

Setting up Google Pay Receipt Verification

To setup Google Pay Receipt Verification for backend, you will need 3 things for the .env:

GOOGLE_PLAY_CLIENT_ID

GOOGLE_PLAY_CLIENT_SECRET

GOOGLE_PLAY_REFRESH_TOKEN

Follow the steps below to get these 3 things.

29 Steps [View most recent version on Tango.ai](#) 

Created by

Chester Lok

Creation Date

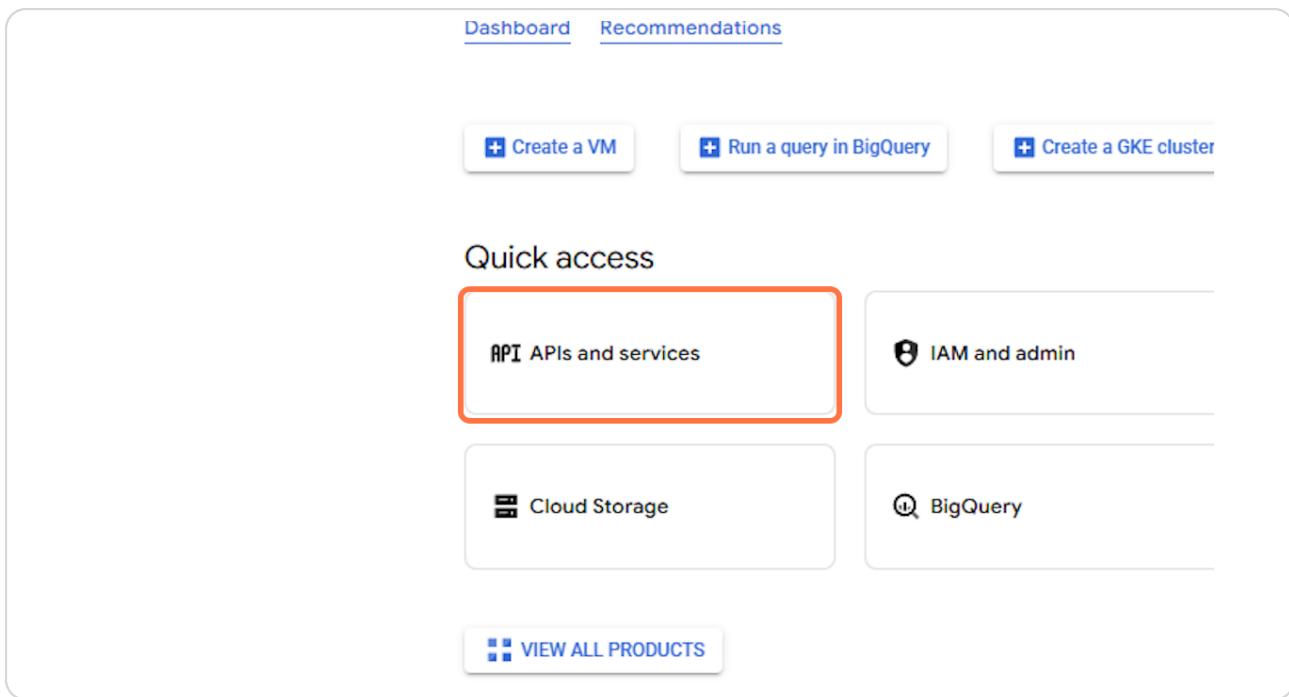
Mar 07, 2025

Last Updated

Mar 07, 2025

STEP 1

Click on APIs and services



STEP 2

Type "google play"

The screenshot shows the 'Welcome to the API Library' page. A search bar at the top contains the query 'google play'. Below the search bar, there is a message: 'The API Library has documentation, links and a smart search experience.' To the right of the search bar is a blue button labeled 'VIEW ALL (24)'. Below the search bar, there are four API cards displayed horizontally:

- for Android
- Maps SDK for iOS
Google
- Maps JavaScript API
Google
- Roads API
Google Enterprise API

