SDK

To develop applications, you need a Software Development Kit (SDK). A typical SDK example is the Java Development Kit (also known as Java SDK, or JDK). This is an SDK for producing Java programs.

IntelliJ IDEA does not include an SDK. So, before you start writing your code, you have to download and install at least one SDK and define it in IntelliJ IDEA. (You can specify an SDK when creating your first project or at a later time.)

To define an SDK in IntelliJ IDEA, generally, all you have to do is to specify its name and location. (This location, normally, is referred to as the SDK home directory. This is the directory in which the SDK is installed). Additional SDK adjustments are possible, however, in most of the cases, they are not necessary.

Supported SDKs

Depending on the nature of your application, you can use:

- Java SDKs to develop Java desktop applications.
- Mobile SDKs (Java Micro Edition (ME) SDKs) to develop applications for Java-enabled mobile devices.
- Android SDKs to develop applications for Android devices.
- Flex and AIR SDKs to develop applications for the Adobe Flash platform.
- Flexmojos SDKs to launch the Flex compiler and debugger. (Such SDKs are created by IntelliJ IDEA automatically when importing Flexmojos projects.)
- IntelliJ IDEA Plugin SDKs to develop plugins for IntelliJ IDEA. (An IntelliJ IDEA installation acts as an IntelliJ IDEA Plugin SDK.)

Note that the set of available SDK types depends on the IntelliJ IDEA edition being used (Community or Ultimate) and also on which plugins are currently enabled.

SDK levels

SDKs, generally, are defined at the global (i.e., the IDE) level and can be used in any of your projects.

One of the SDKs is assigned to the project level and is referred to as a project SDK. This SDK is used by default by all the modules of the corresponding project.

A module can use ("inherit") the project SDK, or, alternatively, a module-specific SDK can be specified.

SDK classpath, sourcepath, and documentation paths

When you add an SDK definition in IntelliJ IDEA, the IDE analyses the contents of the SDK installation folder (home directory) to find out where the corresponding binaries, sources and associated documentation are located. As a result, all the appropriate locations within the SDK are translated into corresponding paths and assigned to one of the following categories:

- Classpath. This is a set of paths to directories and archives containing the SDK binaries. For a JDK, for example, this set includes the paths to class directories and JAR files that make up the Java API and its extensions.

  You can add more SDK libraries to this set, however, this is never necessary for any standard SDK.

- Sourcepath. This is a set of paths to directories and archives containing the source code for the SDK binaries included in the classpath set.

  Including the source files isn't required, but doing so improves IntelliJ IDEA's ability to provide inline help while coding.

- Documentation paths. This is a set of paths to documentation that describes the SDK source code.

  Including documentation paths is also optional, but doing so gives you the ability to view documentation directly from the IDE.

  If online documentation is available for the SDK, you can add the corresponding URL to the SDK documentation paths to be able to access this documentation when needed.

See Also

Procedures:

- Configuring Global, Project and Module SDKs

Reference:

- SDKs

Web Resources:

- Developer Community