Adobe Gaming SDK

⚠ Redirection Notice
This page will redirect to https://www.jetbrains.com/help/idea/actionscript-and-flex.html.

IntelliJ IDEA 12.1 has complete support for the Adobe Gaming SDK, i.e. Adobe AIR SDK bundled together with Away3D, Starling and Feathers frameworks. In this tutorial you'll get step-by-step instructions on how to set up, build and run sample applications included in the SDK.

We assume you have Adobe Gaming SDK installed on your computer and IntelliJ IDEA 12.1 (or later) is running. Samples included in the SDK are shipped with Flash Builder project files (.project, .actionScriptProperties, etc.), and this makes setting up an IntelliJ IDEA project quick and easy, because IntelliJ IDEA is able to import a Flash Builder project in just a few clicks.

If you are new to IntelliJ IDEA but have experience with Flash Builder or other Eclipse-based products, there are some differences in terminology that you should keep in mind:

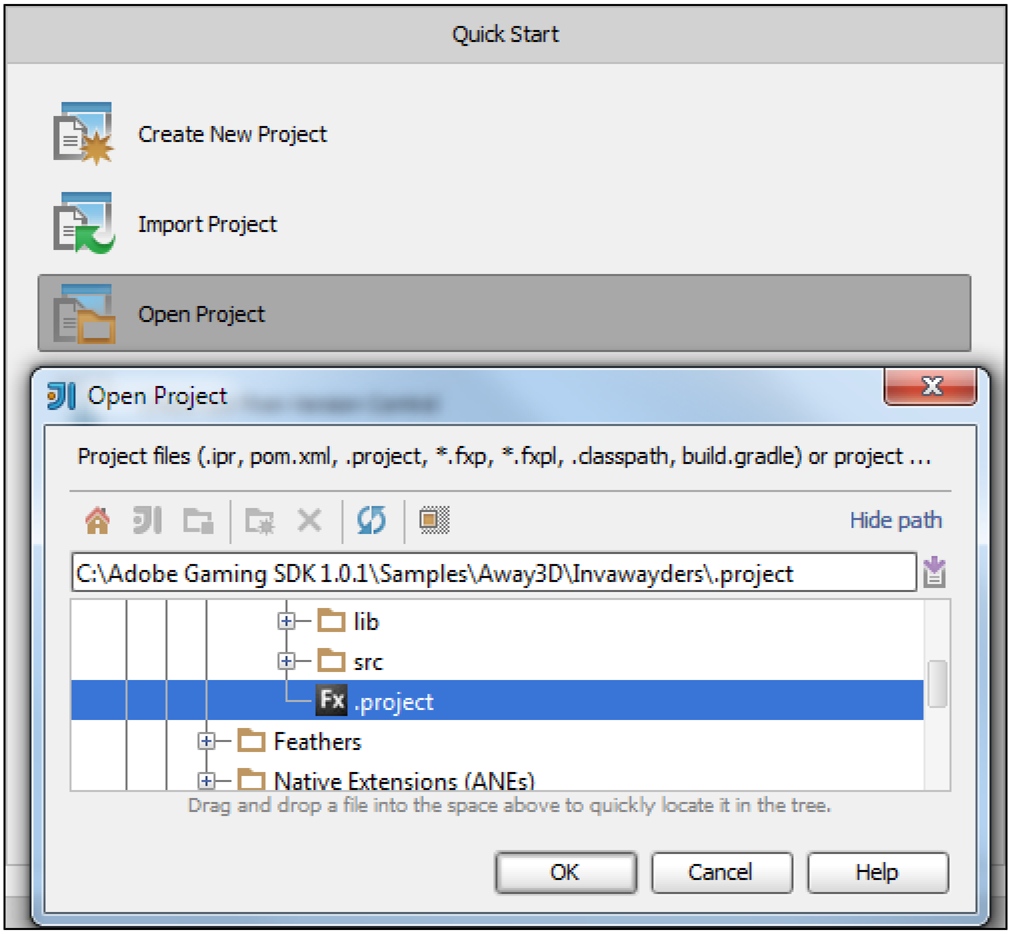
- Flash Builder workspace is the equivalent of IntelliJ IDEA project.
- Flash Builder project is the equivalent of IntelliJ IDEA Flash module.

IntelliJ IDEA Flash module contains one or more Flash build configurations. A build configuration is a set of parameters that define how to produce SWF or SWC, HTML wrapper or AIR descriptor, Desktop, Android or iOS package from the module source.

1. Import Flash Builder project(s)

1a. Import a single Flash Builder project

To import a single Flash Builder project, select File | Open... or click Open Project on the IntelliJ IDEA welcome screen. Then in the file chooser select a .project file, e.g. [SDK]/Samples/Away3D/Invawayders/.project. If you decide to import a single project, you will skip step 2.