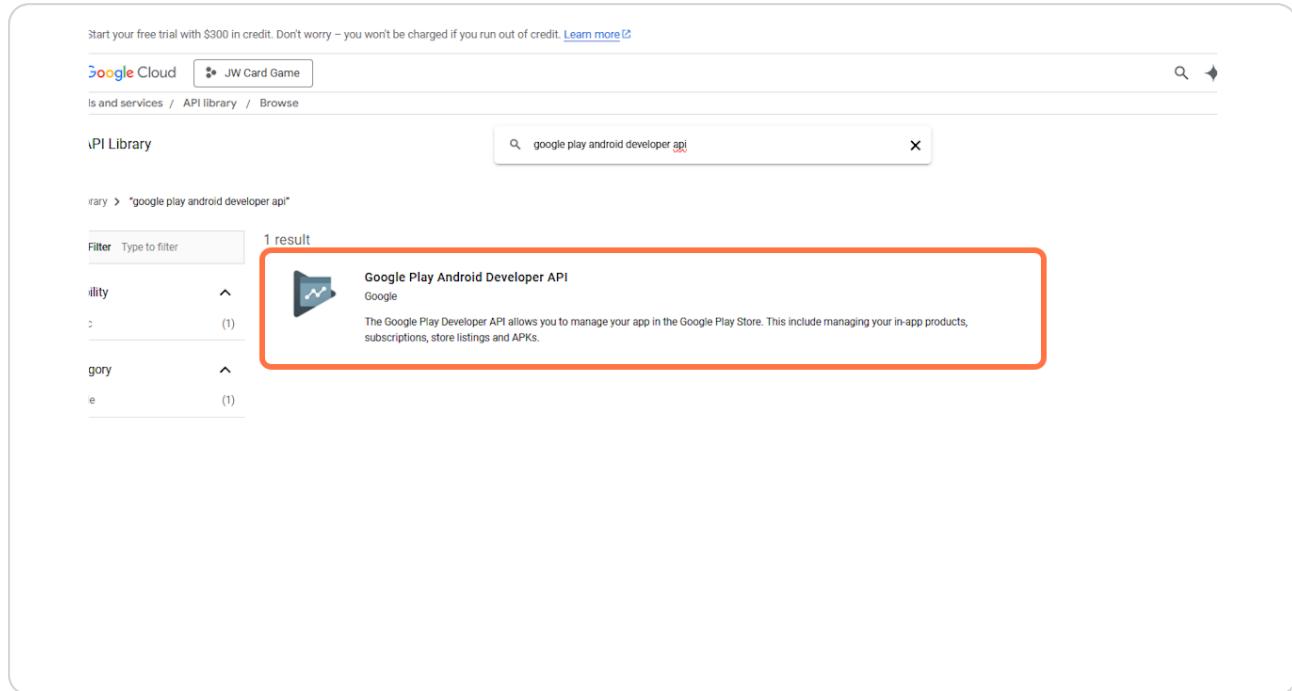
STEP 3

Click on google play android developer api

The screenshot shows the Google Cloud API library interface. At the top, there is a banner with the text 'Start your free trial with \$300 in credit. Don't worry - you won't be charged if you run out of credit.' followed by a 'Learn more' link and a 'Start free' button. Below the banner, the navigation bar includes 'Google Cloud' and 'JW Card Game' under the 'APIs and services' section. The main content area displays the 'Google Play Android Developer API' result, which is highlighted with a red box. The result card includes the API name, provider, and a brief description.

STEP 4

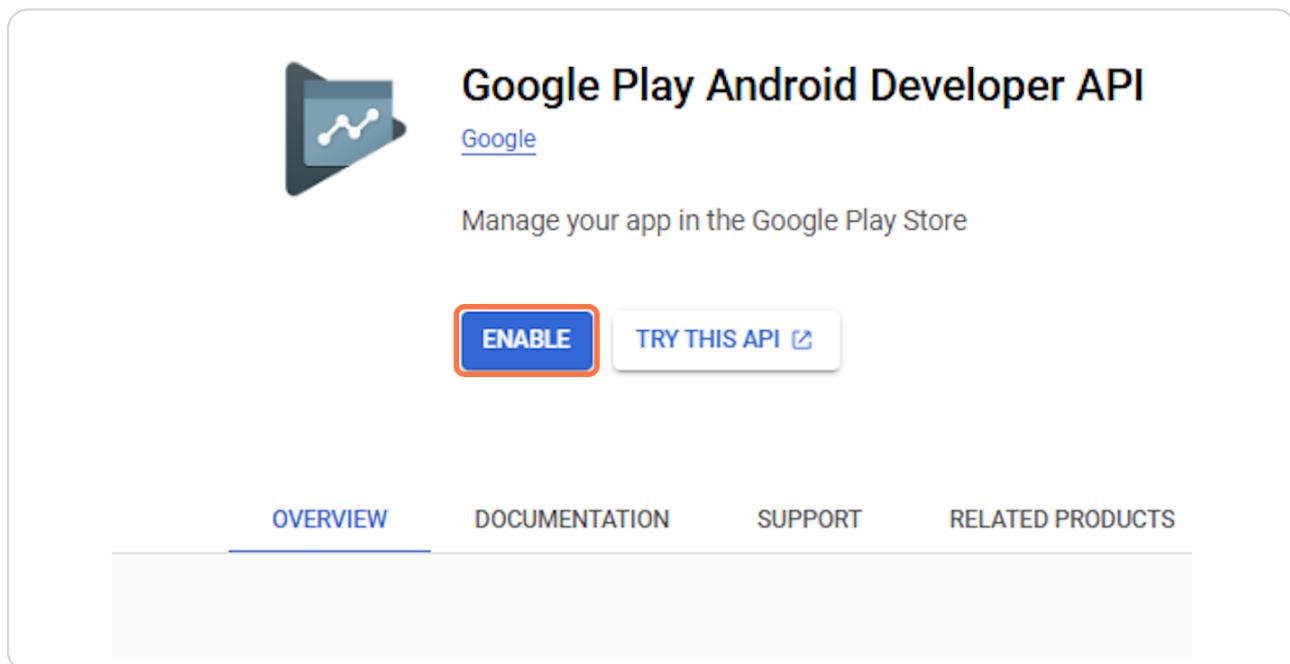
Click on Google Play Android Developer API Google The Google Play Developer API allows you to manage your app in the Google Play Store. This include managing your in-app products, subscriptions, store listings and APKs.



The screenshot shows the Google Cloud API Library interface. At the top, there is a banner offering a free trial with \$300 in credit. Below the banner, the navigation bar includes 'Google Cloud', 'JW Card Game', and a search bar. The search bar contains the query 'google play android developer api'. The main content area displays a search result for '1 result'. The result is titled 'Google Play Android Developer API' and is associated with 'Google'. A small icon of a smartphone with a graph is next to the title. The description below the title reads: 'The Google Play Developer API allows you to manage your app in the Google Play Store. This include managing your in-app products, subscriptions, store listings and APKs.' The entire result card is highlighted with a thick red border.

