

Easy Android Plugins

What is an Android Plugin for Unity

When coding in unity, you make your code in an intermediate language. That means that your c sharp code (or unityscript, or boo) is gonna be translated to the target platform. The good thing is that you only have one codebase for all platforms. The bad thing is that you lose the specific features of every platform.

Unity tries to remedy that with the usage of plugins. Plugins are pieces of code that are coded in the specific target platform. This pieces of code can be called from your unity code so you can use the specific features of the desired platform. The plugins have to be coded in a way that let you compile the code for that platform.

The basic building blocks of android projects are jar files, and one manifest file; when making a plugin for Android, you are going to need at least one jar file. Let's see how to do that.

Requirements

Before anything can be done. You must be sure you have the tools needed:

- Java JDK: This let you compile your java code into jar files.
- Android SDK: In order to be able to use the android framework you need the libs provided by the Android SDK. Also, unity needs this tool to be able to form a resulting apk.

Note that unity need these tools to make a valid apk android executable. So if you are able to compile your android project in unity, you most likely have them configured properly.

How to make a plugin

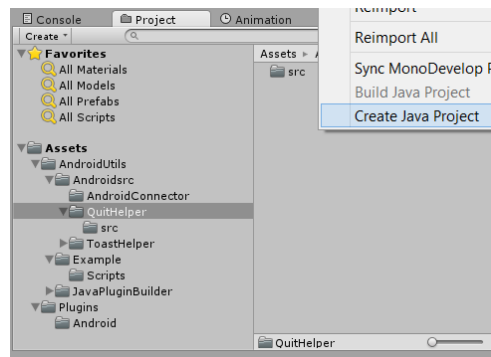
To make a Unity plugin for android, you need:

- jar files with your android code compiled. They should be placed under Plugins/Android folder.
- You can place an Android Manifest file in that same location if you want some extra-features..

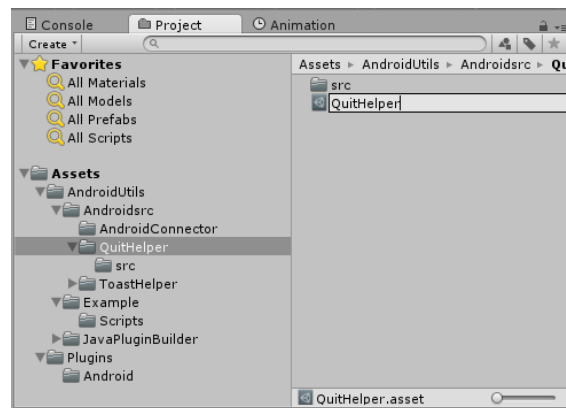
With **Easy Android Plugins** editor extension, you can compile your java files directly from the editor. This is how you do it:

Create project

Choose the folder you are going to use and place your .java code there. To be able to build your jar file, you first need to create a **Java Project file**. Then right click in the directory->Create java project. Be sure the desired root directory is selected.

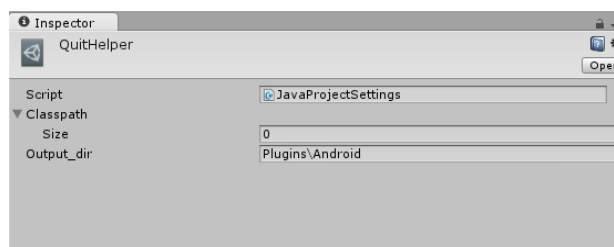


An asset file will be created in that directory. The asset file have the configuration for the jar file to be built. Change the name of the file according to the android plugin you are building.



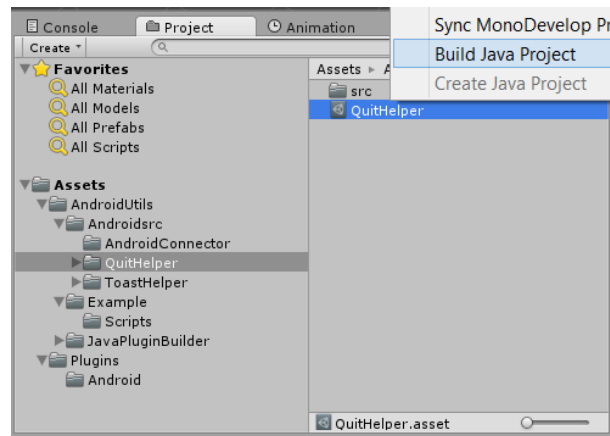
Check out the project properties:

- Output_dir: the directory where your .jar file will be generated relative to your Assets folder..
- Classpath: If your java code have any dependency to another jar file, you must add the route to those jars in this list.



Build project

Next thing to do is to build the jar file. Select the Android code project file->Right click->Build Java project.



All .java files under this directory are going to be included in the jar file.

Your project will be built into the target directory indicated by the **Java Project file** (by default is Plugins/Android). Also, the file name of the .jar file is going to be same name as the name of the **Java Project File**. Any existing .jar file with that name will be deleted and replaced with the new version.

The real requirement when compiling your java code is the bootclasspath. The bootclasspath is the file containing the basics of the android framework. This file is called android.jar and comes with the android sdk. There is one android.jar with every platform installed (it's under android-sdk/platforms/android-version/android.jar). **Easy Android Plugins** picks the latest android.jar you have installed.

Building your Android application

One last thing remains: How to call the android code from your unity code (c sharp, unitiescript, boo).

You need a connector to call that code. Luckily, unity provides a few helper classes to do that. Check out the example scene to see how to do that. Note that unity only executes the android code when executing in an android platform. When executing your app/game in the editor they are going to result in an error.

TL;DR

1. Place your java files in any folder.
2. Create a **Java Project file** in that folder: Select folder->Right click->Create Android project
3. Build the project: Select **Java Project file**->Right click->Build project
4. Make your unity connector to call the java code.
5. Build your Unity project.
6. Enjoy!