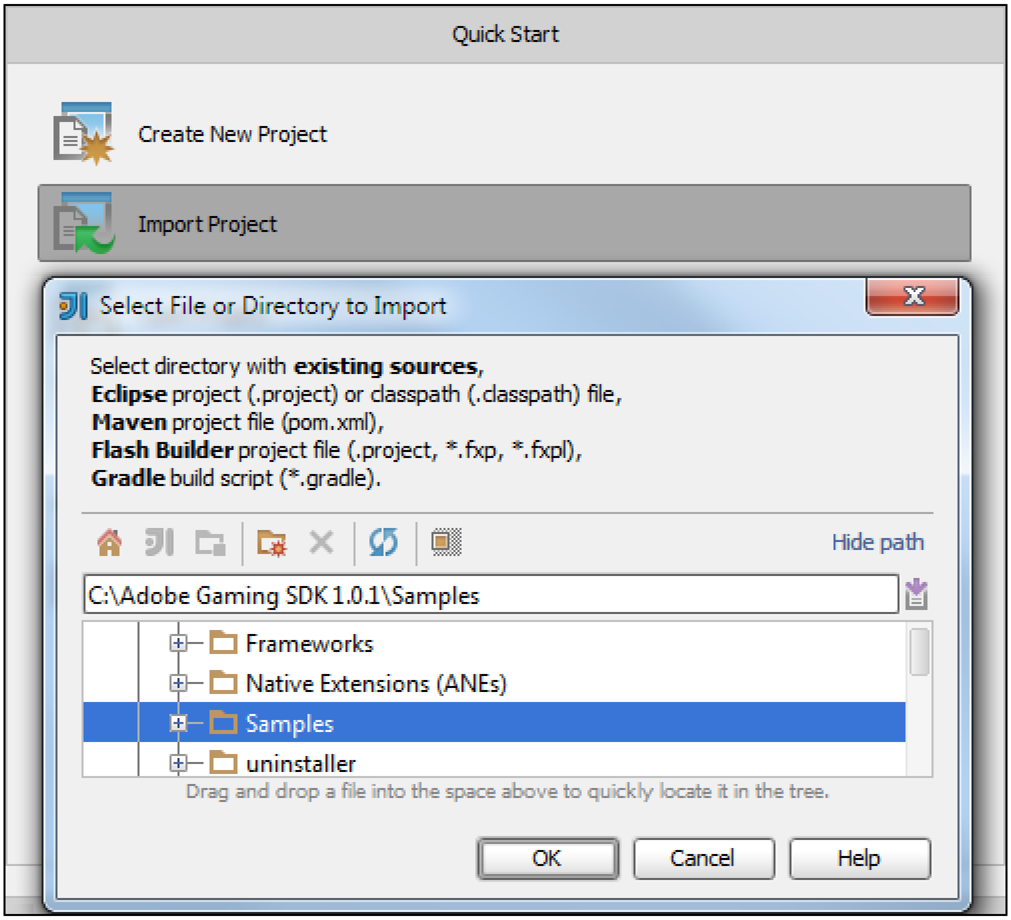
Project files (.ipr, pom.xml, .project, *.fxp, *.fxpl, .classpath, build.gradle) or project ...

C:\Adobe Gaming SDK 1.0.1\Samples\Away3D\Invawayders\.project

Drag and drop a file into the space above to quickly locate it in the tree.

1b. Import multiple Flash Builder projects

To import multiple projects, select File | Import Project... or click Import Project on the IntelliJ IDEA welcome screen. Then in the file chooser select the base folder to search for projects in: [SDK]/Samples



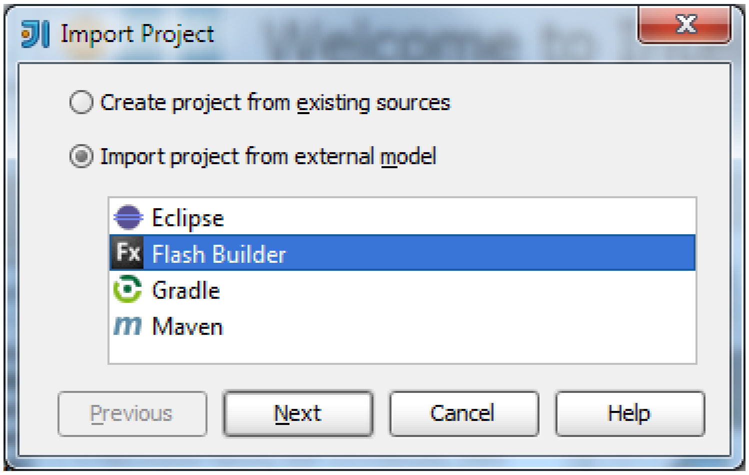
Select directory with **existing sources**,
Eclipse project (.project) or classpath (.classpath) file,
Maven project file (pom.xml),
Flash Builder project file (.project, *.fxp, *.fxpl),
Gradle build script (*.gradle).

C:\Adobe Gaming SDK 1.0.1\Samples

Drag and drop a file into the space above to quickly locate it in the tree.

2. Pass 'Import Project' wizard

At the next step select Import project from external model and Flash Builder.