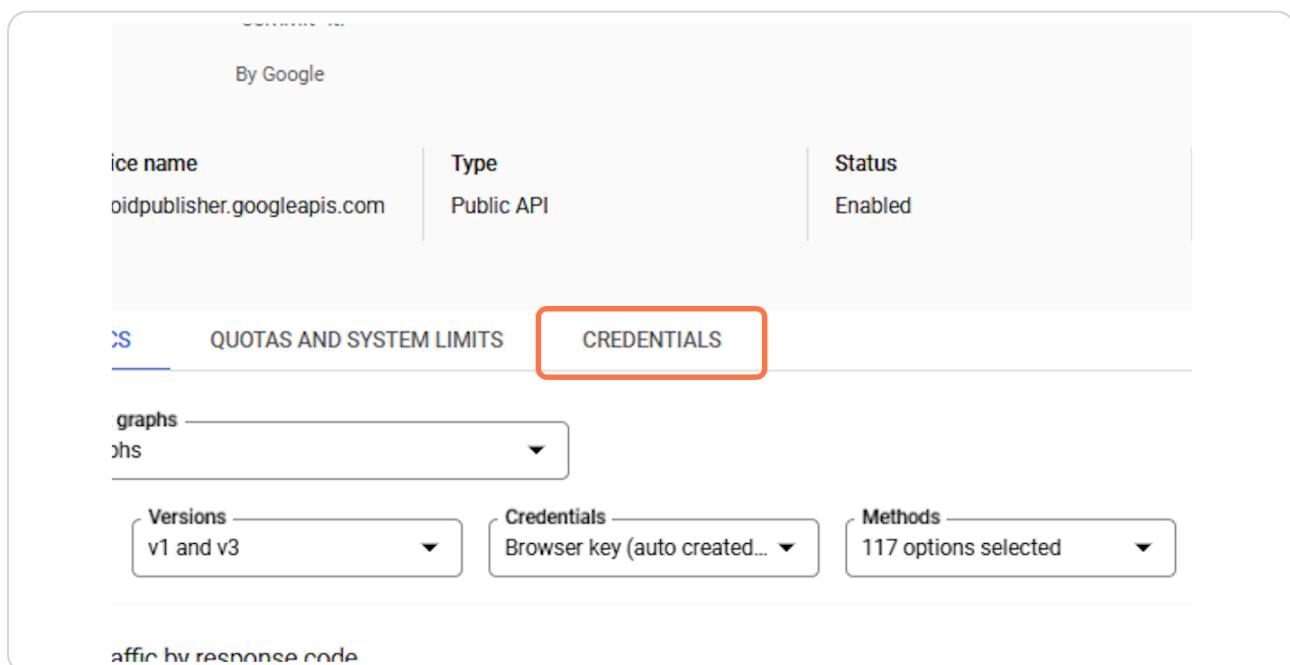
STEP 5

Click on ENABLE



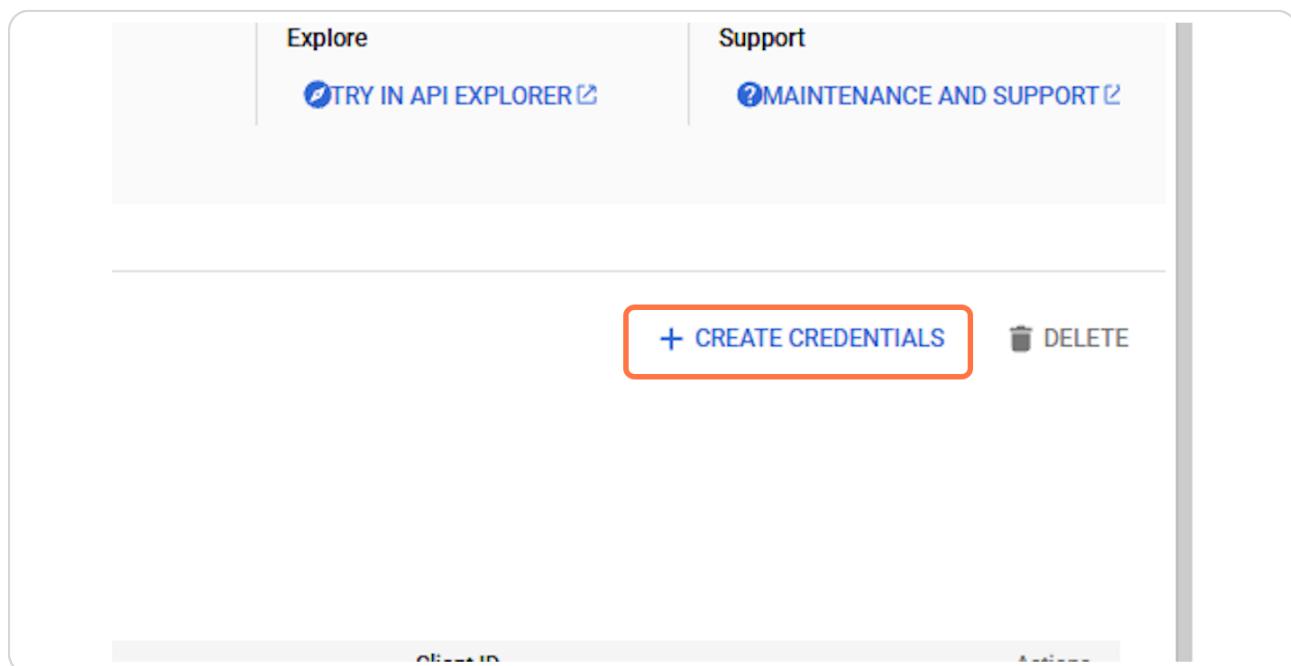
STEP 6

Click on CREDENTIALS



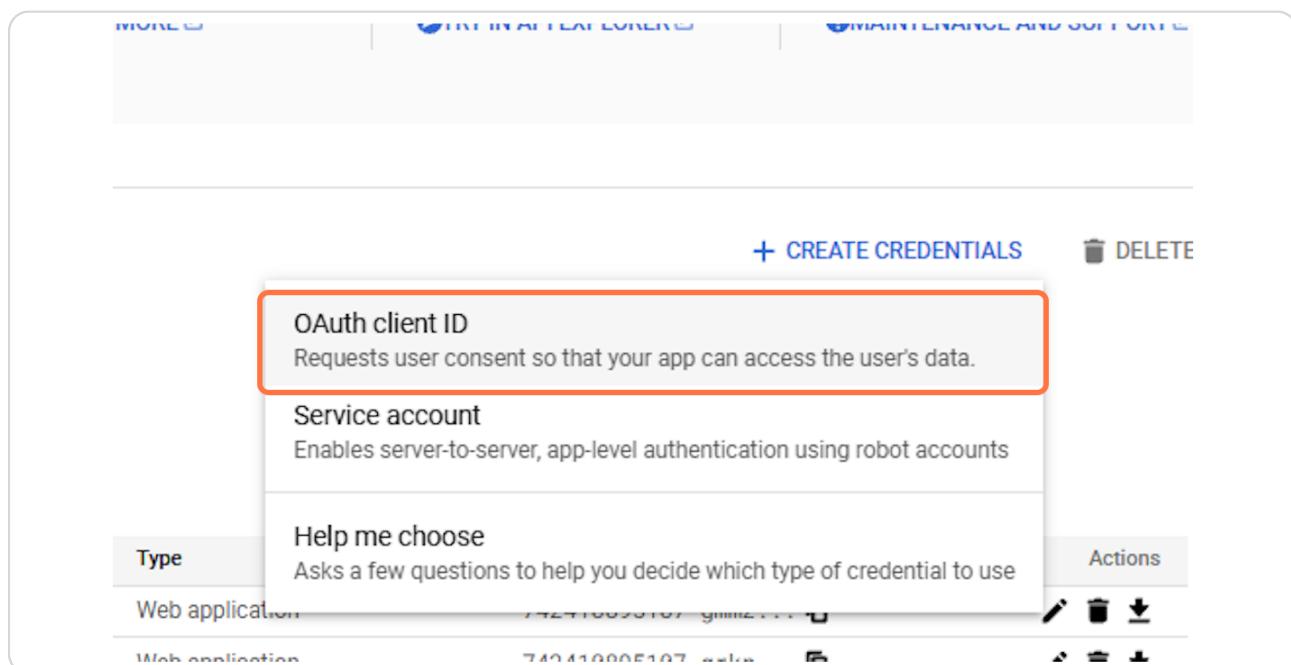
STEP 7

Click on



STEP 8

Click on OAuth client ID...



