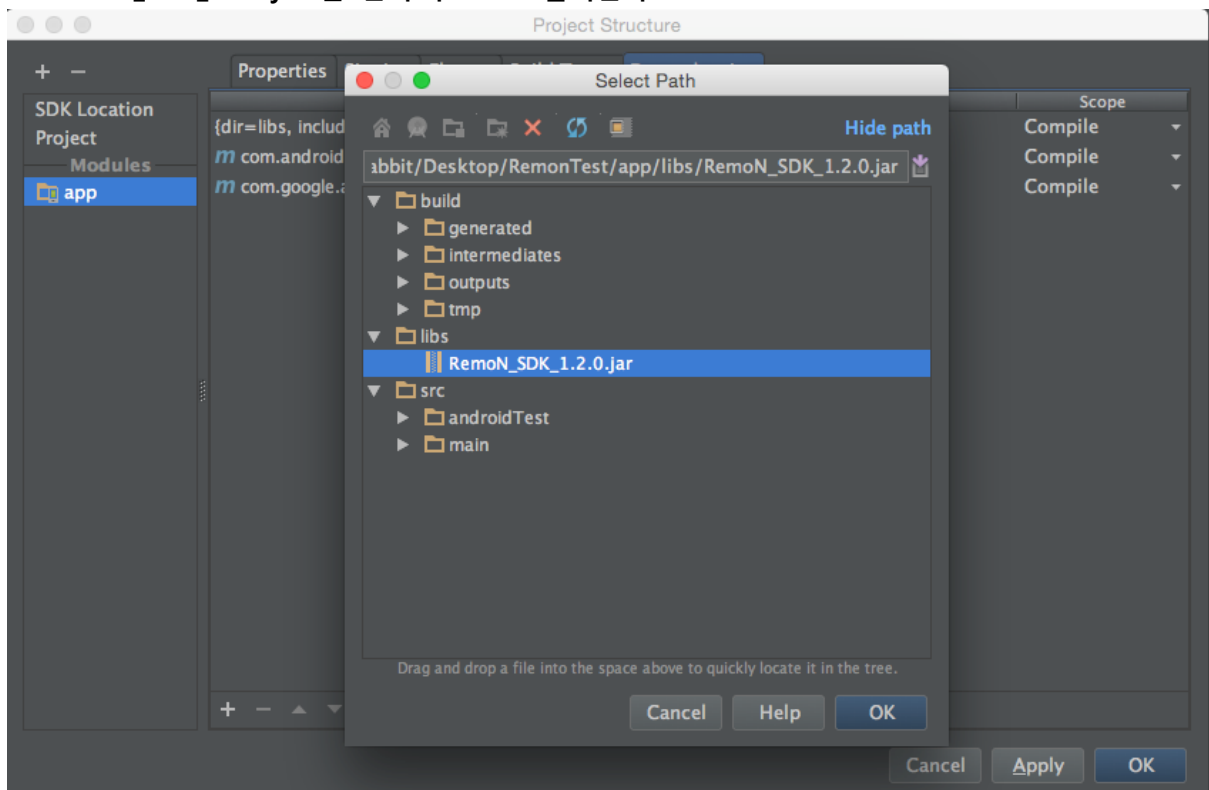
- 프로젝트 위에서 마우스 오른쪽 버튼 클릭 후 “Open Module Settings” 메뉴를 클릭한다..



- 해당 모듈을 선택하고, "Dependencies" 탭을 클릭한다. 이후 화면 하단에 "+" 버튼을 클릭하고 "File Dependency"을 클릭한다.

