

# Introduction to the Google Drive Android API

**Note:** The Drive Android API does not currently support content in Team Drives. To access content in Team Drives, use the [Drive REST API \(https://developers.google.com/drive/v3/web\)](https://developers.google.com/drive/v3/web).

The Drive Android API is a native API that conforms to standard Android best practices and coding conventions. It is an improvement over the generated client API, simplifying many common tasks associated with using the Drive service on mobile devices. The API automatically handles previously complex tasks such as offline access and syncing files. This allows you to read and write files as if Drive were a local file system.

The Drive Android API also significantly reduces the installed size of Drive-enabled apps. Because the API is part of the [Google Play services client library \(http://developer.android.com/google/play-services/index.html\)](http://developer.android.com/google/play-services/index.html), you no longer need to compile your app using the [Google APIs Client Library for Java \(http://code.google.com/p/google-api-java-client/\)](http://code.google.com/p/google-api-java-client/) in order to access the Drive service.

## Audience

---

This documentation assumes you are familiar with [Android development \(http://developer.android.com\)](http://developer.android.com) and object-oriented programming concepts. You should also be familiar with [Google Drive \(http://drive.google.com/\)](http://drive.google.com/) from a user's point of view.

## API Overview

---

The Google Drive Android API provides abstractions for managing file content and metadata. Files are represented by the [\*\*DriveFile\*\*](https://developers.google.com/android/reference/com/google/android/gms/drive/DriveFile.html) (<https://developers.google.com/android/reference/com/google/android/gms/drive/DriveFile.html>) interface; folders by the [\*\*DriveFolder\*\*](https://developers.google.com/android/reference/com/google/android/gms/drive/DriveFolder.html) (<https://developers.google.com/android/reference/com/google/android/gms/drive/DriveFolder.html>) interface. Both share a common superinterface, [\*\*DriveResource\*\*](https://developers.google.com/android/reference/com/google/android/gms/drive/DriveResource.html) (<https://developers.google.com/android/reference/com/google/android/gms/drive/DriveResource.html>), which contains a [\*\*DriveId\*\*](https://developers.google.com/android/reference/com/google/android/gms/drive/DriveId.html) (<https://developers.google.com/android/reference/com/google/android/gms/drive/DriveId.html>). This is the unique identifier for all files and folders.

The API also includes a package for querying for files

(<https://developers.google.com/drive/android/queries>) based on metadata attributes

(<https://developers.google.com/drive/android/metadata>), such as the title.

## Resource metadata

Metadata for resources are encapsulated in the Metadata

(<https://developers.google.com/android/reference/com/google/android/gms/drive/Metadata.html>) class that contains all details about a file or folder including the title, the MIME type, and whether the file is editable, starred or trashed. The Metadata is fetched for a `DriveResource` by calling the `DriveResourceClient.getMetadata()` method.

The Metadata may be set or changed when creating or updating a file using a

MetadataChangeSet

(<https://developers.google.com/android/reference/com/google/android/gms/drive/MetadataChangeSet.html>)

and setting the appropriate values, then calling the `DriveResourceClient.updateMetadata()` method.

For more information, see Working with Metadata

(<https://developers.google.com/drive/android/metadata>).

## Files

You can open files programmatically using the `DriveResourceClient.openFile()` method or by allowing the user to select a file using a file selector activity created with the `DriveClient.newOpenFileActivityResultSender()` method. The file contains a reference to the contents of the file, encapsulated in the DriveContents

(<https://developers.google.com/android/reference/com/google/android/gms/drive/DriveContents.html>) class, which provides access to read and write the underlying data. You can read the contents of a drive file by accessing an input stream of its contents by calling `DriveContents.getInputStream()`.