STEP 9

Click on Application type

The screenshot shows the 'Create OAuth client ID' page in the Google Cloud Platform. On the left, there's a sidebar with links: Overview, Branding, Audience, Clients (which is selected and highlighted in blue), Data access, and Verification centre. The main content area has a heading 'Create OAuth client ID' with a back arrow. Below it is a text block about client IDs and a link to 'Setting up OAuth 2.0'. A dropdown menu labeled 'Application type *' is open, showing several options: Web application (which is selected and highlighted in grey), Android, Chrome extension, iOS, TVs and Limited Input devices, Desktop app, and Universal Windows Platform (UWP). The entire dropdown menu is enclosed in a red border.

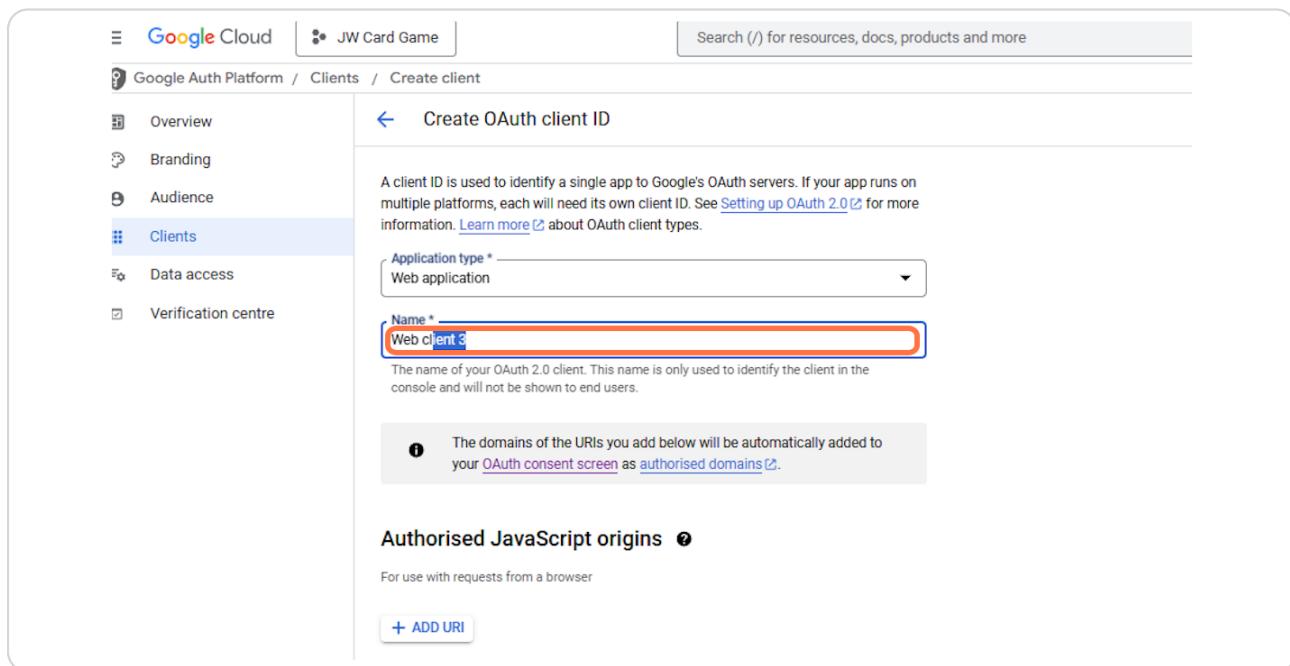
STEP 10

Click on Web application

This screenshot is identical to the one above, showing the 'Create OAuth client ID' page. The sidebar and main content area are the same. The 'Application type *' dropdown menu is open, and the 'Web application' option is clearly visible and highlighted with a blue selection bar around it.

