

AdAppter AndroidSDK инструкция по интеграции, вер.4.1.6

Данный документ детально описывает, как интегрировать библиотеку AdAppter Android SDK 4.1.6 в ваш Android проект на примере демонстрационного приложения (Demo app). Данное SDK позволяет отображать как обычную баннерную рекламу (Banner Ads) так и полноэкранную ричмедиа рекламу (Rich Media Interstitials), в т.ч. и видеорекламу.

Шаг 1: Настройте ваше приложение в личном кабинете

- Войдите в ваш личный кабинет по адресу <http://adappter.mobi/login>
- Создайте новое приложение (Добавить площадку) и заполните необходимые поля
- После создания приложения на вкладке Код Размещения вы увидите идентификатор вашего приложения (Publisher ID)

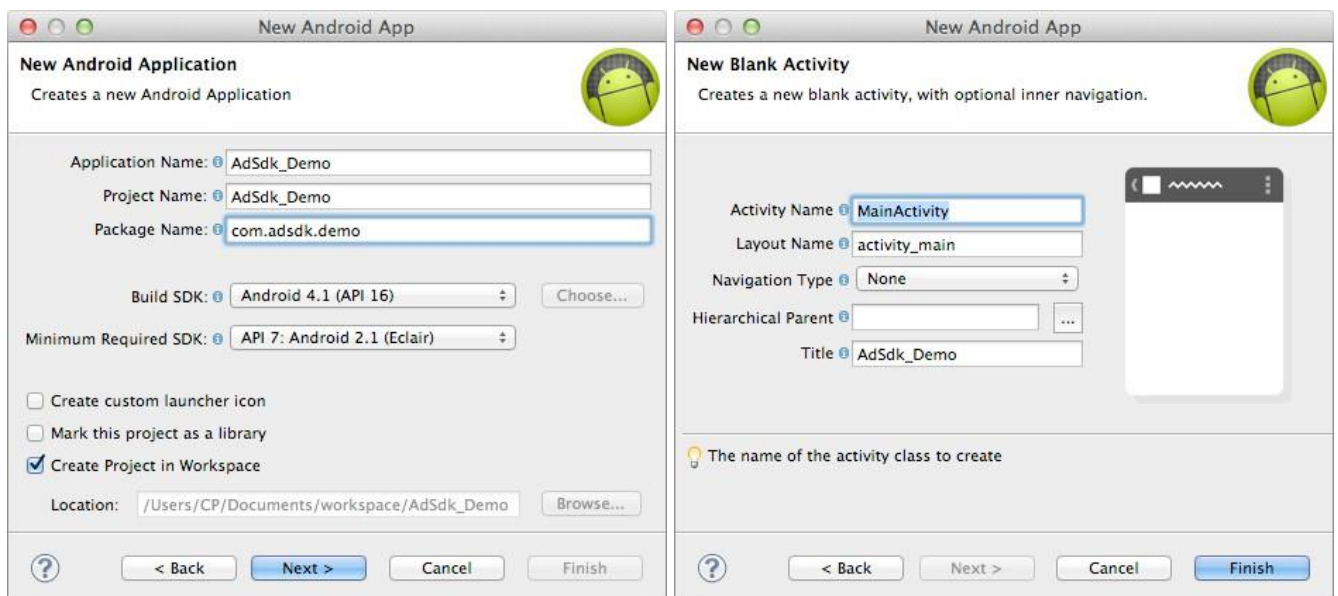
Шаг 2: Скачайте библиотеку SDK

Скачиваемый архив содержит следующие файлы:

- PDF документация "AdAppter Android SDK Setup Guide 4.1.6.pdf" (этот документ)
- AdAppter SDK фреймворк: adappter-android-sdk.jar
- Демо приложение "AdSdk_Demo" (Eclipse 4.2.2, Android SDK Rev. 21.1, SDK Platform Android 4.2.2 Rev. 2 (API 17))

