

Package Index

These are the Android APIs. See all [API classes \(classes.html\)](#).

android Contains resource classes used by applications included in the platform and defines application permissions for system features.

android.accessibilityservice The classes in this package are used for development of accessibility service that provide alternative or augmented feedback to the user.

android.accounts

android.animation These classes provide functionality for the property animation system, which allows you to animate object properties of any type. `int`, `float`, and hexadecimal color values are supported by default. You can animate any other type by telling the system how to calculate the values for that given type with a custom `TypeEvaluator` (</reference/android/animation/TypeEvaluator.html>).

For more information, see the [Animation \(/guide/topics/graphics/prop-animation.html\)](/guide/topics/graphics/prop-animation.html) guide.

android.app Contains high-level classes encapsulating the overall Android application model.

android.app.admin	<p>Provides device administration features at the system level, allowing you to create security-aware applications that are useful in enterprise settings, in which IT professionals require rich control over employee devices.</p> <p>For more information, see the Device Administration (/guide /topics/admin/device-admin.html) guide.</p>
android.app.backup	<p>Contains the backup and restore functionality available to applications. If a user wipes the data on their device or upgrades to a new Android-powered device, all applications that have enabled backup can restore the user's previous data when the application is reinstalled.</p> <p>For more information, see the Data Backup (/guide/topics /data/backup.html) guide.</p>
android.appwidget	<p>Contains the components necessary to create "app widgets", which users can embed in other applications (such as the home screen) to quickly access application data and services without launching a new activity.</p> <p>For more information, see the App Widgets (/guide/topics /appwidgets/index.html) guide.</p>
android.bluetooth	<p>Provides classes that manage Bluetooth functionality, such</p>

as scanning for devices, connecting with devices, and managing data transfer between devices. The Bluetooth API supports both "Classic Bluetooth" and Bluetooth Low Energy.

For more information about Classic Bluetooth, see the [Bluetooth \(/guide/topics/connectivity/bluetooth.html\)](/guide/topics/connectivity/bluetooth.html) guide. For more information about Bluetooth Low Energy, see the [Bluetooth Low Energy \(/guide/topics/connectivity/bluetooth-le.html\)](/guide/topics/connectivity/bluetooth-le.html) (BLE) guide.

`android.content`

Contains classes for accessing and publishing data on a device.

`android.content.pm`

Contains classes for accessing information about an application package, including information about its activities, permissions, services, signatures, and providers.

`android.content.res`

Contains classes for accessing application resources, such as raw asset files, colors, drawables, media or other other files in the package, plus important device configuration details (orientation, input types, etc.) that affect how the application may behave.

For more information, see the [Application Resources \(/guide/topics/resources/index.html\)](/guide/topics/resources/index.html) guide.

android.database	Contains classes to explore data returned through a content provider.
android.database.sqlite	Contains the SQLite database management classes that an application would use to manage its own private database.
android.drm	Provides classes for managing DRM content and determining the capabilities of DRM plugins (agents).
android.gesture	Provides classes to create, recognize, load and save gestures.
android.graphics	Provides low level graphics tools such as canvases, color filters, points, and rectangles that let you handle drawing to the screen directly.
android.graphics.drawable	Provides classes to manage a variety of visual elements that are intended for display only, such as bitmaps and gradients.
android.graphics.drawable.shapes	Contains classes for drawing geometric shapes.
android.graphics.pdf	Contains classes for manipulation of PDF content.
android.hardware	Provides support for hardware features, such as the camera and other sensors.
android.hardware.display	
android.hardware.input	
android.hardware.location	
android.hardware.usb	Provides support to communicate with USB hardware peripherals that are connected to Android-powered devices.

	<p>For more information, see the USB (/guide/topics/connectivity/usb/index.html) guide.</p> <p>Base classes for writing input methods (such as software keyboards).</p> <p>Contains the framework API classes that define Android location-based and related services.</p> <div><p>Note: The Google Location Services API, part of Google Play Services, provides a more powerful, high-level framework that automates tasks such as location provider choice and power management. Location Services also provides new features such as activity detection that aren't available in the framework API. Developers who are using the framework API, as well as developers who are just now adding location-awareness to their apps, should strongly consider using the Location Services API.</p><p>To learn more about the Location Services API, see Location APIs (/google/play-services/location.html).</p></div> <p>For more information about the framework API, see the Location and Maps (/guide/topics/location/index.html) guide.</p> <p>Provides classes that manage various media interfaces in audio and video.</p>
android.inputmethodservice	
android.location	
android.media	