STEP 11

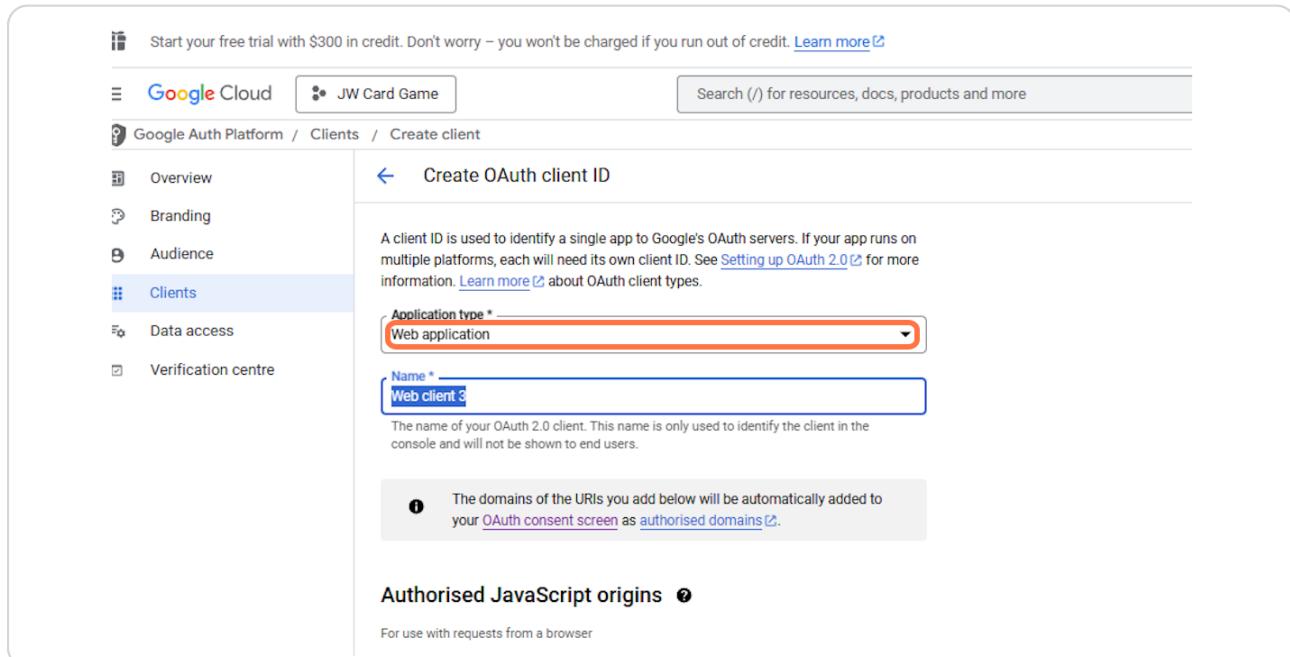
Click on Name



The screenshot shows the 'Create OAuth client ID' page in the Google Cloud Platform. On the left, there's a sidebar with links like Overview, Branding, Audience, Clients (which is selected and highlighted in blue), Data access, and Verification centre. The main area has a back arrow and the title 'Create OAuth client ID'. It explains what a client ID is and provides a link to 'Setting up OAuth 2.0'. Below that is a dropdown for 'Application type' set to 'Web application'. A text input field for 'Name' contains 'Web client 3', which is also highlighted with a red border. A note below says the name is only used for identification. A tooltip indicates that domains added here will be automatically added to the OAuth consent screen. At the bottom, there's a section for 'Authorised JavaScript origins' with a '+ ADD URI' button.

STEP 12

Click on Application type



This screenshot is identical to the one above, showing the 'Create OAuth client ID' page. The 'Clients' link in the sidebar is selected. The 'Application type' dropdown is highlighted with a red border. The 'Name' field contains 'Web client 3'. The 'Authorised JavaScript origins' section is also present at the bottom.

STEP 13

Type "Google Pay Receipt Verification"

The screenshot shows the 'Create OAuth client ID' page in the Google Cloud Platform. On the left, there's a sidebar with links like Overview, Branding, Audience, Clients (which is selected and highlighted in blue), Data access, and Verification centre. The main area has a back arrow and the title 'Create OAuth client ID'. It explains what a client ID is and provides a link to 'Setting up OAuth 2.0'. Below that, there's a dropdown for 'Application type' set to 'Web application', and a 'Name' input field containing 'Google Pay Receipt Verification' which is also highlighted with a red box. A note below says the name is only used for identification. There's also a section for 'Authorised JavaScript origins' with a '+ ADD URI' button.

STEP 14

Click on Application type

This screenshot is identical to the one above, showing the 'Create OAuth client ID' page. The 'Clients' tab is still selected in the sidebar. The 'Application type' dropdown is now explicitly highlighted with a red box and shows 'Web application'. The 'Name' field still contains 'Google Pay Receipt Verification'. The rest of the page content, including the description, note about authorised domains, and the 'Authorised JavaScript origins' section, remains the same.

STEP 15

Click on

The screenshot shows the 'Authorised JavaScript origins' section. It includes a header, a description, a text input field containing 'https://www.example.com', and a blue 'ADD URI' button.

Authorised JavaScript origins ?

For use with requests from a browser

URIs 1 *

+ ADD URI

Authorised redirect URIs ?

For use with requests from a web server

STEP 16

Type "http://localhost"

The screenshot shows the 'Authorised JavaScript origins' section after adding 'http://localhost'. It includes a header, a description, a text input field containing 'http://localhost', a 'Delete item' button, and a blue 'ADD URI' button.

Authorised JavaScript origins ?

For use with requests from a browser

URIs 1 * Delete item

+ ADD URI

Authorised redirect URIs ?

For use with requests from a web server

+ ADD URI

STEP 17

Click on Application type

The screenshot shows the 'Create OAuth client ID' page in the Google Cloud Platform. On the left, a sidebar menu includes 'Overview', 'Branding', 'Audience', **Clients** (which is selected), 'Data access', and 'Verification centre'. The main area has a back arrow and the title 'Create OAuth client ID'. A note explains that a client ID identifies the app to Google's servers. Below this, a dropdown labeled 'Application type *' is set to 'Web application'. A text input field for 'Name *' contains 'Google Pay Receipt Verification'. A note below it says the name is only used for identification in the console. A tooltip indicates that added URIs will be shown in the OAuth consent screen. At the bottom, there's a section for 'Authorised JavaScript origins' with a note about browser requests.

STEP 18

Click on

The screenshot shows the 'Authorised redirect URIs' configuration page. It features a large empty area on the left and a right sidebar. The sidebar has a header 'Authorised redirect URIs' with a help icon. Below it is a note about web server requests. A prominent blue button with a '+' icon and the text '+ ADD URI' is centered in the sidebar. At the bottom, there's a note about setting propagation and two buttons: 'CREATE' and 'CANCEL'.

STEP 19

Paste "https://developers.google.com/oauthplayground" into input

The screenshot shows the 'Authorised JavaScript origins' section with one URI listed: 'http://localhost'. Below it is the 'Authorised redirect URIs' section, which contains the URI 'https://developers.google.com/oauthplayground'. A note at the bottom states: 'Note: It may take five minutes to a few hours for settings to take effect'.

Authorised JavaScript origins ?

For use with requests from a browser

URIs 1 *

+ ADD URI

Authorised redirect URIs ?

For use with requests from a web server

URIs 1 * ✖

+ ADD URI

Note: It may take five minutes to a few hours for settings to take effect

STEP 20

Click on CREATE

The screenshot shows the 'Authorised redirect URIs' section with the URI 'https://developers.google.com/oauthplayground' highlighted with a blue border. At the bottom, there are two buttons: 'CREATE' (highlighted with a red border) and 'CANCEL'.

Authorised redirect URIs ?