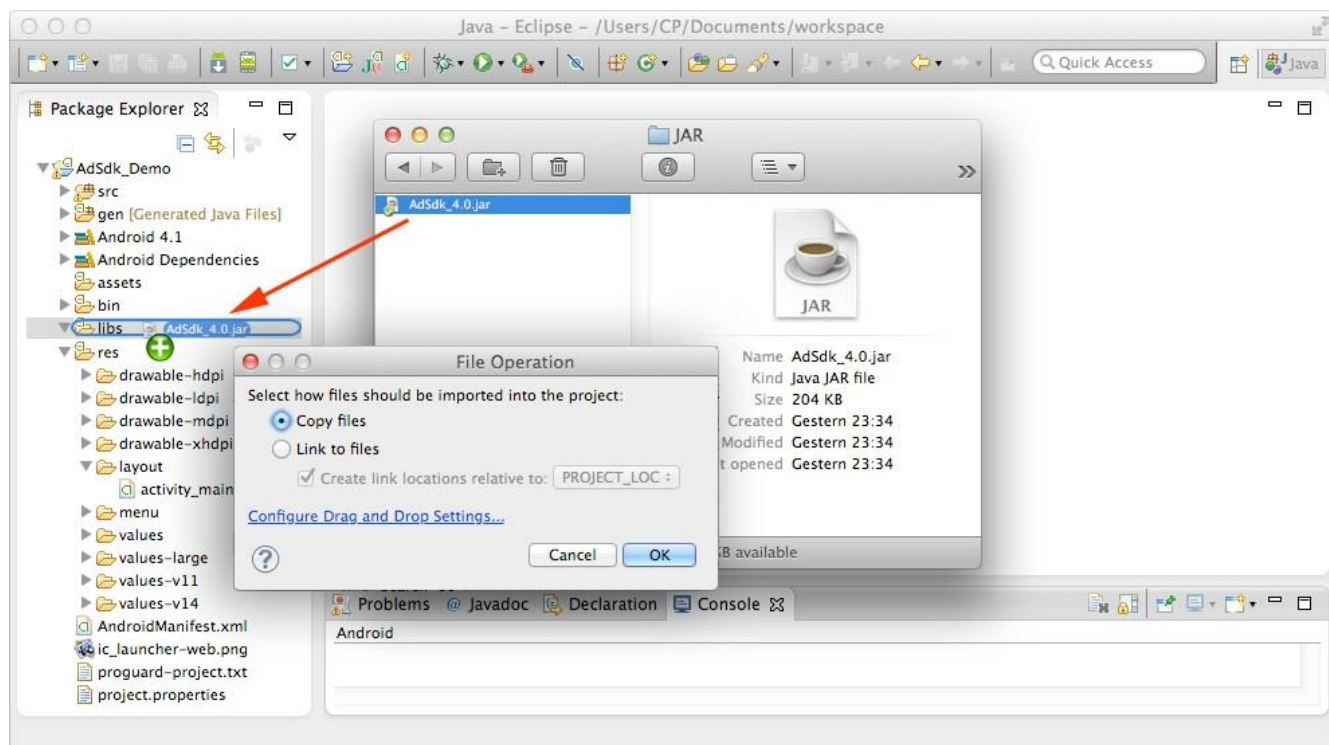
Шаг 3: Создайте новый Android Project в Eclipse

Скачайте и откройте последнюю версию Eclipse. Обновите Android SDK Tools в Android SDK Manager. Теперь откройте мастер "Android Application Project" и заполните все необходимые поля как указано ниже. Установите minimum SDK = API 8 (Androids v2.2).