<code>android.media.audiofx</code>	Provides classes that manage audio effects implemented in the media framework.
<code>android.media.effect</code>	Provides classes that allow you to apply a variety of visual effects to images and videos.
<code>android.mtp</code>	Provides APIs that let you interact directly with connected cameras and other devices, using the PTP (Picture Transfer Protocol) subset of the MTP (Media Transfer Protocol) specification.
<code>android.net</code>	Classes that help with network access, beyond the normal <code>java.net.*</code> APIs.
<code>android.net.http</code> <code>android.net.nsd</code>	
<code>android.net.rtp</code>	Provides APIs for RTP (Real-time Transport Protocol), allowing applications to manage on-demand or interactive data streaming.
<code>android.net.sip</code>	Provides access to Session Initiation Protocol (SIP) functionality, such as making and answering VOIP calls using SIP. For more information, see the Session Initiation Protocol (/guide/topics/connectivity/sip.html) developer guide.
<code>android.net.wifi</code>	Provides classes to manage Wi-Fi functionality on the device.

`android.net.wifi.p2p` Provides classes to create peer-to-peer (P2P) connections with Wi-Fi Direct.

`android.net.wifi.p2p.nsd`

`android.nfc` Provides access to Near Field Communication (NFC) functionality, allowing applications to read NDEF message in NFC tags. A "tag" may actually be another device that appears as a tag.

For more information, see the [Near Field Communication \(/guide/topics/connectivity/nfc/index.html\)](/guide/topics/connectivity/nfc/index.html) guide.

`android.nfc.cardemulation`

`android.nfc.tech` These classes provide access to a tag technology's features, which vary by the type of tag that is scanned.

`android.opengl` Provides an OpenGL ES static interface and utilities.

`android.os` Provides basic operating system services, message passing, and inter-process communication on the device.

`android.os.storage` Contains classes for the system storage service, which manages binary asset filesystems known as Opaque Binary Blobs (OBBs).

`android.preference` Provides classes that manage application preferences and implement the preferences UI.

`android.print`

Overview

	Provides classes for implementing print support in applications and also contains all base classes and abstractions involved in printing.
<code>android.print.pdf</code>	
<code>android.printservice</code>	Provides classes for implementing print services.
<code>android.provider</code>	Provides convenience classes to access the content providers supplied by Android.
<code>android.renderscript</code>	RenderScript provides support for high-performance computation across heterogeneous processors. For more information, see the RenderScript (/guide/topics/renderscript/index.html) developer guide.
<code>android.sax</code>	A framework that makes it easy to write efficient and robust SAX handlers.
<code>android.security</code>	Provides access to a few facilities of the Android security subsystems.
<code>android.service.dreams</code> <code>android.service.notification</code>	
<code>android.service.textservice</code>	Provides classes that allow you to create spell checkers in a manner similar to the input method framework (for IMEs).
<code>android.service.wallpaper</code> <code>android.speech</code> <code>android.speech.tts</code>	

android.support.v13.app	Support classes to access some of the android.app package features introduced after API level 13 in a backwards compatible fashion.
android.support.v4.accessibilityservice	Support android.accessibilityservice classes to assist with development of applications for android API level 4 or later.
android.support.v4.app	Support android.app classes to assist with development of applications for android API level 4 or later.
android.support.v4.content	Support android.content classes to assist with development of applications for android API level 4 or later.
android.support.v4.content.pm	Support android.content.pm classes to assist with development of applications for android API level 4 or later.
android.support.v4.database	Support android.database classes to assist with development of applications for android API level 4 or later.
android.support.v4.graphics.drawable	
android.support.v4.hardware.display	
android.support.v4.media	
android.support.v4.net	
android.support.v4.os	Support android.os classes to assist with development of applications for android API level 4 or later.
android.support.v4.print	
android.support.v4.text	
android.support.v4.util	Support android.util classes to assist with development of applications for android API level 4 or later.
android.support.v4.view	Support android.util classes to assist with development of applications for android API