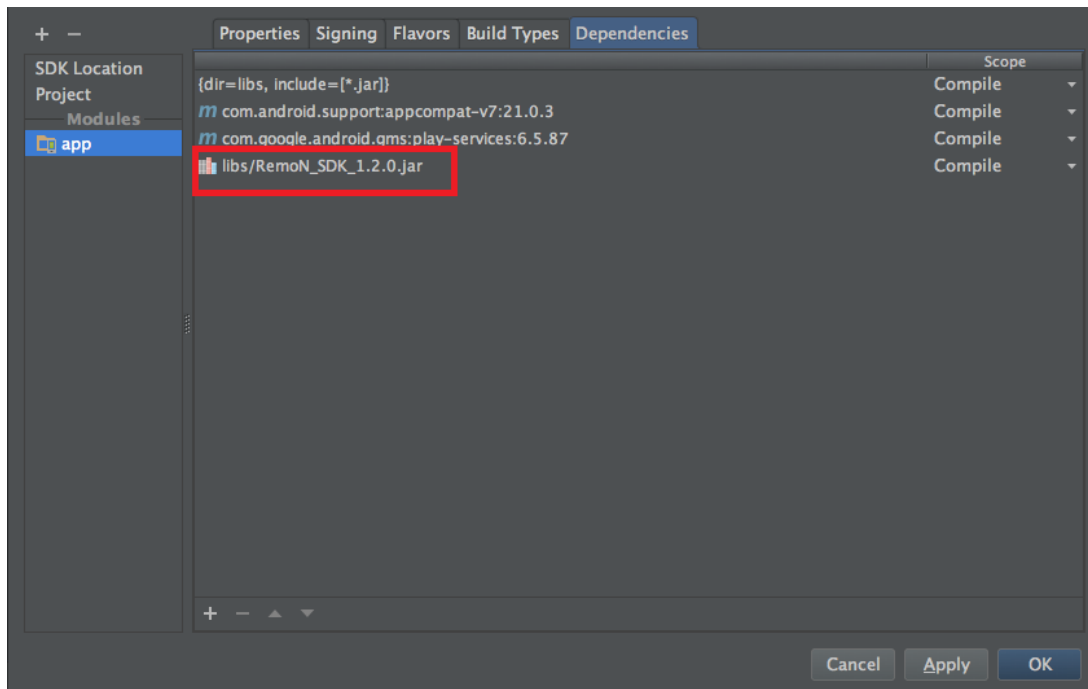


- "RemoN\_SDK\_x.x.x.jar"을 선택 후 "OK" 클릭한다.

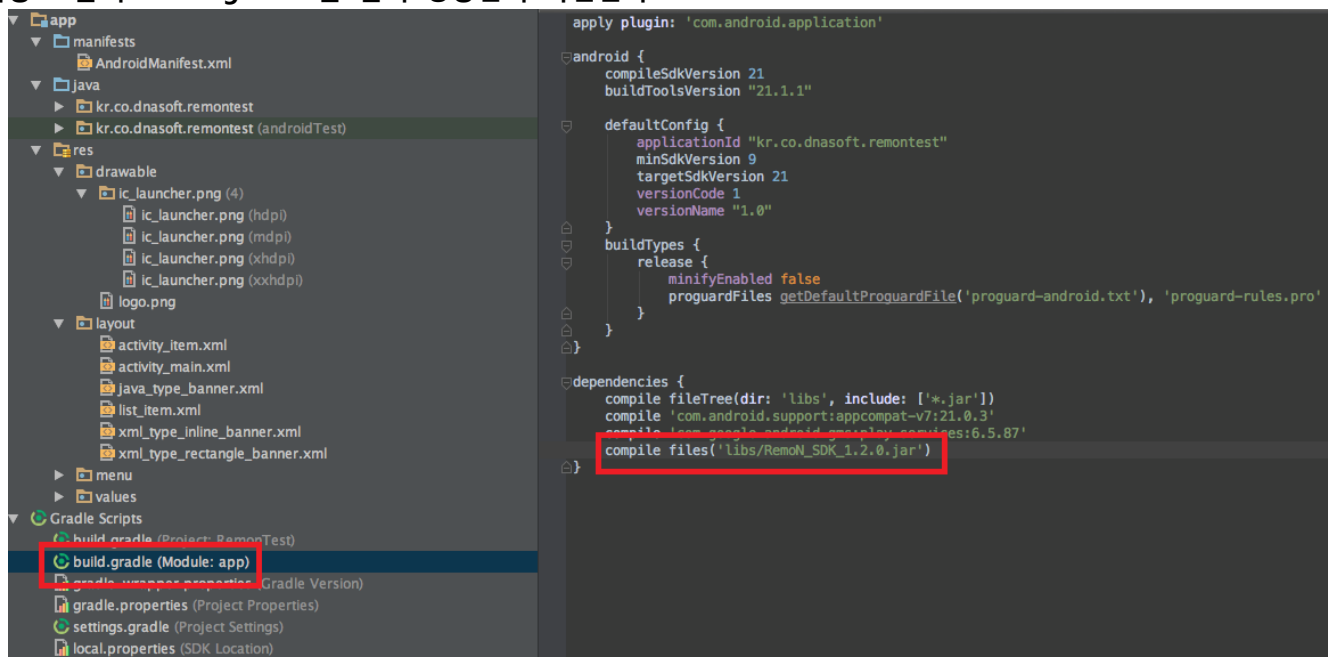




- 아래와 같이 SDK가 추가되었는지 확인 후 "OK"를 클릭한다.