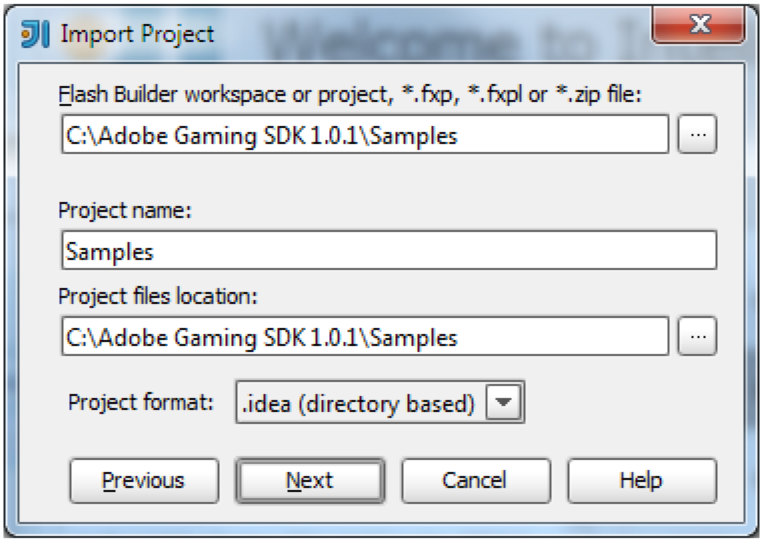


○ Create project from existing sources

● Import project from external model

Flash Builder

Leave the defaults at the next step.



Import Project

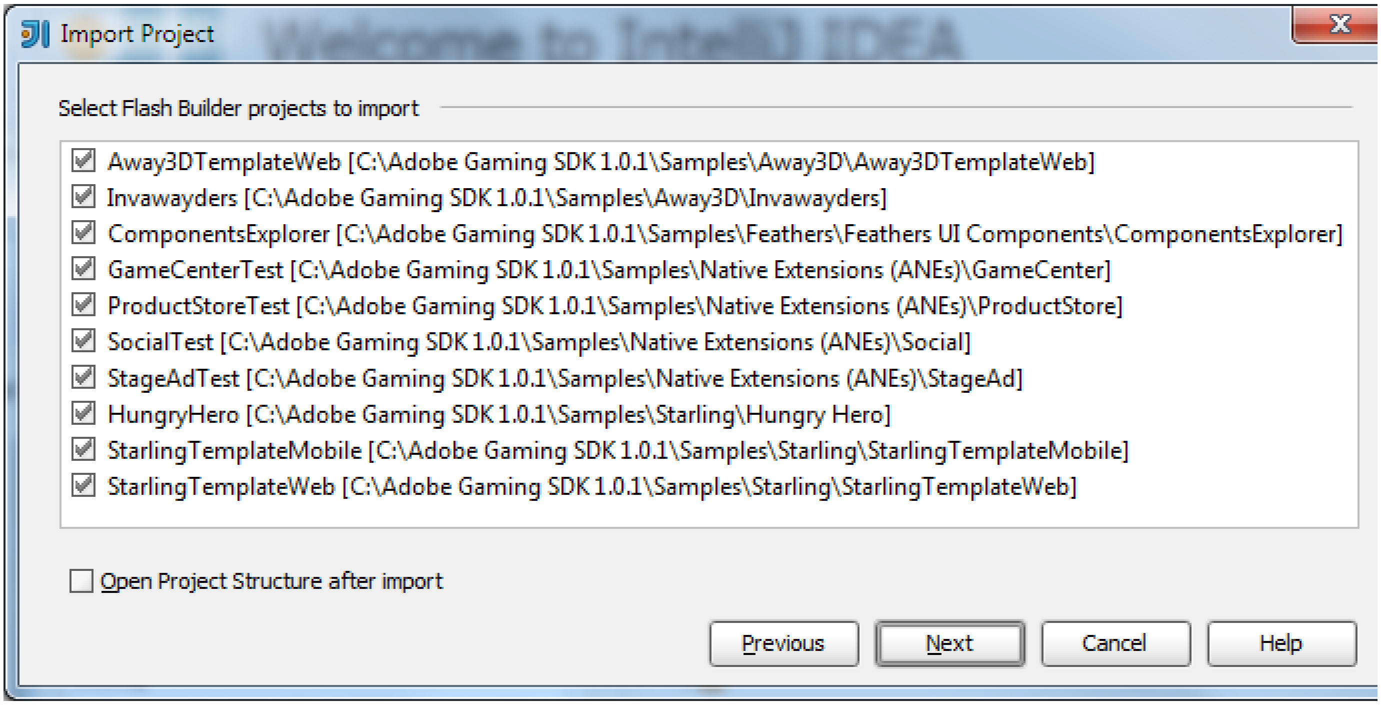
Flash Builder workspace or project, *.fxp, *.fxpl or *.zip file:
C:\Adobe Gaming SDK 1.0.1\Samples

Project name:
Samples

Project files location:
C:\Adobe Gaming SDK 1.0.1\Samples

Project format: .idea (directory based)

Now you can choose which projects to import. Let's keep them all.