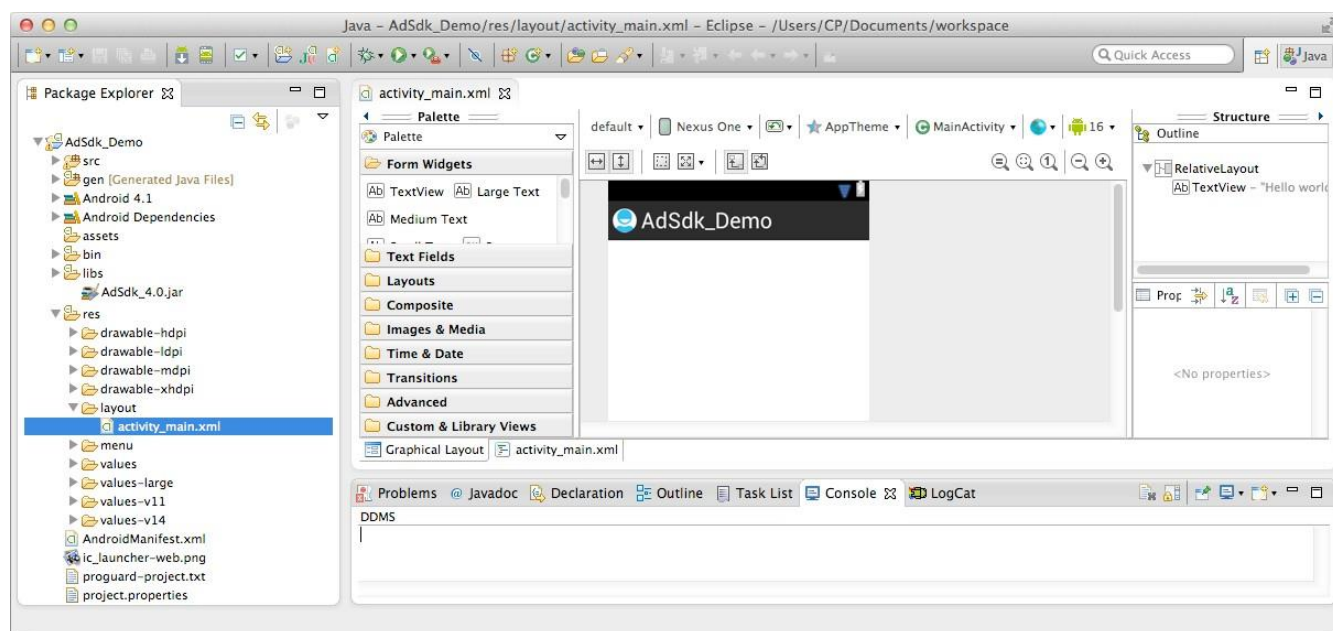
Шаг 4: Добавьте SDK библиотеку в папку "libs" вашего проекта

Если у вас нет папки "libs", создайте ее путем нажатия правой кнопки на корневой папке проекта и выберите "New>Folder". Скопируйте adapter-android-sdk.jar в вашу паку "libs" folder и укажите "Copy files\Скопировать файлы" как показано ниже.



Внимание: если вы используете SDK Rev 16 или более ранний, вам необходимо дополнительно нажать правую кнопку мыши на Проекте и выбрать пункт Свойства (Properties), перейти в раздел "Java Build Path", выбрать "Add JARs...", найти вашу папку с библиотекой и выбрать файл adapter-android-sdk.jar. Но мы настоятельно рекомендуем вам обновить среду разработки до последней версии!

Убедитесь, что папка называется "libs". Вот как будет выглядеть ваш проект с SDK Rev 17 или более свежим.



Для отображения баннерной рекламы (Banner Ads) перейдите к Шагу 5.

Для отображения видеорекламы (vAds), или высокооплачиваемой (интерактивной) полноэкранной, перейдите к Шагу 6.

Шаг 5: Запрос и отображение баннерной рекламы (Banner Ads)

Откройте `activity_main.xml` и добавьте баннер без добавления дополнительного кода как показано в примере.