- 해당 모듈의 "build.gradle"을 열어 정상인지 확인한다.



### 3.1.3 AndroidManifest.xml 설정

- 아래 세가지 필수 권한을 설정하여야 하며, 필수 권한 미 설정시 정상적인 광고 노출 되지 않는다.

```
<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" />
```

- 아래 두가지 옵션 권한은 사용자 약관에 동의시 지역 기반 타겟팅에 사용된다. (사용 약관에 동의하였다고 하더라도 아래 권한을 설정하지 않았을 경우 지역 기반 타겟팅이 되지 않는다.)

```
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
```

- 광고를 넣을 Activity에 Activity 재생성 방지를 위해 반드시

`android:configChanges="orientation"`을 설정해준다.

(Project Build Target API Level이 13 이상인 경우에는 `screenSize` 설정 추가 필요)

- Google Ad ID 사용을 위한 meta-data를 반드시 추가한다.

```
<meta-data android:name="com.google.android.gms.version"
            android:value="@integer/google_play_services_version" />
```

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="kr.co.dnasoft.remontest"
    android:versionCode="1"
    android:versionName="1.1.0" >

    <uses-sdk
        android:minSdkVersion="8"
        android:targetSdkVersion="19" />

    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
    <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />

    <application
        android:allowBackup="true"
        android:icon="@drawable/icon"
        android:label="@string/app_name"
        android:theme="@style/AppTheme" >

        <meta-data android:name="com.google.android.gms.version"
            android:value="@integer/google_play_services_version" />

        <!-- Sample MainActivity -->
        <activity
            android:name="kr.co.dnasoft.remontest.SampleActivity"
            android:label="@string/app_name"
            android:configChanges="orientation" >
```

필수 권한 설정

Google Ad Id사용을  
위한 meta-data 추가

```

        <intent-filter>
            <action android:name="android.intent.action.MAIN" />
            <category android:name="android.intent.category.LAUNCHER" />
        </intent-filter>
    </activity>

    <!-- JAVA Inline Banner Type 광고 Sample -->
    <activity
        android:name="kr.co.dnasoft.remontest.JavaTypeInlineBanner"
        android:configChanges="orientation" />

    <!-- XML Inline Banner Type 광고 Sample -->
    <activity
        android:name="kr.co.dnasoft.remontest.XMLTypeInlineBanner"
        android:configChanges="orientation" />

</application>

</manifest>

```

**Project Build Target API Level이 13 이상인 경우에는 `screenSize` 설정 추가 필요**

3.1.4 광고 요청을 위한 UI 구성 및 설정 광고를 넣고자 하는 View 가 들어 있는 Activity 가 생성될 때 AdView 객체를 생성하고 광고 요청을 위해 광고 View에 필요한 리스너와 할당 받은 `clientId`를 설정 한다. (광고별 Parameter 적용 방법은 샘플을 참고하도록 한다)

- Inline Banner 샘플

```

public class JavaTypeInlineBanner extends Activity {

    private final static String TAG = JavaTypeInlineBanner.class.getName();

    private static AdView adView = null;
    private LinearLayout layout = null;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.java_type_banner);

        initRemon();
    }

    private void initRemon() {

        layout = (LinearLayout) findViewById(R.id.ad_layout);

        //RemoN 광고 View 생성 및 설정
        if(adView != null) {

            adView.stopAd();
            adView.setOnAdClickedListener(null);
            adView.setOnAdClosedListener(null);
            adView.setOnAdFailedListener(null);
            adView.setOnAdLoadedListener(null);
            adView.setOnAdWillLoadListener(null);

            layout.removeView(adView);
        }
    }
}

```

```

        adView = null;
    }

    //RemoN 광고 뷰 생성 및 설정
    adView = new AdView(this.getApplicationContext());

    /*=====
    *      Listener(해당 기능이 필요 없으면 설정하지 않아도 된다.)
    =====*/
    //광고 클릭시 실행될 Listener
    adView.setOnAdClickedListener(new OnAdClickClickedListener() {
        @Override
        public void OnClickedAd() {
            Log.i(TAG, "JAVA Type Inline Banner Click ad");
        }
    });

    //광고 실패시 실행될 Listener
    adView.setOnAdFailedListener(new OnAdFailedListener() {
        @Override
        public void OnAdFailed(ADErrorCode arg0, String arg1) {
            Log.w(TAG, "JAVA Type Inline Banner Ad failed : " + arg1);
        }
    });