Import Project

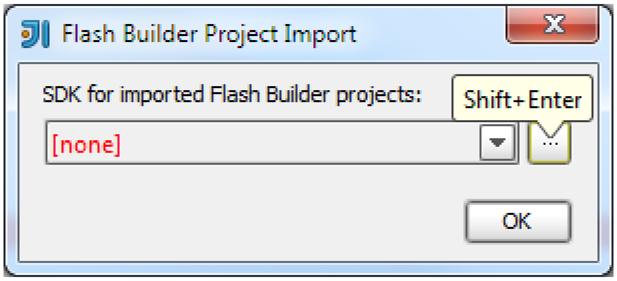
Select Flash Builder projects to import

- ☑ Away3DTemplateWeb [C:\Adobe Gaming SDK 1.0.1\Samples\Away3D\Away3DTemplateWeb]
- ☑ Invawayders [C:\Adobe Gaming SDK 1.0.1\Samples\Away3D\Invawayders]
- ☑ ComponentsExplorer [C:\Adobe Gaming SDK 1.0.1\Samples\Feathers\Feathers UI Components\ComponentsExplorer]
- ☑ GameCenterTest [C:\Adobe Gaming SDK 1.0.1\Samples\Native Extensions (ANEs)\GameCenter]
- ☑ ProductStoreTest [C:\Adobe Gaming SDK 1.0.1\Samples\Native Extensions (ANEs)\ProductStore]
- ☑ SocialTest [C:\Adobe Gaming SDK 1.0.1\Samples\Native Extensions (ANEs)\Social]
- ☑ StageAdTest [C:\Adobe Gaming SDK 1.0.1\Samples\Native Extensions (ANEs)\StageAd]
- ☑ HungryHero [C:\Adobe Gaming SDK 1.0.1\Samples\Starling\Hungry Hero]
- ☑ StarlingTemplateMobile [C:\Adobe Gaming SDK 1.0.1\Samples\Starling\StarlingTemplateMobile]
- ☑ StarlingTemplateWeb [C:\Adobe Gaming SDK 1.0.1\Samples\Starling\StarlingTemplateWeb]

☐ Open Project Structure after import

3. Configure SDK

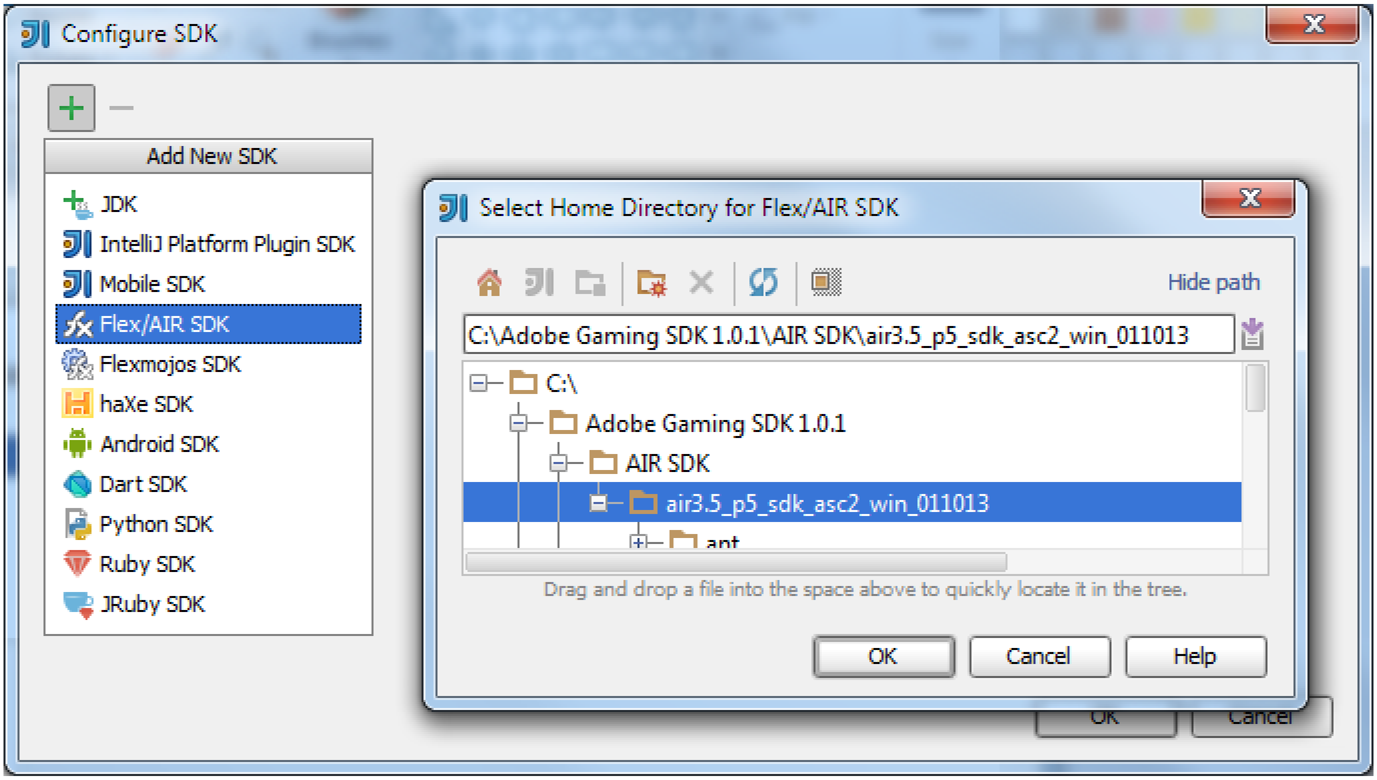
If you have Flash Builder installed on your computer, IntelliJ IDEA will automatically select Flex or AIR SDK from the Flash Builder installation. Otherwise you'll be asked to select SDK manually.