Вы можете создать кнопку для вызова "onClickShowBanner" (скопируйте `activity_main.xml` из `AdSdk_Demo`)

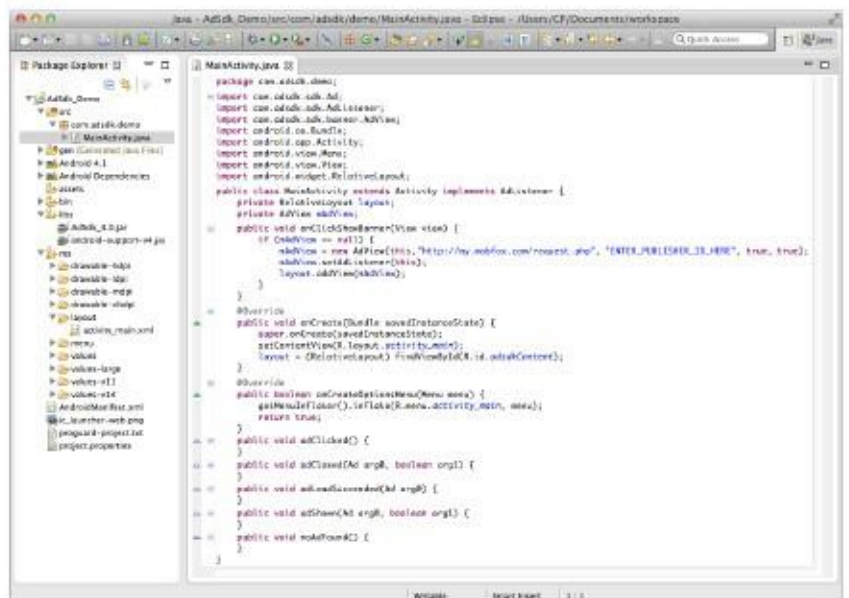
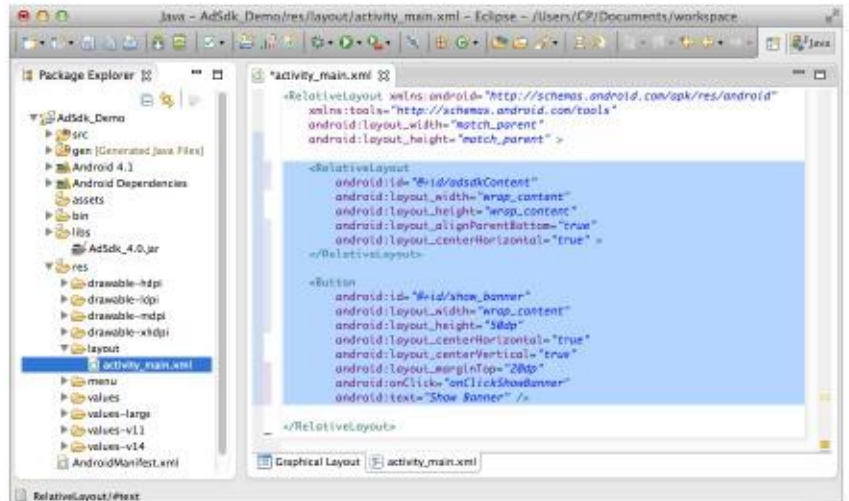
или вместо кнопки можете отобразить баннер напрямую из `MainActivity.java`:

```
if (mAdView == null) {  
    mAdView = new AdView(this, "http://show.adappter.mobi/request.php", "ENTER_PUBLISHER_ID_HERE", true, true);  
    mAdView.setAdListener(this);  
    layout.addView(mAdView);  
}
```

Добавьте теперь немного кода в вашу Activity (скопируйте из `MainActivity.java` (см. `AdSdk_Demo ZIP`)).

Это проинициализирует экземпляр `View Ad` с вашим `Publisher ID`, и установит обработчик нажатия кнопки для запроса рекламы.

Обработчики `View Ad` будут обрабатывать события, такие как например `adLoadSucceeded`,...



