<uses-sdk>

SYNTAX:

```
<uses-sdk android:minSdkVersion="integer"
          android:targetSdkVersion="integer"
          android:maxSdkVersion="integer" />
```

CONTAINED IN:

<manifest>

DESCRIPTION:

Lets you express an application's compatibility with one or more versions of the Android platform, by means of an API Level integer. The API Level expressed by an application will be compared to the API Level of a given Android system, which may vary among different Android devices.

Despite its name, this element is used to specify the API Level, *not* the version number of the SDK (software development kit) or Android platform. The API Level is always a single integer. You cannot derive the API Level from its associated Android version number (for example, it is not the same as the major version or the sum of the major and minor versions).

For more information, read about Android API Levels and Versioning Your Applications.

ATTRIBUTES:

android:minSdkVersion

An integer designating the minimum API Level required for the application to run. The Android system will prevent the user from installing the application if the system's API Level is lower than the value specified in this attribute. You should always declare this attribute.

> **Caution:** If you do not declare this attribute, the system assumes a default value of "1", which indicates that your application is compatible with all versions of Android. If your application is *not* compatible with all versions (for instance, it uses APIs introduced in API Level 3) and you have not declared the proper android:minSdkVersion, then when installed on a system with an API Level less than 3, the application will crash during runtime when attempting to access the unavailable APIs. For this reason, be certain to declare the appropriate API Level in the minSdkVersion attribute.

Android Market and <uses-sdk> attributes

Android Market filters the applications that are visible to users, so that users can only see and download applications that are compatible with their devices. One of the ways Market filters applications is by Android version-compatibility. To do this, Market checks the <uses-sdk> attributes in each application's manifest to establish its version-compatibility range, then shows or hides the application based on a comparison with the API Level of the user's Android system version. For more information, see Market Filters.

android:targetSdkVersion

An integer designating the API Level that the application is targetting.

With this attribute set, the application says that it is able to run on older versions (down to minSdkVersion), but was explicitly tested to work with the version specified here. Specifying this target version allows the platform to disable compatibility settings that are not required for the target version (which may otherwise be turned on in order to maintain forward-compatibility) or enable newer features that are not available to older applications. This does not mean that you can program different features for different versions of the platform—it simply informs the platform that you have tested against the target version and the platform should not perform any extra work to maintain forward-compatibility with the target version.

Introduced in: API Level 4

`android:maxSdkVersion`

An integer designating the maximum API Level on which the application is designed to run.

In Android 1.5, 1.6, 2.0, and 2.0.1, the system checks the value of this attribute when installing an application and when revalidating the application after a system update. In either case, if the application's `android:maxSdkVersion` attribute is lower than the API Level used by the system itself, then the system will not allow the application to be installed. In the case of revalidation after system update, this effectively removes your application from the device.

To illustrate how this attribute can affect your application after system updates, consider the following example:

An application declaring `android:maxSdkVersion="5"` in its manifest is published on Android Market. A user whose device is running Android 1.6 (API Level 4) downloads and installs the app. After a few weeks, the user receives an over-the-air system update to Android 2.0 (API Level 5). After the update is installed, the system checks the application's `android:maxSdkVersion` and successfully revalidates it. The application functions as normal. However, some time later, the device receives another system update, this time to Android 2.0.1 (API Level 6). After the update, the system can no longer revalidate the application because the system's own API Level (6) is now higher than the maximum supported by the application (5). The system prevents the application from being visible to the user, in effect removing it from the device.

> **Warning:** Declaring this attribute is not recommended. First, there is no need to set the attribute as means of blocking deployment of your application onto new versions of the Android platform as they are released. By design, new versions of the platform are fully backward-compatible. Your application should work properly on new versions, provided it uses only standard APIs and follows development best practices. Second, note that in some cases, declaring the attribute can **result in your application being removed from users' devices after a system update** to a higher API Level. Most devices on which your appplication is likely to be installed will receive periodic system updates over the air, so you should consider their effect on your application before setting this attribute.

Introduced in: API Level 4

Future versions of Android (beyond Android 2.0.1) will no longer check or enforce the `android:maxSdkVersion` attribute during installation or revalidation. Android Market will continue to use the attribute as a filter, however, when presenting users with applications available for download.

INTRODUCED IN:

API Level 1

← Back to The AndroidManifest.xml File

↑ Go to top