Flash Builder Project Import

SDK for imported Flash Builder projects:
[none]

Click [...], at the next dialog click [+], select Flex/AIR SDK on the popup, and then select the path to AIR SDK bundled with Adobe Gaming SDK: [SDK]/AIR SDK/air3.5_p5_sdk_asc2_win_011013.

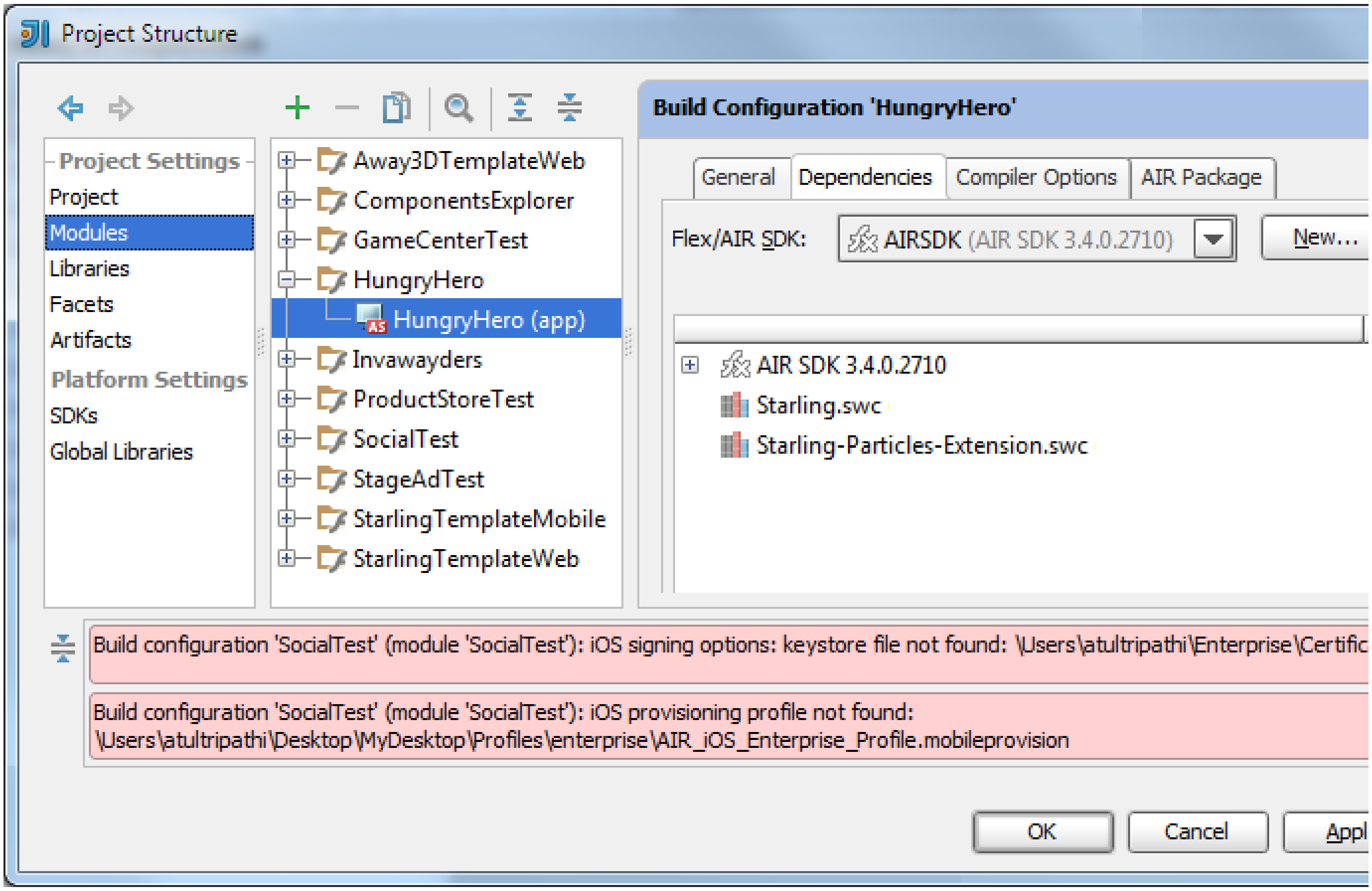


C:\Adobe Gaming SDK 1.0.1\AIR SDK\air3.5_p5_sdk_asc2_win_011013

Drag and drop a file into the space above to quickly locate it in the tree.