Добавьте этот код для баннерной рекламы в Manifest (скопируйте из `AndroidManifest.xml`):

Оба разрешения (`READ_PHONE_STATE` и `LOCATION Permissions`) опциональны:

```
<uses-permission android:name="android.permission.INTERNET" />  
<uses-permission android:name="android.permission.READ_PHONE_STATE" />  
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />  
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />  
<activity  
    android:name="com.adappter.sdk.banner.InAppWebView"  
    android:configChanges="keyboard|keyboardHidden|orientation|screenLayout|  
        uiMode|screenSize|smallestScreenSize" />  
<activity  
    android:name="com.adappter.sdk.mraid.MraidActivity"  
    android:configChanges="keyboard|keyboardHidden|orientation|screenLayout|  
        uiMode|screenSize|smallestScreenSize" />
```

И следующий код в вашу активити с рекламой (см. `MainActivity` в примере `AdSdk_Demo ZIP`):

```
android:configChanges="keyboard|keyboardHidden|orientation|screenLayout|  
    uiMode|screenSize|smallestScreenSize"
```

Если вы не хотите добавлять видео и полноэкранную премиум рекламу перейдите к Шагу 7.

Шаг 6: Запрос и отображение видеорекламы (vAds) и высокооплачиваемой рекламы полноэкранного формата.

Откройте activity_main.xml и создайте кнопку которая будет при нажатии вызывать метод "onClickShowVideoInterstitial" (скопируйте из activity_main.xml)

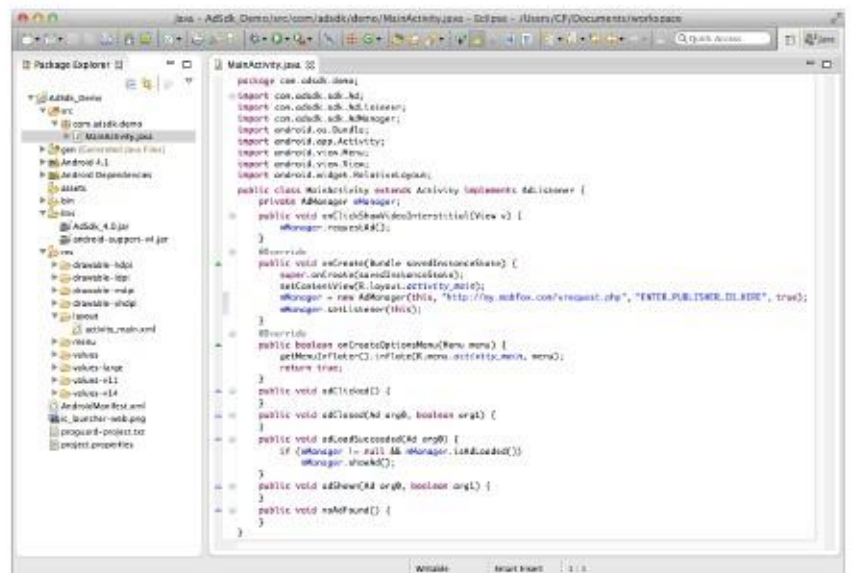
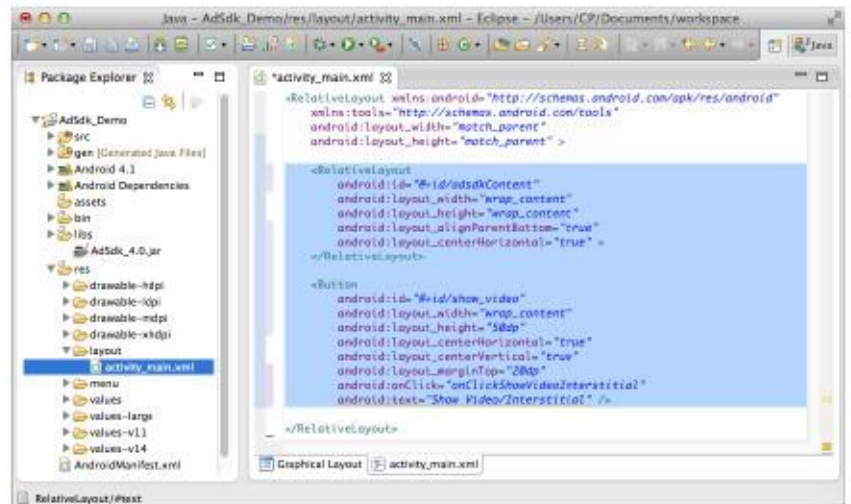
Или вместо этого отображайте vAd напрямую из MainActivity.java:

```
mManager = new AdManager(this,"http://show.adappter.mobi/request.php","ENTER_PUBLISHER_ID_HERE",true);
mManager.setListener(this);
mManager.requestAd();
```