For use with requests from a web server

URIs 1 *

+ ADD URI

Note: It may take five minutes to a few hours for settings to take effect

CREATE CANCEL

STEP 21

Click on Copy to clipboard:

742410895107-61pjgohm98kgjk3tcu0o9ffcfgfh5ted8.apps.googleusercontent.com

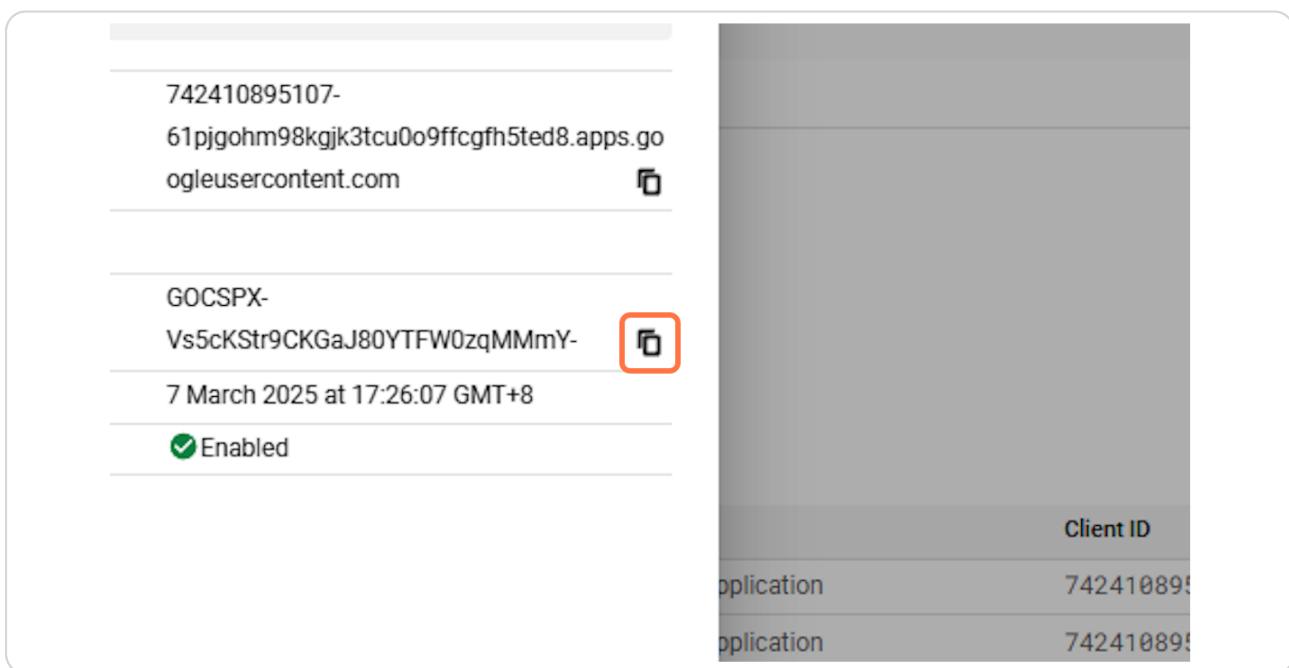
The screenshot shows a list of API endpoints. The first endpoint is highlighted with a red box around its copy icon. The endpoint details are as follows:

- Message: "100 sensitive scope logins until the OAuth verified. This may require a verification process several days."
- Endpoint URL: "742410895107-61pjgohm98kgjk3tcu0o9ffcfgfh5ted8.apps.googleusercontent.com"
- Method: "GET" (indicated by a blue icon)
- Auth: "GOOCSPX-Vs5cKStr9CKGaJ80YTFW0zqMMmY"
- Created: "7 March 2025 at 17:26:07 GMT+8"
- Status: "Enabled" (indicated by a green checkmark icon)

On the right side of the interface, there is a sidebar with "Explore" and a "TRY IN API EXPLORER" button.

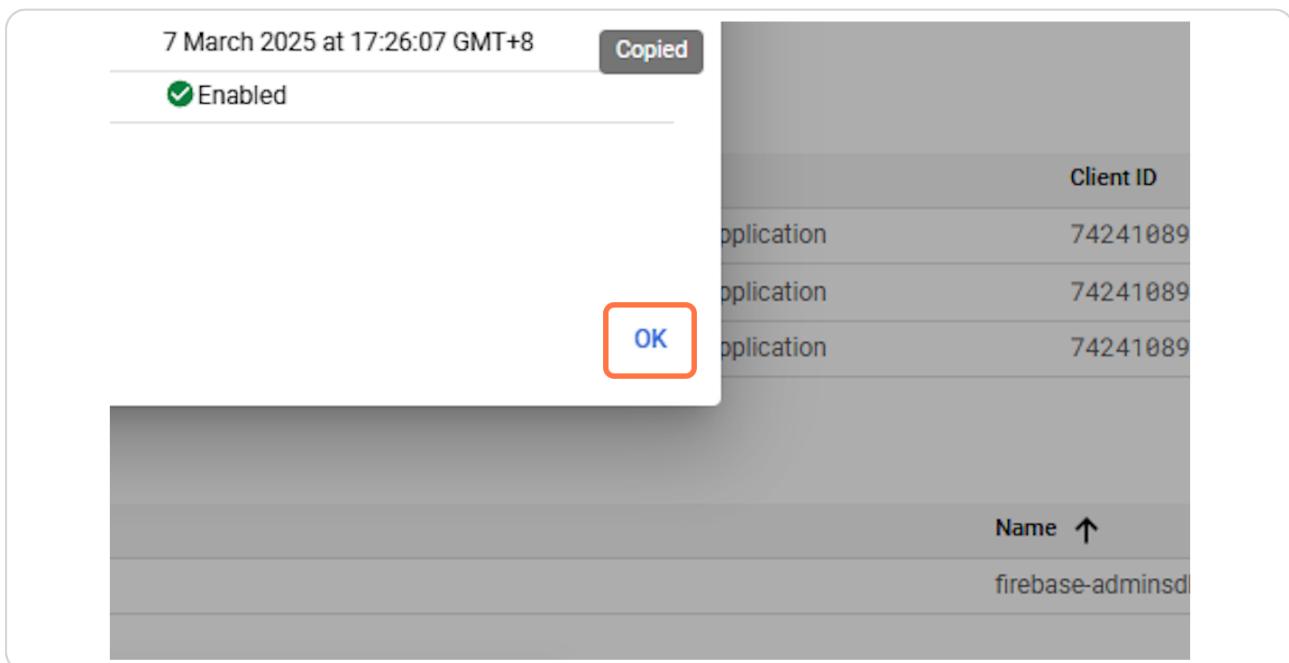
STEP 22

Click on Copy to clipboard: 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



STEP 23

Click on OK



STEP 24

Paste "https://www.googleapis.com/auth/androidpublisher" into input

- [Analytics Hub API v1](#)
- [Analytics Reporting API v4](#)
- [Android Management API v1](#)
- [Apigee API v1](#)
- [Apigee Registry API v1](#)
- [App Engine Admin API v1](#)
- [App Hub API v1](#)
- [Application Integration API v1](#)
- [Apps Script API v1](#)
- [Area120 Tables API v1alpha1](#)
- [Artifact Registry API v1](#)