4. Explore Project Structure

Now all samples are imported and ready to be started. But first let's explore the structure of the IntelliJ IDEA project we just created (the Project Structure button on the main toolbar).



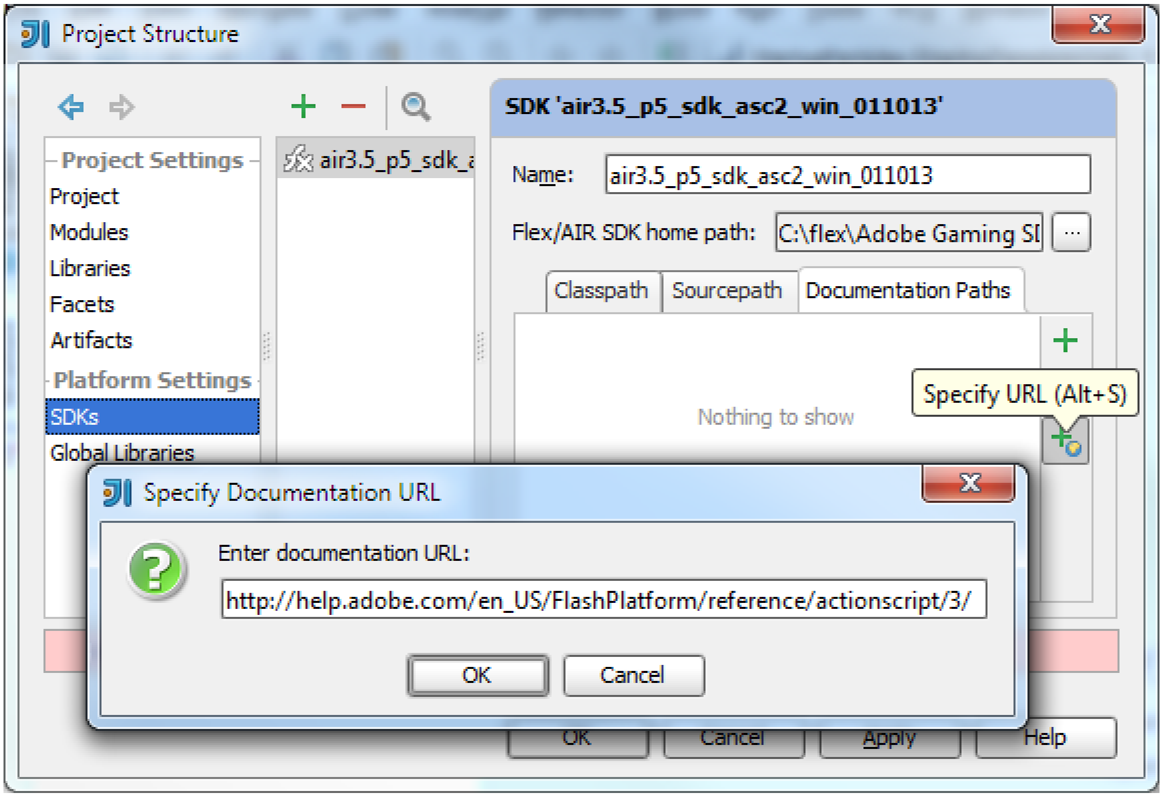
Build Configuration 'HungryHero'

Flex/AIR SDK: AIRSDK (AIR SDK 3.4.0.2710)

Build configuration 'SocialTest' (module 'SocialTest'): iOS signing options: keystore file not found: \Users\atultripathi\Enterprise\Certific

Build configuration 'SocialTest' (module 'SocialTest'): iOS provisioning profile not found: \Users\atultripathi\Desktop\MyDesktop\Profiles\enterprise\AIR_iOS_Enterprise_Profile.mobileprovision

- At the bottom you will see a list of errors, the reason being that keystore and provisioning profile paths were hard-coded in the Flash Builder project. When you decide to launch some app on the real device, you'll need to fix them (click the arrow icon to the right to navigate to the field with the error).
- If you are new to IntelliJ IDEA, we suggest looking through all build configurations and all tabs just to get used to IntelliJ IDEA settings.
- Two things to fix in StarlingTemplateMobile module: disable iOS support for Scaffold_Android build configuration and disable Android support for Scaffold_iOS build configuration. Flash Builder doesn't allow you to select supported mobile targets per app (only per project) - that's why this was not imported automatically.
- If you were not asked to select SDK manually (step 3), that means that IntelliJ IDEA has found a suitable SDK in the Flash Builder installation and has configured it (as is the case on the screenshot). You may keep it as is, or you may change it to AIR SDK from the Adobe Gaming SDK - just click New... button next to the Flex/AIR SDK field on the Dependencies tab.
- In any case we suggest that you configure documentation URL for the SDK. Having the URL, IntelliJ IDEA will fetch documentation automatically when you invoke 'Quick documentation popup', use code completion, etc. To do so, click the Edit button next to the Flex/AIR SDK field, select Documentation Paths tab, click Specify URL button and then OK (c orrect URL is already preselected). Alternatively, you can download the docs and configure a local path.



