

Creating an Android Run/Debug Configuration

An *Android* run/debug configuration is associated with a specific `.apk` and activity. To run or debug the entire application, choose the relevant module and associate the configuration with the *default start-up* activity from the *default package* built from the module specified in the **Module** drop-down list above. The `.apk` is built automatically, no preliminary artifact configuration is required from your side.

To run or debug a custom `.apk` that will be later embedded in an application, first define an artifact to build the `.apk` from. Then, in the run configuration, appoint this `.apk` for deployment and specify an activity from it as the *launch activity*.

To debug an already running application, do not appoint any `.apk` file for deployment and any activity for launch. See [Debugging a Running Application](#).

To create an Android run/debug configuration

For general information about run/debug configurations, see [Creating and Editing Run/Debug Configurations](#).

1. To start creating an Android run configuration, choose **Run | Edit Configuration** on the main menu. Alternatively press **Shift+Alt+F10** and choose **Edit Configuration** from the pop-up menu.
2. Click the **Add New Configuration** button **+** on the toolbar and choose **Android** from the pop-up list.
3. On the [Run/Debug Configuration: Android](#) page that opens, specify the configuration name and choose the module to apply the current configuration to.
4. Specify the `.apk` file to deploy and the activity to launch in the **Package** and **Activity** areas respectively.
 - To run or debug the entire application, choose **Deploy default APK** in the **Package** area and **Launch default activity** in the **Activity** area. IntelliJ IDEA will upload the `.apk` built from the module specified in the **Module** drop-down list above and launch the activity marked as start-up for the chosen module. The `.apk` is built automatically, no preliminary artifact configuration is required from your side.
 - To run or debug a custom `.apk` that will be later embedded in an application, choose **Deploy custom artifact** in the **Package** area and choose the artifact to build the `.apk` from. In this case, you have to define the relevant artifact manually before creating a run/debug configuration, see [Generating a Signed Release APK Through an Artifact and Configuring Artifacts](#). Then choose the **Launch** option in the **Activity** area and specify the start-up activity from the chosen artifact (`.apk`). Type the activity name manually or click the **Browse** button  and select the desired activity in the **Select Activity Class** dialog box, that opens.

The list of available activities is determined by the choice of the module.
 - If you are going to start a debugging session for an already running application, choose **Do not deploy anything** in the **Packages** area to suppress uploading data to the device and then choose **Do not launch activity** in the **Activity** area. Executing a run configuration with these settings is equal to clicking the **Attach debugger to Android process** button  on the toolbar, see [Debugging a Running Application](#).
5. In the **Target Device** area, specify the device to launch the application on or appoint the method for choosing this device.
 - To specify the target virtual device for deployment within the configuration, select the **Emulator** option and choose the desired virtual device from the **Prefer Android Virtual Device** drop-down list.

If none of the existing virtual devices is applicable, [create a new device](#).

- To enable manual selection of the target device during the running/debugging procedure, select the **Show chooser dialog** option. When you launch the configuration, IntelliJ IDEA will open the [Choose Device](#) dialog box, where you will be able to select the relevant device.
- To have IntelliJ IDEA detect a plugged in USB device upon the application start, select the **USB device** option.

Selecting the **Show chooser dialog** or **USB device** option may be helpful if you are going to run the application on a physical device which will be plugged in later and therefore the set of available devices cannot be foreseen.

See Also

Concepts:

- [Run/Debug Configuration](#)

Procedures:

- [Creating and Editing Run/Debug Configurations](#)

Reference:

- [Run/Debug Configuration: Android Application](#)

Web Resources:

- [Developer Community](#) 