To create a file, you can use either the DriveResourceClient.createFile

([https://developers.google.com/android/reference/com/google/android/gms/drive/DriveResourceClient.html#createFile\(com.google.android.gms.drive.DriveFolder,com.google.android.gms.drive.MetadataChangeSet,com.google.android.gms.drive.DriveContents\)](https://developers.google.com/android/reference/com/google/android/gms/drive/DriveResourceClient.html#createFile(com.google.android.gms.drive.DriveFolder,com.google.android.gms.drive.MetadataChangeSet,com.google.android.gms.drive.DriveContents)))

method or the DriveClient.newCreateFileActivityResultSender()

([https://developers.google.com/android/reference/com/google/android/gms/drive/DriveClient.html#newCreateFileActivityIntentSender\({gms\\_drive\\_package\\_name}.CreateFileActivityOptions\)](https://developers.google.com/android/reference/com/google/android/gms/drive/DriveClient.html#newCreateFileActivityIntentSender({gms_drive_package_name}.CreateFileActivityOptions)))

method, which presents a dialog to the user that allows them to create the new file in their Drive.

For more information about how to work with files, see [Working with File Contents](#)

(<https://developers.google.com/drive/android/files>). For more information about how to create files, see [Creating Files](#) (<https://developers.google.com/drive/android/create-file>)

## Folders

The Drive for Android API has several methods for manipulating folders:

- `DriveResourceClient.createFolder()`, `DriveResourceClient.createFile()`, methods for creating folders or files, respectively, within a folder
- `DriveResourceClient.listChildren()`, `DriveResourceClient.queryChildren()` methods for listing or querying the children of a folder.

For more information, see [Working with Folders](#)

(<https://developers.google.com/drive/android/folders>).

## Activity Builders

The Drive for Android API has two methods that implement UI components to simplify common tasks associated with opening and creating files.

`DriveClient.newCreateFileActivityIntentSender()` method helps you quickly create a file, set its initial folder, title, contents, and metadata. The class also implements a simple UI that allows a user to specify the file's title and destination folder. For more information, see [Creating Files](#) (<https://developers.google.com/drive/android/create-file>).

`DriveClient.newOpenFileActivityIntentSender()` method helps you easily open a file selected by the user. The class displays a simple UI to let the user navigate and select from a list of files that match the specified MIME types. When you invoke the class, you specify an initial folder where the user's navigation begins. You can also specify a preferred set of MIME types for the file list.

## Queries

Queries allow you to search for specific files or folders based on their metadata attributes, such as the title. Queries are represented by the [Query](#)

(<https://developers.google.com/android/reference/com/google/android/gms/drive/query/Query.html>)

class and can be constructed using the [Query.Builder](#)

(<https://developers.google.com/android/reference/com/google/android/gms/drive/query/Query.Builder.html>)

class.

For more information, see [Querying for Files](https://developers.google.com/drive/android/queries) (<https://developers.google.com/drive/android/queries>).

## Developer Resources

---

In addition to the sample code bundled with the [Google Play services SDK](#)

(<http://developer.android.com/google/play-services/index.html>), many more examples can be found at the official [Google Drive GitHub repository](http://github.com/googledrive) (<http://github.com/googledrive>).

---

*Except as otherwise noted, the content of this page is licensed under the [Creative Commons Attribution 3.0 License](http://creativecommons.org/licenses/by/3.0/) (<http://creativecommons.org/licenses/by/3.0/>), and code samples are licensed under the [Apache 2.0 License](http://www.apache.org/licenses/LICENSE-2.0) (<http://www.apache.org/licenses/LICENSE-2.0>). For details, see our [Site Policies](https://developers.google.com/terms/site-policies) (<https://developers.google.com/terms/site-policies>). Java is a registered trademark of Oracle and/or its affiliates.*

*Last updated November 6, 2017.*



### [Blog](#)

Stay up-to-date on developing  
for G Suite



### [Google+ Community](#)

Add us to your circles and  
connect with developers



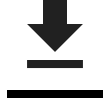
### [GitHub](#)

Explore our sample apps or  
fork them to build your own



### [Stack Overflow](#)

Ask questions with the google-  
drive-sdk tag



### Client Libraries

Download a client library to  
help you get started



### Videos

Check out Drive API videos on  
YouTube