    //광고가 정상적으로 Load 될때 실행될 Listener
    adView.setOnAdLoadedListener(new OnAdLoadedListener() {
        @Override
        public void OnAdLoaded() {
            Log.i(TAG, "JAVA Type Inline Banner Ad loaded");
        }
    });

    //광고를 호출하는 시점에 실행될 Listener
    adView.setOnAdWillLoadListener(new OnAdWillLoadListener() {
        @Override
        public void OnAdWillLoad() {
            Log.i(TAG, "JAVA Type Inline Banner Ad will load");
        }
    });

    //광고를 종료할때 실행될 Listener
    adView.setOnAdClosedListener(new OnAdClosedListener() {
        @Override
        public void OnAdClosed() {
            Log.i(TAG, "JAVA Type Inline Banner Ad closed");
        }
    });

    //할당 받은cId 설정(필수)
    adView.setcId("MTQxMDg3NDQ1MzY3OTc4");

    //광고 배경색 설정
    //(서버에서 내려오는 배경색이 없는 경우 사용자 지정 색으로 설정)

```

```

adView.setBackgroundColor("#FFEAEA");
//광고 갱신 주기 설정(default : 30초, max : 300초)
adView.setRefreshInterval(30);

//SDK 광고 Test 설정(default : false)
adView.setTestMode(true);

//SDK 광고 요청 상태 Log 설정(default : false)
adView.setSDKLog(true, "JavaTypeInlineBanner");

//광고 활성화 설정 : INVISIBLE, GONE로 설정시 광고 중단(default : VISIBLE)
adView.setVisibility(View.VISIBLE);
layout.addView(adView);

//약관동의 여부 설정
adView.setTerms(true);

//사용자 정보 설정
AdInfo adInfo = new AdInfo();

//사용자 성별
adInfo.setGender("f");

//사용자 연령대
adInfo.setAge(38);
adView.setAdInfo(adInfo);

//광고 요청
adView.requestAd();
}

@Override
protected void onPause() {
    if(adView != null) {
        adView.onPause();
    }
    super.onPause();
}

@Override
protected void onResume() {
    if(adView != null) {
        adView.onResume();
    }
    super.onResume();
}

@Override
protected void onDestroy() {
    if(adView != null) {
        adView.stopAd();
        adView.setOnAdClickedListener(null);
        adView.setOnAdClosedListener(null);
        adView.setOnAdFailedListener(null);
        adView.setOnAdLoadedListener(null);
        adView.setOnAdWillLoadListener(null);
    }
}

```

```

        layout.removeView(adView);
        adView = null;
    }
    super.onDestroy();
}

```

### ● 전면 광고 샘플

```

public class JavaTypeInterstitialBanner extends Activity {

    //전역 광고 객체
    AdInterstitial interstitial = null;

    @Override
    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

        initRemon();
    }

    private void initRemon() {

        //광고 Listener
        AdListener listener = new AdListener();
        listener.OnAdClickClickedListener = new OnAdClickClickedListener() {
            @Override
            public void OnClickedAd() {
                Log.i("JavaTypeInterstitialBanner", "JAVA Type Interstitial Banner Click ad");
            }
        };
        listener.OnAdWillLoadListener = new OnAdWillLoadListener() {

            @Override
            public void OnAdWillLoad() {
                Log.i("JavaTypeInterstitialBanner", "JAVA Type Interstitial Banner Will Load");
            }

        };

        listener.OnAdClickClosedListener = new OnAdClosedListener() {

            @Override
            public void OnAdClosed() {
                Log.i("JavaTypeInterstitialBanner", "JAVA Type Interstitial Banner Closed");
            }

        };

        listener.OnAdLoadedListener = new OnAdLoadedListener() {

            @Override
            public void OnAdLoaded() {
                Log.i("JavaTypeInterstitialBanner", "JAVA Type Interstitial Banner Loaded");
            }

        };

        listener.OnAdFailedListener = new OnAdFailedListener() {

            @Override

```

```

        public void OnAdFailed(ADErrorCode arg0, String arg1) {
            Log.i("JavaTypeInterstitialBanner", "JAVA Type Interstitial Banner Failed");
            Log.i("JavaTypeInterstitialBanner", "Error Code : " + arg0 + ", Reason : " + arg1);
        }
    };

    //초기화 (인자로 Activity와 AdListener 객체 전달)
    interstitial = new AdInterstitial(this, listener);