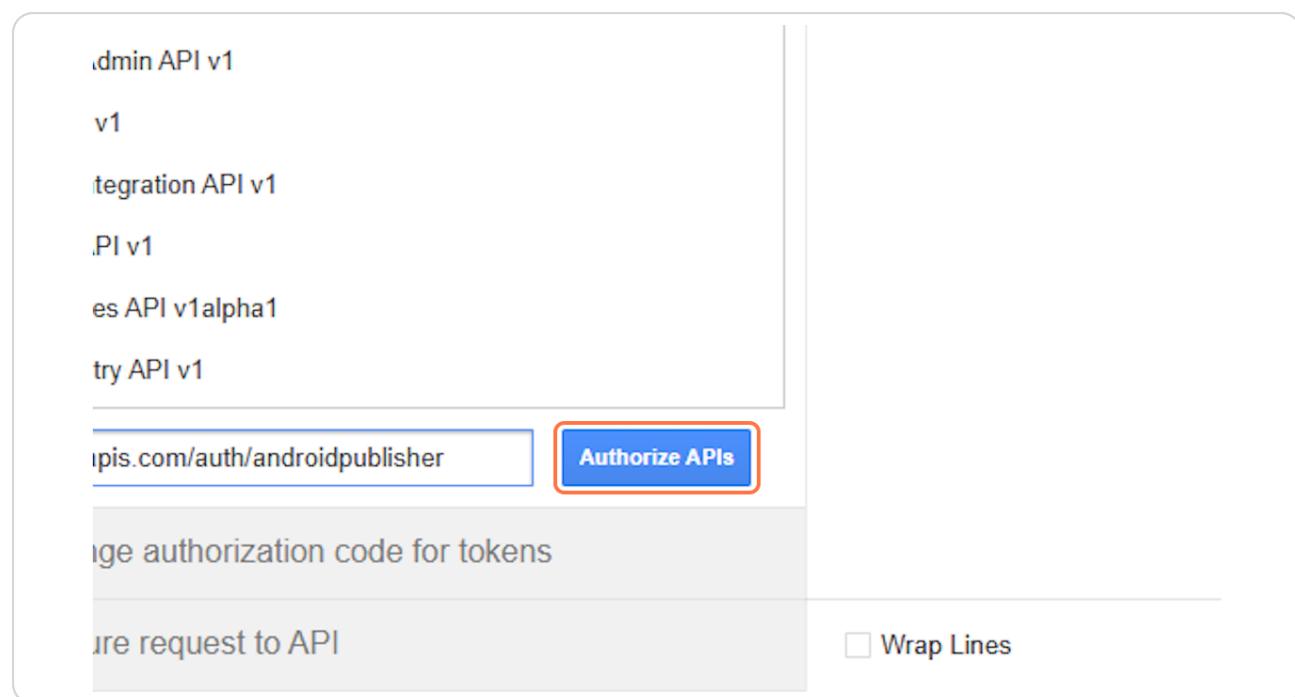
[Authorize APIs](#)

[Step 2 Exchange authorization code for tokens](#)

[Step 3 Configure request to API](#) [Wrap Lines](#)