	level 4 or later. Support classes to access some of the <code>android.view.accessibility</code> package features introduced after API level 4 in a backwards compatible fashion.
<code>android.support.v4.view.accessibility</code>	
<code>android.support.v4.widget</code>	Support <code>android.widget</code> classes to assist with development of applications for android API level 4 or later.
<code>android.support.v7.app</code> <code>android.support.v7.appcompat</code> <code>android.support.v7.gridlayout</code>	
<code>android.support.v7.media</code>	Contains APIs that control the routing of media channels and streams from the current device to external speakers and destination devices.
<code>android.support.v7.mediarouter</code> <code>android.support.v7.view</code> <code>android.support.v7.widget</code> <code>android.support.v8.renderscript</code>	
<code>android.telephony</code>	Provides APIs for monitoring the basic phone information, such as the network type and connection state, plus utilities for manipulating phone number strings.
<code>android.telephony.cdma</code>	Provides APIs for utilizing CDMA-specific telephony features.
<code>android.telephony.gsm</code>	Provides APIs for utilizing GSM-specific telephony features, such as text/data/PDU SMS messages.
<code>android.test</code>	A framework for writing Android test cases and suites.

For more information, see the

	Testing (/tools/testing/index.html) developer guide.
<code>android.test.mock</code>	Utility classes providing stubs or mocks of various Android framework building blocks. For more information, see the Testing (/tools/testing/index.html) guide.
<code>android.test.suitebuilder</code>	Utility classes supporting the test runner classes.
<code>android.text</code>	Provides classes used to render or track text and text spans on the screen.
<code>android.text.format</code>	This package contains alternative classes for some text formatting classes defined in <code>java.util</code> and <code>java.text</code> .
<code>android.text.method</code>	Provides classes that monitor or modify keypad input.
<code>android.text.style</code>	Provides classes used to view or change the style of a span of text in a View object.
<code>android.text.util</code>	Utilities for converting identifiable text strings into clickable links and creating RFC 822-type message (SMTP) tokens.
<code>android.transition</code>	The classes in this package enable "scenes & transitions" functionality for view hierarchies.
<code>android.util</code>	Provides common utility methods such as date/time manipulation, base64 encoders and decoders, string

	and number conversion methods, and XML utilities.
android.view	Provides classes that expose basic user interface classes that handle screen layout and interaction with the user.
android.view.accessibility	The classes in this package are used to represent screen content and changes to it as well as APIs for querying the global accessibility state of the system.
android.view.animation	Provides classes that handle tweened animations.
android.view.inputmethod	Framework classes for interaction between views and input methods (such as soft keyboards).
android.view.textservice	
android.webkit	Provides tools for browsing the web.
android.widget	The widget package contains (mostly visual) UI elements to use on your Application screen.
dalvik.bytecode	
dalvik.system	
java.awt.font	
java.beans	
java.io	
java.lang	
java.lang.annotation	
java.lang.ref	
java.lang.reflect	
java.math	
java.net	
java.nio	

java.nio.channels

java.nio.channels.spi

java.nio.charset

java.nio.charset.spi

java.security

Extensible cryptographic *service provider infrastructure* (SPI) for using and defining services such as **Certificates** ([/reference /java/security /Certificate.html](/reference/java/security/Certificate.html)), **Keys** ([/reference/java/security /Key.html](/reference/java/security/Key.html)), **KeyStores** ([/reference/java/security /KeyStore.html](/reference/java/security/KeyStore.html)), **MessageDigests** ([/reference /java/security /MessageDigest.html](/reference/java/security/MessageDigest.html)), and **Signatures** ([/reference /java/security/Signature.html](/reference/java/security/Signature.html)).

java.security.acl

This package provides the classes and the interfaces needed to build Access Control Lists.

java.security.cert

This package provides all the classes and all the interfaces needed to generate, administer and verify X.509 certificates.

java.security.interfaces

This package provides the interfaces needed to generate:
(1) Keys for the RSA asymmetric encryption algorithm using the PKCS#1 standard; (2) Keys for the Digital Signature Algorithm (DSA) specified by FIPS-186; (3) Keys for a generic Elliptic Curve asymmetric encryption

algorithm.

java.security.spec

This package provides the classes and interfaces needed to specify keys and parameters for encryption and signing algorithms.

java.sql

java.text

java.util

java.util.concurrent

Utility classes commonly useful in concurrent programming.

java.util.concurrent.atomic

A small toolkit of classes that support lock-free thread-safe programming on single variables.

java.util.concurrent.locks

Interfaces and classes providing a framework for locking and waiting for conditions that is distinct from built-in synchronization and monitors.

java.util.jar

java.util.logging

java.util.prefs

java.util.regex

java.util.zip

javax.crypto

This package provides the classes and interfaces for cryptographic applications implementing algorithms for encryption, decryption, or key agreement.

javax.crypto.interfaces

This package provides the interfaces needed to implement the Diffie-Hellman (DH) key agreement's algorithm as specified by PKCS#3.