    //발급 받은 client Id 설정(Mandatory)
    interstitial.setClientId("MTQyODkzNTMxNTcyNjE4");

    //자동 닫힘 설정(Optional, Default : false)
    interstitial.setTestMode(true);

    //자동 닫힘 설정(Optional, Default : false)
    interstitial.setLogMode(true);

    //광고 배경색 설정(서버에서 내려오는 배경색이 없는 경우 사용자 지정 색으로 설정)
    interstitial.setUserBGColor("#cef279");

    //자동 닫힘 설정(Optional, Default : false)
    interstitial.setUseAutoClose(true);

    //자동 닫힘 시간 설정(Optional, Default : 10, 범위 : 0 추가 10 이하)
    interstitial.setAutoCloseTime(30);

    //약관동의 여부 설정(Optional, Default : false)
    interstitial.setTerms(true);

    //사용자 정보 설정(Optional)
    AdInfo adInfo = new AdInfo();

    //사용자 성별(Optional, m : 남성, f : 여성)
    adInfo.setGender("f");

    //사용자 연령대(Optional)
    adInfo.setAge(38);
    interstitial.setAdInfo(adInfo);

    //광고 요청
    interstitial.requestAd();
}

@Override
protected void onPause() {

    super.onPause();
}

@Override
protected void onResume() {

    super.onResume();
}

```

```

@Override
protected void onDestroy() {

    //종료시 광고 객체 해제

    if (interstitial != null) {
        interstitial.removeAd();
        interstitial = null;
    }

    super.onDestroy();
}
}

```

- 약관 동의 방식

약관동의 여부를 사용자가 설정할 수 있다.

약관동의 이용시 아래의 명시된 Activity를 추가해야 한다

```

<activity
    android:name="kr.co.dnasoft.remonsdk.AgreeActivity"
    android:screenOrientation="portrait"
    android:configChanges="orientation" >
</activity>

```

약관동의를 사용하기 위해서는 위 부분을 반드시 추가 해야 한다.

```

public class AgreeSampleActivity extends Activity {

    private final static String TAG =
    AgreeSampleActivity.class.getName();

    private static AdView adView = null;
    private LinearLayout layout = null;

    /** 사용자에게 노출 하여 사용자가 선택한 값 저장후 사용 */
    private boolean agreeFlag = false;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.java_type_banner);
        layout = (LinearLayout) findViewById(R.id.ad_layout);

        initRemon();
        Agreement();
    }

    private void initRemon() {

        //RemoN 광고 View 생성 및 설정
        if(adView != null) {

            adView.stopAd();

            adView.setOnAdClickedListener(null);
            adView.setOnAdClosedListener(null);
            adView.setOnAdFailedListener(null);
            adView.setOnAdLoadedListener(null);
            adView.setOnAdWillLoadListener(null);

```



```

        layout.removeView(adView);
        adView = null;
    }

    //Remon 광고 뷰 생성 및 설정
    adView = new AdView(this);

    /*=====
                               Listener
    =====*/

    //광고 클릭시 실행될 Listener
    adView.setOnAdClickedListener(new OnAdClickClickedListener() {
        @Override
        public void OnClickedAd() {
            Log.i(TAG, " AgreeSampleActivity Inline Banner Click ad");
        }
    });

    //광고 실패시 실행될 Listener
    adView.setOnAdFailedListener(new OnAdFailedListener() {

        @Override
        public void OnAdFailed(ADErrorCode arg0, String arg1) {
            Log.w(TAG, " AgreeSampleActivity Inline Banner Ad failed : " + arg1);
        }
    });

    //광고가 정상적으로 Load 될때 실행될 Listener
    adView.setOnAdLoadedListener(new OnAdLoadedListener() {

        @Override
        public void OnAdLoaded() {
            Log.i(TAG, " AgreeSampleActivity Inline Banner Ad loaded");
        }
    });

    //광고를 호출하는 시점에 실행될 Listener
    adView.setOnAdWillLoadListener(new OnAdWillLoadListener() {

        @Override
        public void OnAdWillLoad() {
            Log.i(TAG, "AgreeSampleActivity Inline Banner Ad will load");
        }
    });

    //광고를 종료할때 실행될 Listener
    adView.setOnAdClosedListener(new OnAdClosedListener() {

        @Override
        public void OnAdClosed() {
            Log.i(TAG, " AgreeSampleActivity Inline Banner Ad closed");
        }
    });

    //할당 받은client Id 설정(필수)
    adView.setClientId("MTQxMDg3NDQ1MzY3OTc4");

