**OAuth2:** Open Authorization version-2

OAuth2 is an authorization framework that allows third party application to access the user’s resources without exposing sensitive information (username & password).

It enables users to share specific information/data with an application while keeping their credentials.

**Roles/Actor:**

1. **Resource Owner:**  Website user(LeetCode User) who has capability to grant to access to his resource.
2. **Client:** The third party application (LeetCode App) wanting to access user’s profile.
3. **Authorization Server:** The sever that authenticate Use and issues tokens
4. **Resource Server:** The server that hold user information and provide access to it based on token.

**Let’s take an example of LeetCode app login page:**

1. Users clicked on Leet Code website url, which will provide an option to login with google/facebook/github 🡪 based on OAuth2
2. LC 🡪 Auth Server 🡺 Auth Code
3. LC 🡪 google endpoint(server end point) 🡺 Access Token/ Refresh Token
4. LC will use access token 🡪 Google(user’s) profile endpoint 🡺 Get required info of user and complete the sign-up/sign-in.

**Steps involved in the OAuth2 Process (Example: Google Auth):**

1. Click on login with google.
2. Google login page which will internally hold {“client\_id”,”scope”,etc.}
3. Login to google
4. Approve/Click consent
5. Auth code generated.
6. Front End will receive the Auth Code.
7. Front end will send the **auth code** to back end.
8. Backend will send the **auth code & secret** to google
9. Backend will receive the **Access Token.**
10. Backend will **validate** tokens to grant access to users.

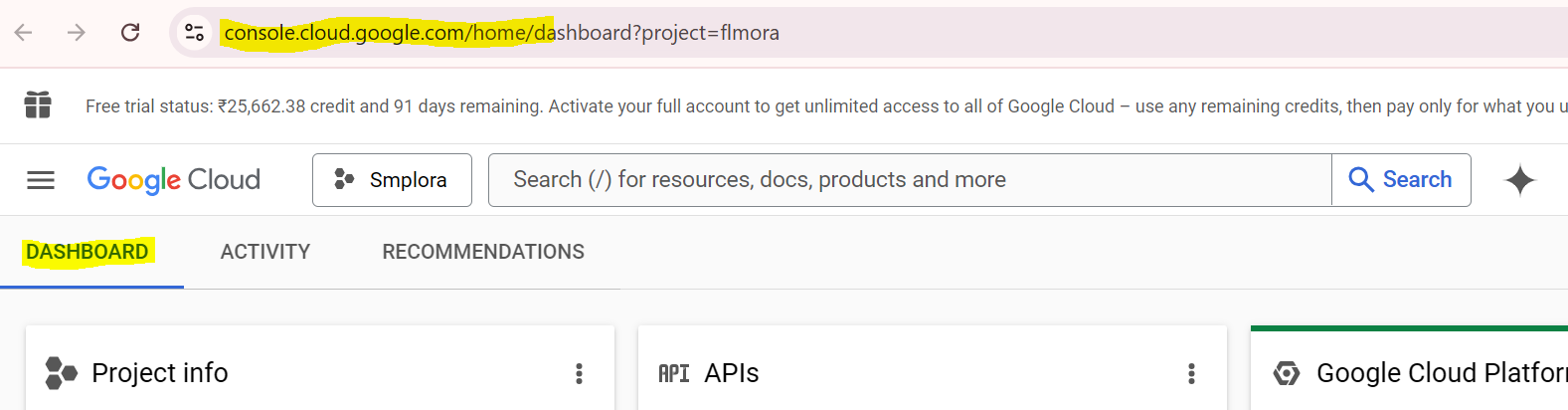
**Note:**

1. At the moment let’s use <https://developers.google.com/oauthplayground> as a frontend to get the Auth Code.
2. Later this code will be used in a controller endpoint manually to get the access token.
3. Once the Front end App is Ready, this point-1 &2 can be achieved through the front end itself, no need of creating the manual work.