Добавьте теперь немного кода в вашу Activity (скопируйте из MainActivity.java (см. AdSdk_Demo ZIP)).

AdManager проинициализированный с вашим Publisher ID запросит рекламу.

Обработчики View Ad будут обрабатывать события, такие как например adLoadSucceeded,...



Добавьте данные строки в Manifest для vAd рекламы (скопируйте из AndroidManifest.xml):

Both READ_PHONE_STATE and LOCATION Permissions are OPTIONAL

```
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.READ_PHONE_STATE" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
<activity
    android:name="com.adappter.sdk.video.RichMediaActivity"
    android:configChanges="keyboard|keyboardHidden|orientation|screenLayout|
        uiMode|screenSize|smallestScreenSize"
    android:hardwareAccelerated="false" />
```

Опционально для vAds рекламы вы можете создать папку "anim" внутри папки "res". Это позволит использовать анимированные переходы для vAds. Распакуйте animations.zip, и перенесите их в папку "anim".

Step 7: Все готово!

Вы увидите тестовую рекламу. После того как вы готовы получать рекламу залогиньтесь в свой личный кабинет по адресу <http://adappter.mobi/login> и поставьте приложение на модерацию.

Мы рассмотрим и активируем ваше приложение в течение 24х часов.

Внимание: всегда очищайте heap перед показом Ads и восстанавливайте ориентацию вашего приложения после того как реклама показана.

Publisher IDs для тестирования:

ag04mgh1hyio - для 320x50

afnd0v0o445c - для Interstitial

Объяснение кода в Manifest

Internet Access Permission (обязательное, требуется для работы SDK)

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

Read Phone State Permission (опциональное)

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

Network State Permission (рекомендуется для vAds чтобы определить тип соединения с интернетом)

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

Location Permissions (для более точного таргетирования пользователей, опционально)

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

Описание методов

Аргумент конструктора баннерной рекламы "Location" позволяет SDK использовать текущее положение пользователя для более точного таргетинга, а "animation" позволяет сделать анимированные переходы между баннерами:

```
AdView(Context, "http://show.adappter.mobi/request.php", "PUBLISHER_ID", Location,  
animation);
```

Аргумент конструктора vAd "Location" позволяет SDK использовать текущее положение пользователя для более точного таргетинга:

```
AdManager(Context, "http://show.adappter.mobi/request.php", "PUBLISHER_ID", Location);
```

Как только реклама загружена, вызывается этот обработчик в основном User Interface трэде:

```
public void adLoadSucceeded(Ad advertisement) {}
```

Если реклама не может быть загружена, вызывается данный обработчик:

```
public void noAdFound() {}
```

Для vAds Ads вы будете уведомлены, когда RichMediaActivity завершит работу, чтобы возобновить свою Активити:

```
public void adClosed(Ad advertisement, boolean completed) {}
```

Уведомление после показа vAd и приостановки вашей Активити:

```
public void adShown(Ad advertisement, boolean succeeded) {}
```