    //광고 배경색 설정(서버에서 내려오는 배경색이 없는 경우 사용자 지정 색으로 설정)
    adView.setBackgroundColor("#FFEAEAEA");

```

```

//광고 갱신 주기 설정(default : 30초, max : 300초)
adView.setRefreshInterval(30);

//SDK 광고 Test 설정(default : false)
adView.setTestMode(true);

//SDK 광고 요청 상태 Log 설정(default : false)
adView.setSDKLog(true, " AgreeSampleActivity");

//광고 활성화 설정 : INVISIBLE, GONE로 설정시 광고 중단(default : VISIBLE)
adView.setVisibility(View.VISIBLE);
layout.addView(adView);

//사용자 정보 설정
AdInfo adInfo = new AdInfo();
adInfo.setGender("F");
adInfo.setAge(38);
adView.setAdInfo(adInfo);
}

private void Agreement() {
    Intent intent = new Intent(AgreeSampleActivity.this, AgreeActivity.class);
    startActivityForResult(intent, 1);
    intent = null;
}

@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
    super.onActivityResult(requestCode, resultCode, data);

    if(requestCode == 1 || resultCode == RESULT_OK){
        if(data.getStringExtra("AgreeFlag").equals("Y")){//use data

            if(adView != null) {
                agreeFlag = true;
            }

        } else {
            if(adView != null) {
                agreeFlag = false;
            }

            //약관동의 여부 설정
            adView.setTerms(agreeFlag);
        }
    }

    //광고 요청
    adView.requestAd();
}

@Override
protected void onPause() {
    if(adView != null) {
        adView.onPause();
    }
    super.onPause();
}
}

```

```

@Override
protected void onResume() {
    if(adView != null) {
        adView.onResume();
    }
    super.onResume();
}

@Override
protected void onDestroy() {
    if(adView != null) {
        adView.stopAd();
        adView.setOnAdClickedListener(null);
        adView.setOnAdClosedListener(null);
        adView.setOnAdFailedListener(null);
        adView.setOnAdLoadedListener(null);
        adView.setOnAdWillLoadListener(null);
        layout.removeView(adView);
        adView = null;
    }
    layout = null;
    super.onDestroy();
}
}

```

### 3.1.5 ProGuard 설정(옵션)

Proguard 설정시 아래 사항을 추가 한다.