<code>javax.crypto.spec</code>	This package provides the classes and interfaces needed to specify keys and parameter for encryption.
<code>javax.microedition.khronos.egl</code>	
<code>javax.microedition.khronos.opengles</code>	Provides a standard OpenGL interface.
<code>javax.net</code>	
<code>javax.net.ssl</code>	This package provides classes and interfaces needed to use the Secure Sockets Layer (SSL) protocol and the successor Transport Layer Security (TLS) protocol.
<code>javax.security.auth</code>	
<code>javax.security.auth.callback</code>	This package provides classes and interfaces needed to interact with the application in order to execute the authentication and authorization processes.
<code>javax.security.auth.login</code>	This package provides a pluggable and stackable authentication system based on ideas and concepts from the Unix-PAM module.
<code>javax.security.auth.x500</code>	This package provides classes needed to store X.500 principals and their credentials.
<code>javax.security.cert</code>	This package is provided only for compatibility reasons.
<code>javax.sql</code>	
<code>javax.xml</code>	

javax.xml.datatype	
javax.xml.namespace	
javax.xml.parsers	
javax.xml.transform	
javax.xml.transform.dom	
javax.xml.transform.sax	
javax.xml.transform.stream	
javax.xml.validation	
javax.xml.xpath	
junit.framework	The junit test framework.
junit.runner	Utility classes supporting the junit test framework.
org.apache.http	The core interfaces and classes of the HTTP components.
org.apache.http.auth	The API for client-side HTTP authentication against a server, commonly referred to as <i>HttpAuth</i> .
org.apache.http.auth.params	Parameters for configuring <i>HttpAuth</i> .
org.apache.http.client	The API for client-side HTTP communication and entry point to the <i>HttpClient</i> module.
org.apache.http.client.entity	
org.apache.http.client.methods	Request implementations for the various HTTP methods like GET and POST.
org.apache.http.client.params	Parameters for configuring <i>HttpClient</i> .
org.apache.http.client.protocol	Additional request and response interceptors.
org.apache.http.client.utils	Helpers and utility classes for <i>HttpClient</i> .
org.apache.http.conn	The client-side connection management and handling API at the heart of what is referred to as <i>HttpConn</i> .
org.apache.http.conn.params	Parameters for configuring <i>HttpConn</i> .
org.apache.http.conn.routing	The client-side route representation and tracking API, part of <i>HttpConn</i> .

org.apache.http.conn.scheme	
org.apache.http.conn.ssl	TLS/SSL specific parts of the <i>HttpConn</i> API.
org.apache.http.conn.util	
org.apache.http.cookie	The API for client-side state management via cookies, commonly referred to as <i>HttpCookie</i> .
org.apache.http.cookie.params	Parameters for configuring <i>HttpCookie</i> .
org.apache.http.entity	Representations for HTTP message entities.
org.apache.http.impl	Default implementations for interfaces in <code>org.apache.http</code> .
org.apache.http.impl.auth	
org.apache.http.impl.client	
org.apache.http.impl.conn	
org.apache.http.impl.conn.tsccm	The implementation of a thread-safe client connection manager.
org.apache.http.impl.cookie	
org.apache.http.impl.entity	Default implementations for interfaces in <code>org.apache.http.entity</code> .
org.apache.http.impl.io	Default implementations for interfaces in <code>org.apache.http.io</code> .
org.apache.http.io	The transport layer abstraction of the HTTP components.
org.apache.http.message	A selection of HTTP message implementations.
org.apache.http.params	The parameterization framework for HTTP components.
org.apache.http.protocol	HTTP protocol execution framework.
org.apache.http.util	Mostly utility classes with static helper methods for various purposes.
org.json	
org.w3c.dom	

`org.w3c.dom.ls`

`org.xml.sax`

`org.xml.sax.ext`

`org.xml.sax.helpers`

`org.xmlpull.v1`

`org.xmlpull.v1.sax2`