Android Tab

In this tab, configure the Android Application Package to be generated.

Item	Description
Debug signed with default certificate	When you choose this option, IntelliJ IDEA signs the extracted package in the *debug mode* using the debug keystore or key that is generated by the Android SDK tools and has the predefined names and passwords: ■ **Keystore name:** debug.keystore ■ **Keystore password:** android ■ **Key alias:** androiddebugkey ■ **Key password:** android ■ **CN** (*common name*): CN=Android Debug,O=Android,C=US This setting is default in IntelliJ IDEA. This means that if you do not configure any artifact manually and choose **Deploy default APK** in the Run/Debug Configuration: Android Application dialog box, IntelliJ IDEA will use the predefined values in the certificate for the generated .apk.
Debug signed with custom certificate	When you choose this option, IntelliJ IDEA signs the extracted package in the *debug mode* using the debug keystore or key that you specify yourself below. You can have a new certificate generated or reuse an existing one. The latter approach is helpful, for example, if you have several applications and you want them all signed with he same certificate so they can be stored in the same folder on the device.

> The debug mode signature is sufficient for testing and debugging applications but does not allow publishing them.

Item	Description
Release unsigned	Choose this option to have a package extracted without a signature for running on an emulator.
Release signed	Choose this option to have a package extracted and signed so the application can be published and run on physical devices. When you choose this option, the fields **Key store path**, **Key store password**, **Key alias**, and **Key password** for specifying the key to sign the package become available.
Key store path	In this text box, specify the location of the file where the key to be used is stored. Type the path manually or click the **Use existing keystore** button to choose the relevant file in the dialog that opens.
Create new	Click this button to open the New Key Store Dialog and configure a new keystore and/or the release key to be generated.
Choose existing	Click this button to have the package signed with a key from an existing keystore file. Choose the relevant keystore file in the dialog that opens. > Later you can choose to use an existing key from this keystore or to have a new key generated in it.

Item	Description
Key store password	In this text box, type the password for the selected existing keystore.
Key alias	In this text box, specify the alias to address the key to use.
Key password	In this text box, specify the password to access the selected key.
Run ProGuard	Select this check box to have IntelliJ IDEA obfuscate the application ⏷ through integration with the ProGuard ⏷ built-in tool. If this check box is selected, the following controls become available: ■ **Config file**: Configuration file for *ProGuard*. By default, this field shows the default configuration file `proguard-project.txt`, created together with the Android module. Type the fully-qualified path to the desired file, or click the **Browse** button ⏹ and choose it in the Select Path dialog box that opens. ■ **Include system proguard file**: if this check box is selected, the system configuration file will be included in the user's configuration file. For Android SDK tools revision 17 and higher, this check box is selected by default. For already existing users projects this check box is cleared.

See Also

Concepts:

■ Artifact

Procedures:

■ Generating Signed and Unsigned Android Application Packages

■ Generating a Signed Release APK Through an Artifact

■ Generating an APK in the Debug Mode

■ Generating an Unsigned Release APK

■ Configuring Artifacts

■ Android

Reference:

■ Artifacts

■ Generate Signed APK Wizard

Web Resources:

■ Developer Community ⏷