```

-dontwarn android.support.**,com.google.android.gms.**
-keep class kr.co.dnasoft.remonsdk.** {*;}

```

## 4. Class reference

### 4.1 Package kr.co.dnasoft.remonsdk

#### 4.1.1 Class Summary

<a href="#">AdView</a>	광고가 보여지는 View의 Object에 해당하는 클래스 이다.
------------------------	-------------------------------------

#### 4.1.2 Interface Summary

<a href="#">AdView.OnAdClickedListener</a>	광고를 클릭시 실행될 Listener
<a href="#">AdView.OnAdFailedListener</a>	광고 다운로드 실패시 실행될 Listener
<a href="#">AdView.OnAdClosedListener</a>	광고를 종료할때 실행될 Listener
<a href="#">AdView.OnAdLoadedListener</a>	광고가 정상적으로 Load 될때 실행될 Listener
<a href="#">AdView.OnAdWillLoadListener</a>	광고를 호출하는 시점에 실행될 Listener
<a href="#">AdView.OnAdWillLoadListener</a>	광고를 호출하는 시점에 실행될 Listener

#### 4.1.3 Enum Summary

<b>ADState</b>	광고 상태 표시
<b>ADErrorCode</b>	광고 노출과 관련된 Error Type을 지정한다.

### 4.2 Class AdView

#### 4.2.1 Constructor Summary

<b>AdView</b> (Context context)
Java 기반 광고 객체 생성
<b>AdView</b> (Context context, AttributeSet attrs)
XML 기반 광고 객체 생성
<b>AdView</b> (Context context, AttributeSet attrs, int defStyle)
XML 기반 광고 객체 생성

#### 4.2.2 Method Summary

<b>void</b>	<b>onPause()</b> 광고 갱신을 중지
<b>void</b>	<b>onResume()</b> 광고 갱신을 일시적으로 재개한다.
<b>void</b>	<b>requestAd()</b> 광고 요청
<b>void</b>	<b>stopAd()</b> 광고 중지 이 함수가 호출된 이후에는 다시 광고 갱신을 재개할 수 없다.
<b>void</b>	<b>setBackgroundColor</b> (String userBGColor) Server에서 내려오는 배경색이 없을 경우 사용자가 지정한 배경색을 설정한다.
<b>String</b>	<b>getBackgroundColor()</b> 지정한 사용자 배경색을 반환한다.
<b>String</b>	<b>setId</b> (String clientId) 사용자 cId를 지정한다.
<b>String</b>	<b>getId()</b> 지정한 cId를 반환한다.
<b>void</b>	<b>setTestMode</b> (Boolean value) 테스트 모드 여부를 설정한다.
<b>boolean</b>	<b>getTestMode()</b> 지정한 테스트 모드를 반환한다.
<b>void</b>	<b>setRefreshInterval</b> (int refreshInterval) 광고 갱신 주기를 설정한다.

<b>Int</b>	<b>getRefreshInterval()</b> 지정한 광고 갱신 주기를 반환한다.
<b>void</b>	<b>setVisibility(int visibility)</b> 광고 View의 활성화 여부를 설정한다.
<b>int</b>	<b>getVisibility()</b> 지정한 광고 View의 활성화 여부를 반환한다.
<b>void</b>	<b>setSDKLog(Boolean isSDKLog, String tag)</b> SDK 진행상황 Log 출력을 설정한다.
<b>void</b>	<b>setTerms(Boolean isTerms)</b> 개인정보 약관 동의 여부 설정한다.
<b>boolean</b>	<b>isTerms()</b> 개인정보 약관 동의 여부 반환한다.
<b>void</b>	<b>setOnAdClickedListener(AdView.OnAdClickedListener listener)</b> 광고를 클릭시 실행될 Listener
<b>void</b>	<b>setOnAdFailedListener(AdView.OnAdFailedListener listener)</b> 광고 다운로드 실패시 실행될 Listener
<b>void</b>	<b>setOnAdClosedListener(AdView.OnAdClosedListener listener)</b> 광고를 종료할때 실행될 Listener
<b>void</b>	<b>setOnAdLoadedListener(AdView.OnAdLoadedListener listener)</b> 광고가 정상적으로 Load 될때 실행될 Listener
<b>void</b>	<b>setOnAdWillLoadListener(AdView.OnAdWillLoadListener listener)</b> 광고를 호출하는 시점에 실행될 Listener

#### 4.2.3 Interface AdView.OnAdClickedListener

##### - Method Summary

<b>void</b>	<b>OnClickedAd()</b> 광고를 클릭하면 실행된다.
-------------	--

#### 4.2.4 Interface AdView.OnAdFailedListener

##### - Method Summary

<b>void</b>	<b>OnAdFailed(ADErrorCode error, String message)</b> 광고 다운로드 실패시 실행된다.
-------------	---

#### 4.2.5 Interface AdView.OnAdClosedListener

##### - Method Summary

<b>void</b>	<b>OnAdClosed()</b> 광고가 닫힌 경우 실행된다.
-------------	--

#### 4.2.6 Interface AdView.OnAdLoadedListener

#### - Method Summary

void	<b>OnAdLoaded()</b> 광고가 정상적으로 Load 될때 실행된다.