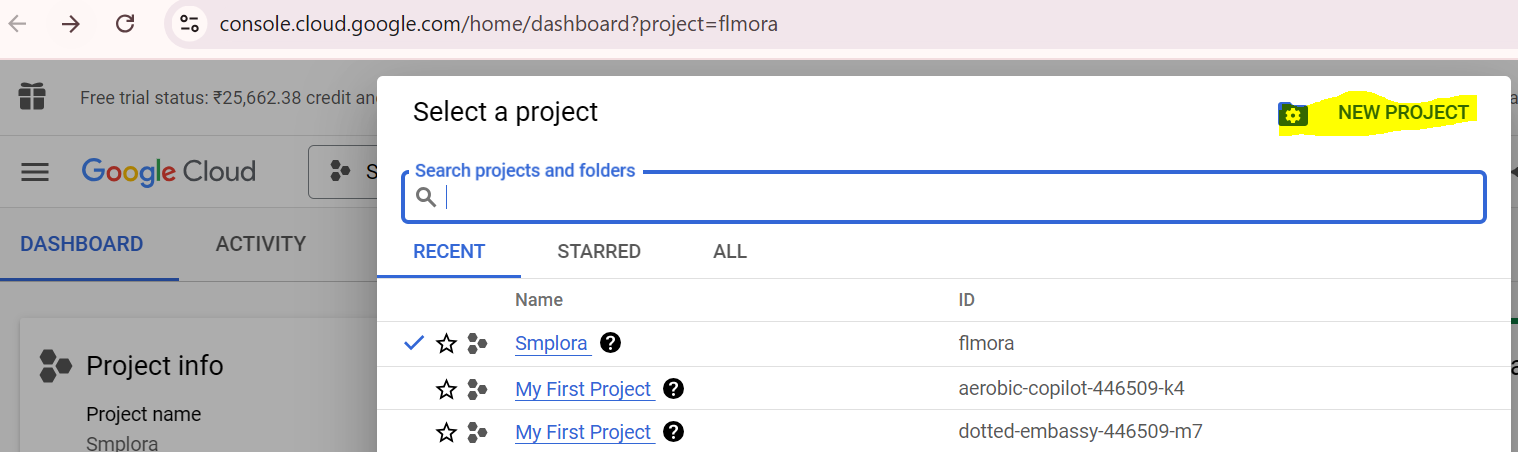
**Google OAuth:**

If we have to use google OAuth, we need to follow thw following steps:

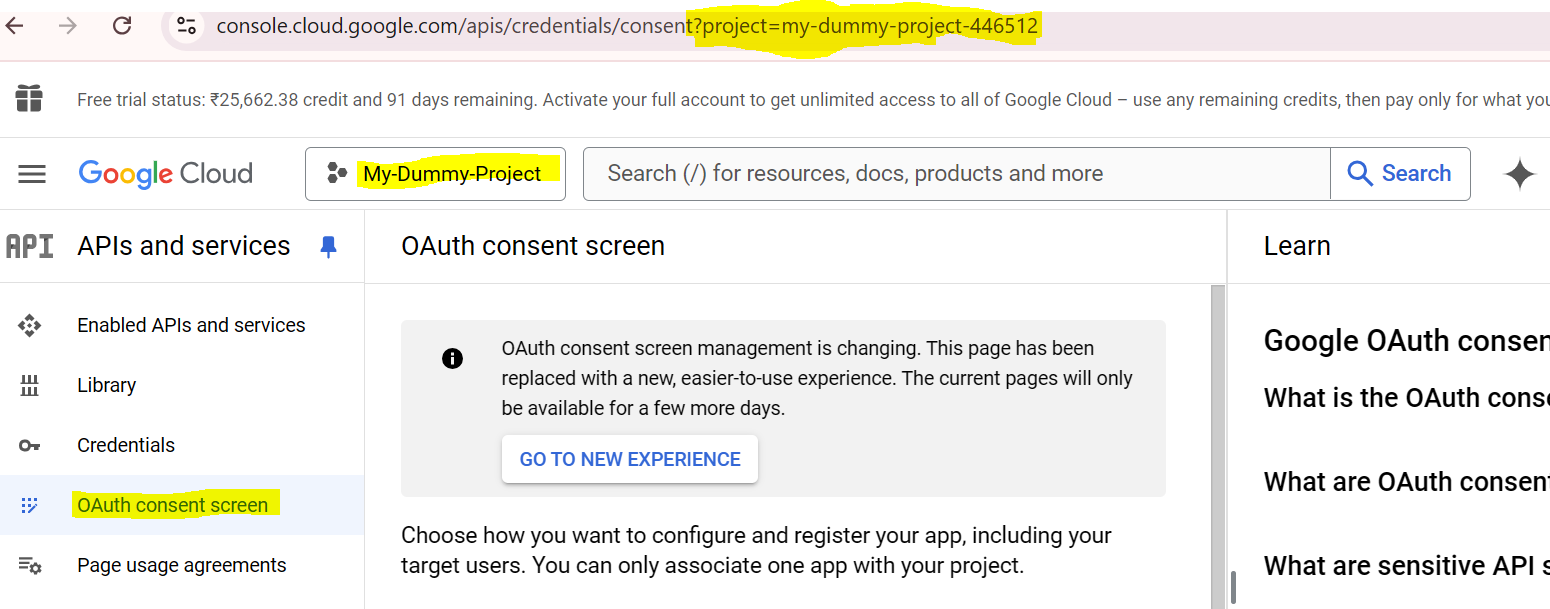
1. To interact with google Auth the third party application need to be registered with the google OAuth.
2. Create a google cloud console account and jump to the page.