SDK 'air3.5_p5_sdk_asc2_win_011013'

Name: air3.5_p5_sdk_asc2_win_011013

Flex/AIR SDK home path: C:\flex\Adobe Gaming SI

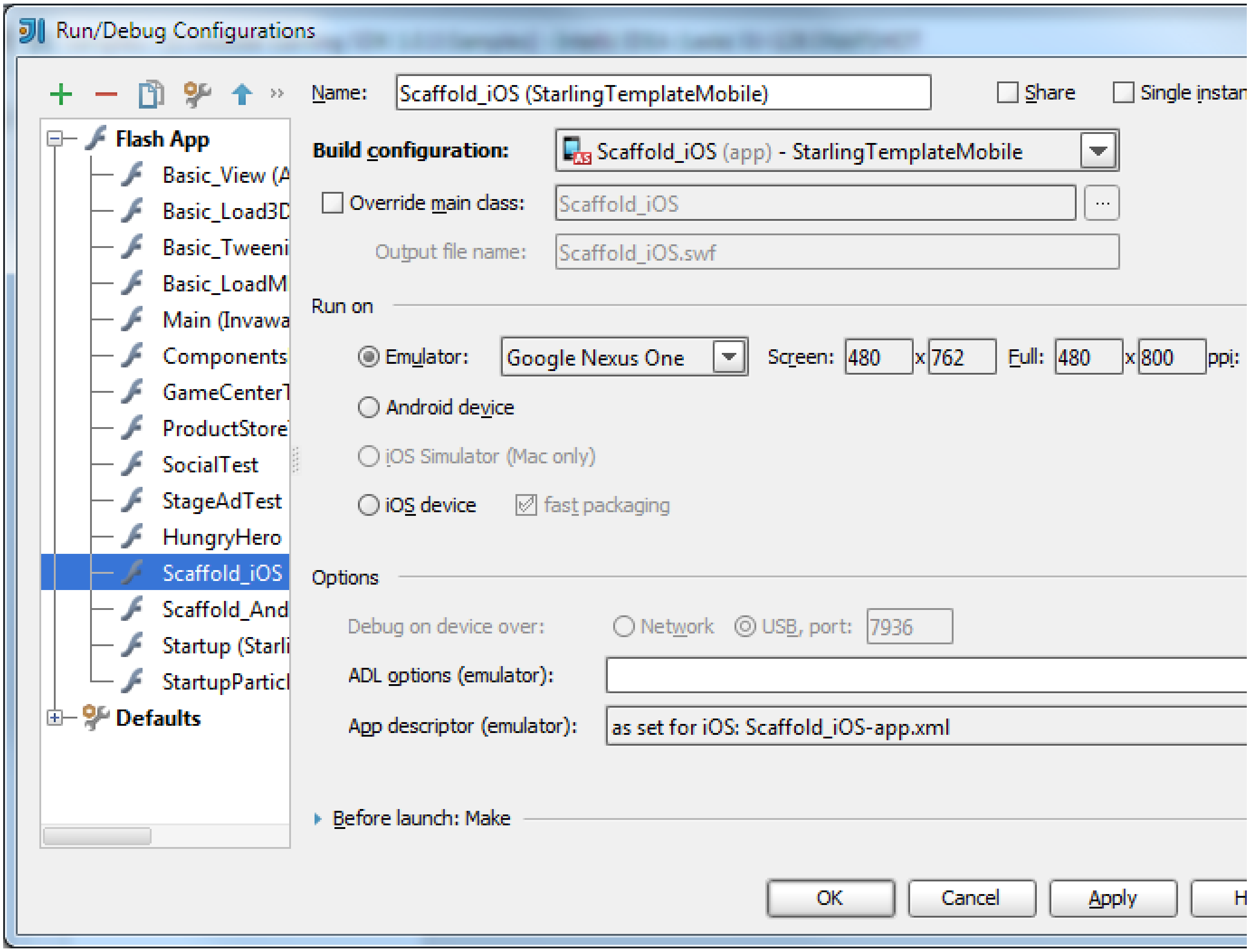
Enter documentation URL:

http://help.adobe.com/en_US/FlashPlatform/reference/actionscript/3/

5. Look through the Run Configurations

Look through the Run configurations created by IntelliJ IDEA (Run | Edit Configurations...). Flash run configurations have different options depending on the target platform of the build configuration on which run configuration is based: Web, Desktop or Mobile. You don't have anything to fix here; the purpose of this step is just to understand that you can configure the following:

- where to launch a Web-targeted app: OS default app, browser or standalone Flash Player;
- where lo launch a Mobile app: Emulator, iOS Simulator, Android device, or iOS device;
- etc.