STEP 25

Click on Authorize APIs



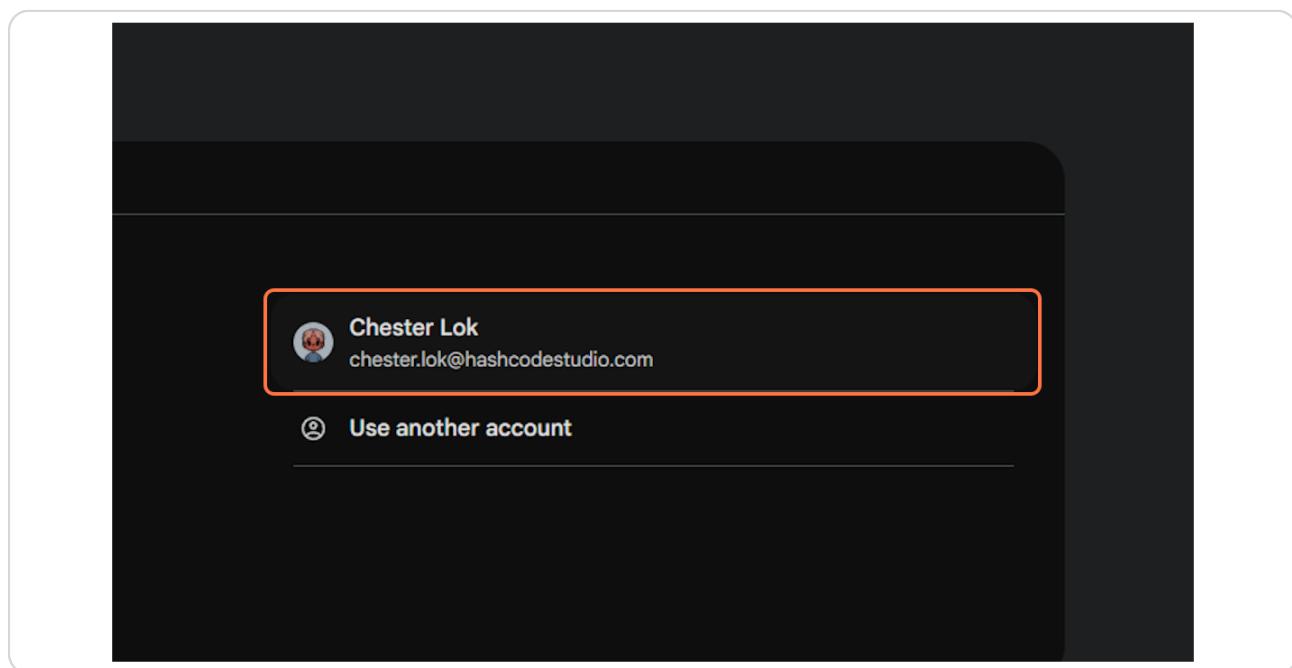
Sign in – Google accounts

2 Steps



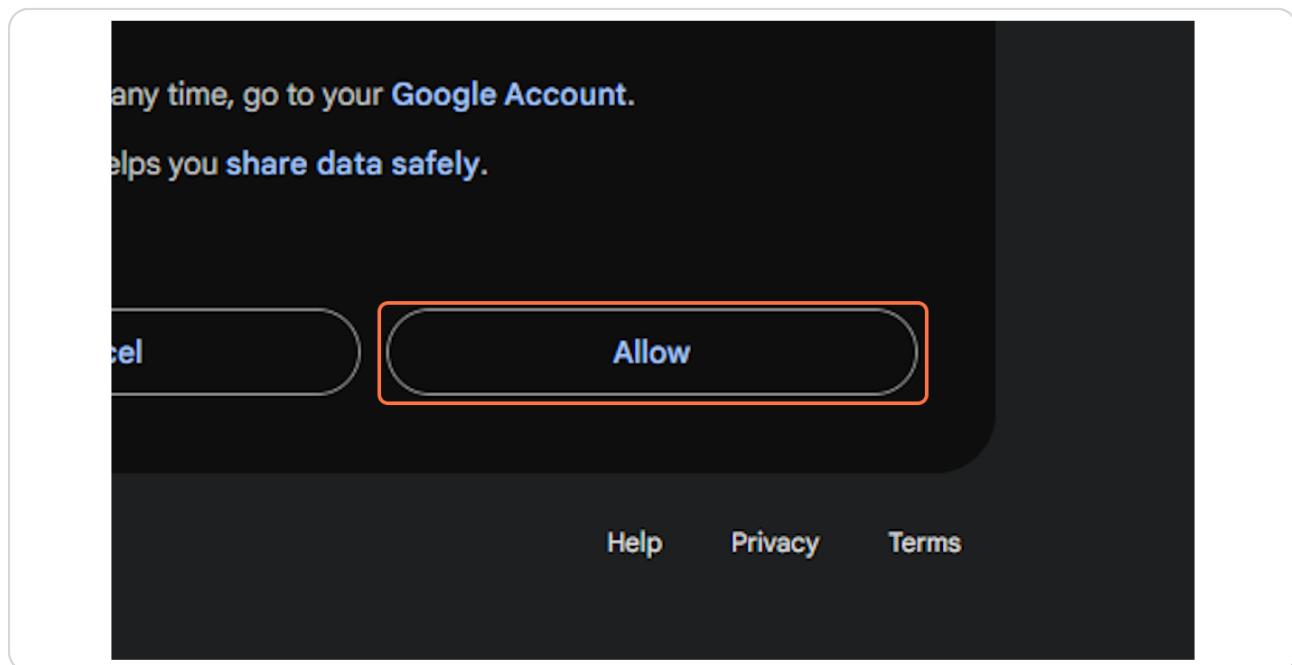
STEP 26

Click on Chester Lok chester.lok@hashcodestudio.com



STEP 27

Click on Allow



STEP 28

Click on Exchange authorization code for tokens

Step 1 Select & authorize APIs

Step 2 Exchange authorization code for tokens

Once you got the Authorization Code from Step 1 click the Exchange authorization code for tokens button, you will get a refresh and an access token which is required to access OAuth protected resources.

Authorization code: 4/0AQSTgQH-b2vclgYBVCyPNuZvmDC9dWFrxvFQhX

Exchange authorization code for tokens

Refresh token: Refresh token

Access token: Access token Refresh access token

Auto-refresh the token before it expires.

Note: The OAuth Playground will automatically revoke refresh tokens after 24h. You can avoid this by specifying your own application OAuth credentials using the Configuration panel.

Request / Response

HTTP/1.1 302 Found
Location: https://accounts.google.com/o/oauth2/v2/auth?code=4/0AQSTgQH-b2vclgYBVCyPNuZvmDC9dWFrxvFQhX&state=4/0AQSTgQH-b2vclgYBVCyPNuZvmDC9dWFrxvFQhX&redirect_uri=https%3A%2F%2Fdevelopers.google.com%2Foauthplayground%2F&scope=https%3A%2F%2Fdevelopers.google.com%2Fapi%2Fuserinfo%2Femail

GET /oauthplayground/?code=4/0AQSTgQH-b2vclgYBVCyPNuZvmDC9dWFrxvFQhX
Host: developers.google.com

STEP 29

Copy input

▼ Step 2 Exchange authorization code for tokens

Once you got the Authorization Code from Step 1 click the Exchange authorization code for tokens button, you will get a refresh and an access token which is required to access OAuth protected resources.

Authorization code: 4/0AQSTgQH-b2vcIgYBVCyPNuZvmDC9dWFrxvFQhX

Exchange authorization code for tokens

Refresh token: **1//045ZhMG4mrzYRCqYIARAAGAQSNwF-L9lrgW1xB2o_d**

Access token: ya29.a0AeXRpp78Ba9haOlUSwc4o | Refresh access token

Auto-refresh the token before it expires.

The access token will expire in **3560** seconds.

Note: The OAuth Playground will automatically revoke refresh tokens after 24h. You can avoid this by specifying your own application OAuth credentials using the Configuration panel.

POST /token HTTP/1.1
Host: oauth2.googleapis.com
Content-length: 261
content-type: application/x-www-form-urlencoded
user-agent: google-oauth-playground

code=4%2F0AQSTgQH-b2vcIgYBVCyPNuZvmDC9dWFrxvFQhX&refresh_token=1//045ZhMG4mrzYRCqYIARAAGAQSNwF-L9lrgW1xB2o_d&client_id=*****&client_secret=*****&grant_type=authorization_code

HTTP/1.1 200 OK
Content-length: 521
X-xss-protection: 0
X-content-type-options: nosniff
Transfer-encoding: chunked
Expires: Mon, 01 Jan 1990 00:00:00 GMT
Vary: Origin, X-Origin, Referer
Server: scaffolding on HTTPServer2
Content-Encoding: gzip
Pragma: no-cache
Cache-control: no-cache, no-store, max-age=0,
Date: Fri, 07 Mar 2025 09:27:09 GMT
X-frame-options: SAMEORIGIN
Alt-svc: h3=":443"; ma=2592000,h3-29=":443";
Content-type: application/json; charset=utf-8

{
 "access_token": "ya29.a0AeXRpp78Ba9haOlUSwc4oUaoBAId_OReT2VnjdbQknEfmmhRrRwXoD1HhwHfurWbxhOxJqzaCgYKAX8SAR",
 "refresh_token_expires_in": 604799,
 "token_type": "Bearer",
 "scope": "https://www.googleapis.com/auth/calendar.readonly",
 "expires_in": 3560
}

Tango

Never miss a step again. Visit Tango.ai