------	--

#### 4.2.7 Interface AdView.OnAdWillLoadListener

##### - Method Summary

void	<b>OnAdWillLoad()</b> 광고를 호출하면 실행된다.
------	---

#### 4.3 Class AdInfo

##### 4.3.1 Constructor Summary

<b>AdInfo()</b> 광고 추가 정보 객체 생성
-----------------------------------

##### 4.3.2 Method Summary

void	<b>setGender</b> (String gender) 앱 사용자의 성별 정보를 설정한다.  앱 내부에서 사용자의 성별 정보를 가지는 경우, gender 속성에 할당하여 타겟팅 광고를 수신 받을 수 있다. 다음과 같은 포맷의 문자열만 할당 가능하며, 유효하지 않은 포맷으로 할당한 경우 무시된다. - 남성 : "M" - 여성 : "F"
String	<b>getGender</b> () 설정된 사용자의 성별을 반환한다.
Void	<b>setAge</b> (int age) 앱 사용자의 나이 정보를 설정한다.  앱 내부에서 사용자의 나이 정보를 가지는 경우 age 속성에 할당하여 타겟팅 광고를 수신 받을 수 있다. 숫자 형식만 할당 가능 하며, 유효하지 않은 경우 무시 된다.
int	<b>getAge</b> () 설정된 사용자의 나이가 있다면 반환한다.

#### 4.4 Class AdInterstitial

##### 4.4.1 Constructor Summary

<b>AdInterstitial</b> (Activity activity, AdListener listener) 전면 광고 객체 생성
---

#### 4.4.2 Method Summary

<b>void</b>	<b>onPause()</b> 광고 갱신을 중지
<b>void</b>	<b>onResume()</b> 광고 갱신을 일시적으로 재개한다.
<b>void</b>	<b>requestAd()</b> 광고 요청
<b>void</b>	<b>stopAd()</b> 광고 중지 이 함수가 호출된 이후에는 다시 광고 갱신을 재개할 수 없다.
<b>void</b>	<b>setBackgroundColor</b> (String userBGColor) Server에서 내려오는 배경색이 없을 경우 사용자가 지정한 배경색을 설정한다.
<b>String</b>	<b>getBackGroundColor()</b> 지정한 사용자 배경색을 반환한다.
<b>String</b>	<b>setId</b> (String clientId) 사용자 cId를 지정한다.
<b>String</b>	<b>getId()</b> 지정한 cId를 반환한다.
<b>void</b>	<b>setTestMode</b> (Boolean value) 테스트 모드 여부를 설정한다. (Default : false)
<b>boolean</b>	<b>getTestMode()</b> 지정한 테스트 모드를 반환한다.
<b>void</b>	<b>setUseAutoClose</b> ((boolean useAutoClose) 광고의 자동 닫힘 여부를 설정한다. (Default : false)
<b>boolean</b>	<b>getUseAutoClose()</b> 광고의 자동 닫힘 여부를 반환한다.
<b>void</b>	<b>seAutoCloseTime</b> (int autoCloseTime) 광고의 자동 닫힘 시간을 설정한다. (Default : 3초, 범위 : 0 초과 10 이하)
<b>boolean</b>	<b>getAutoCloseTime()</b> 광고의 자동 닫힘 시간을 반환한다.
<b>void</b>	<b>setSDKLog</b> (Boolean isSDKLog, String tag) SDK 진행상황 Log 출력을 설정한다.
<b>void</b>	<b>setTerms</b> (Boolean isTerms) 개인정보 약관 동의 여부 설정한다. (Default : false)
<b>boolean</b>	<b>isTerms()</b> 개인정보 약관 동의 여부 반환한다.
<b>void</b>	<b>AdListener.OnAdClickedListener</b>

	광고를 클릭시 실행될 Listener
void	<b>AdListener.OnAdFailedListener</b> 광고 다운로드 실패시 실행될 Listener
void	<b>AdListener.OnAdClosedListener</b> 광고를 종료할때 실행될 Listener
void	<b>AdListener.OnAdLoadedListener</b> 광고가 정상적으로 Load 될때 실행될 Listener
void	<b>AdListener.OnAdWillLoadListener</b> 광고를 호출하는 시점에 실행될 Listener

## 5. Enum ADErrorCode

광고 요청 실패시, Adview.OnAdFailedListener.OnAdFailed() 메서드 첫 번째 인자로 전달 된다.  
에러에 대한 대응을 하고 싶다면, AdView.OnAdFailedListener를 구현해 Adview 객체의  
setOnAdFailedListener() 메서드로 설정 하면 된다.

자세한 에러 코드에 대한 설명은 아래와 같다.

<b>NO_AD</b>	광고가 없는 경우 발생
<b>PERMISSION_DENIED_ERROR</b>	필수 권한을 추가 하지 않은 경우 발생. 이경우 AndroidManifest.xml 에 필수 권한을 설정해 주면 된다.
<b>ACTIVITIES_IS_NOT_VAILD</b>	Activity가 유효하지 않은 경우 발생
<b>INVALID_AD_LAYOUT</b>	광고 영역이 없는 경우 발생
<b>INVALID_AD_LAYOUT_SMAILL</b>	광고 영역이 기준보다 경우 발생
<b>INVALID_CLINET_ID</b>	cid가 유효하지 않은 경우 발생
<b>INVALID_PACKAGENAME</b>	패키지명이 유효하지 않은 경우 발생
<b>INVALID_AD_TYPE</b>	광고 타입이 유효하지 않은 경우 발생
<b>INVALID_AD_URL</b>	광고 Uri이 유효하지 않는 경우 발생
<b>NETWORK_ERROR</b>	네트워크 오류
<b>CONNECTION_ERROR</b>	연결 오류
<b>LOAD_AD_ERROR</b>	광고 로딩 오류
<b>SHOW_AD_ERROR</b>	광고 노출 오류