Name: Scaffold_iOS (StarlingTemplateMobile) □ Share □ Single instan

Build configuration: Scaffold_iOS (app) - StarlingTemplateMobile

□ Override main class: Scaffold_iOS

Output file name: Scaffold_iOS.swf

Run on

● Emulator: Google Nexus One Screen: 480 x 762 Full: 480 x 800 ppi:

○ Android device

○ iOS Simulator (Mac only)

○ iOS device ☑ fast packaging

Options

Debug on device over: ○ Network ● USB, port: 7936

ADL options (emulator):

App descriptor (emulator): as set for iOS: Scaffold_iOS-app.xml

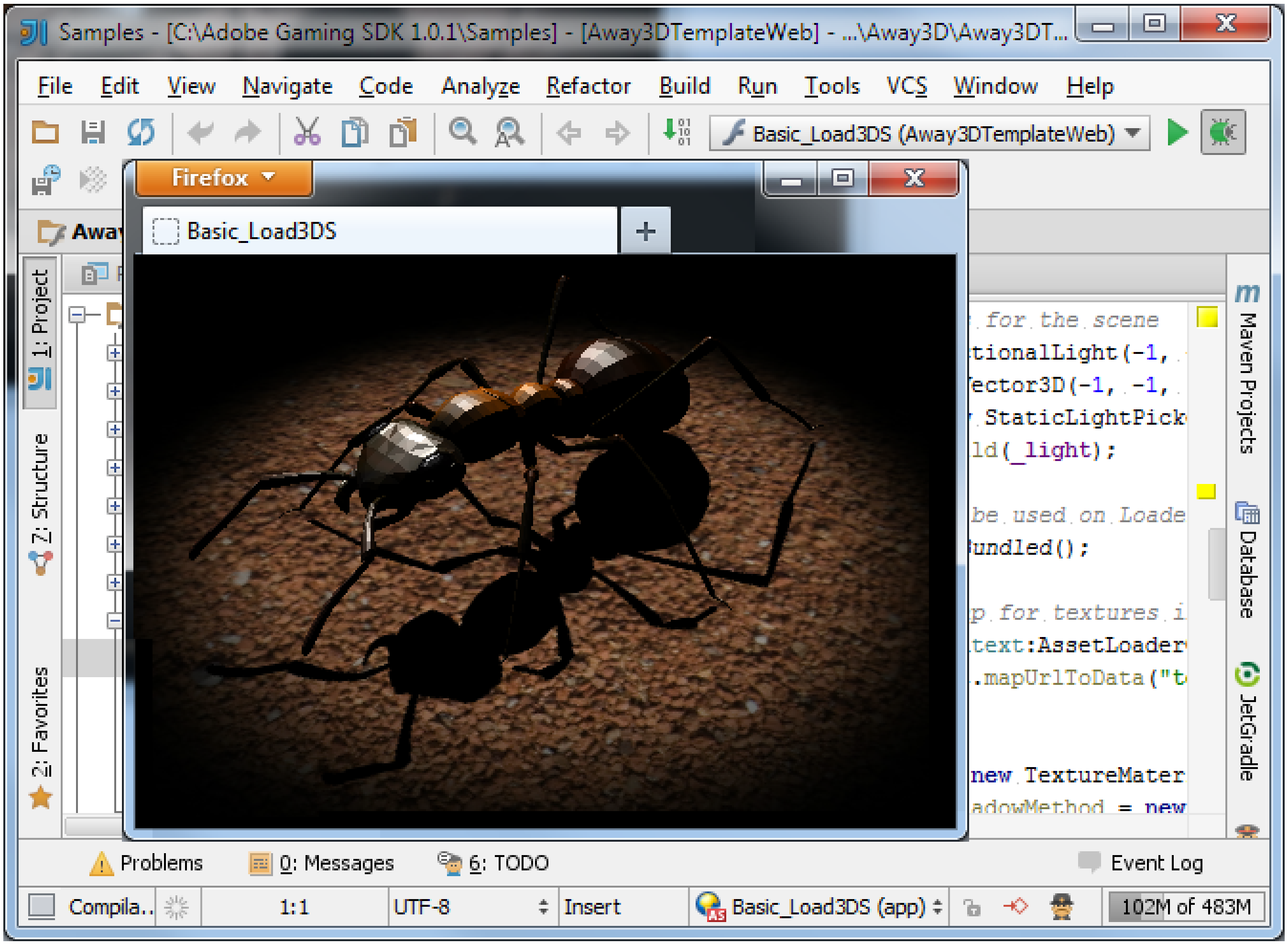
Before launch: Make

6. Explore the source code

Coding assistance is the strongest feature of IntelliJ IDEA. It includes code highlighting, completion, analysis (including UML), refactorings, intentions, inspections, code generation, navigation and much more. We won't focus on these features in this tutorial (a whole book could be written on the topic), but at the first glance you'll notice a lot of warnings about unused imports, unused classes/fields/methods/parameters, methods that could be static, fields that could be converted to local vars, and so on.

7. Start the app.

Just select any run configuration in the corresponding tool bar combo box and click Run or Debug.



```
 for the scene
tionalLight(-1,
ector3D(-1, -1,
 StaticLightPick
ld(_light);

be used on Loade
undled();

p for textures i
text:AssetLoader
.mapUrlToData("t

new TextureMater
adowMethod = new
```

8. Develop with pleasure!