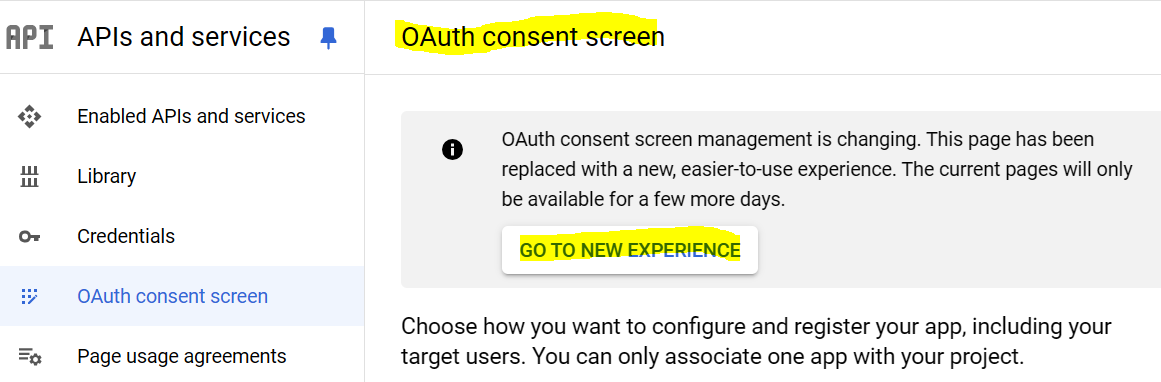
1. Create a Project.



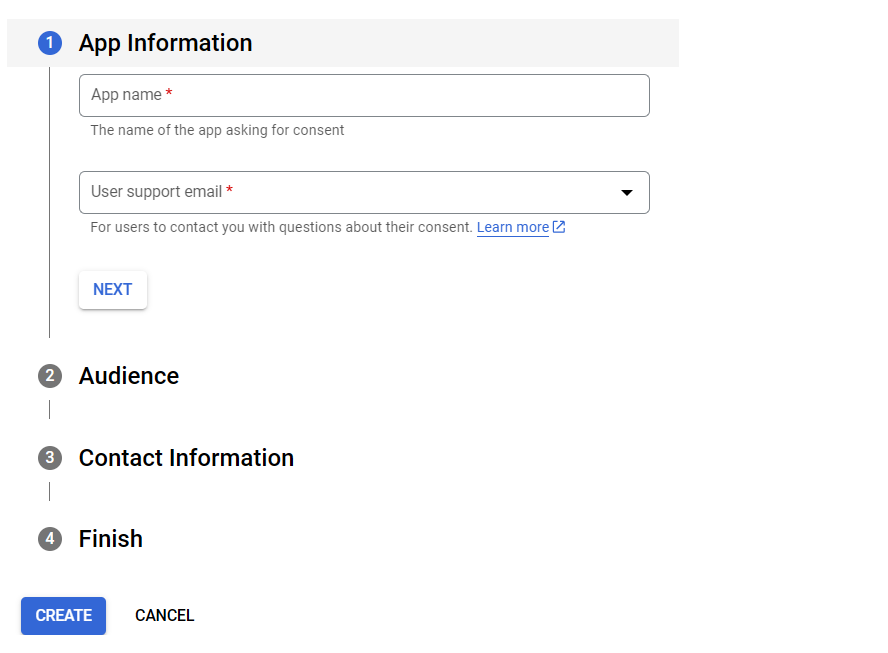
1. Once the project is created, select the project and under left panel Go To **OAuth Consent Screen.**



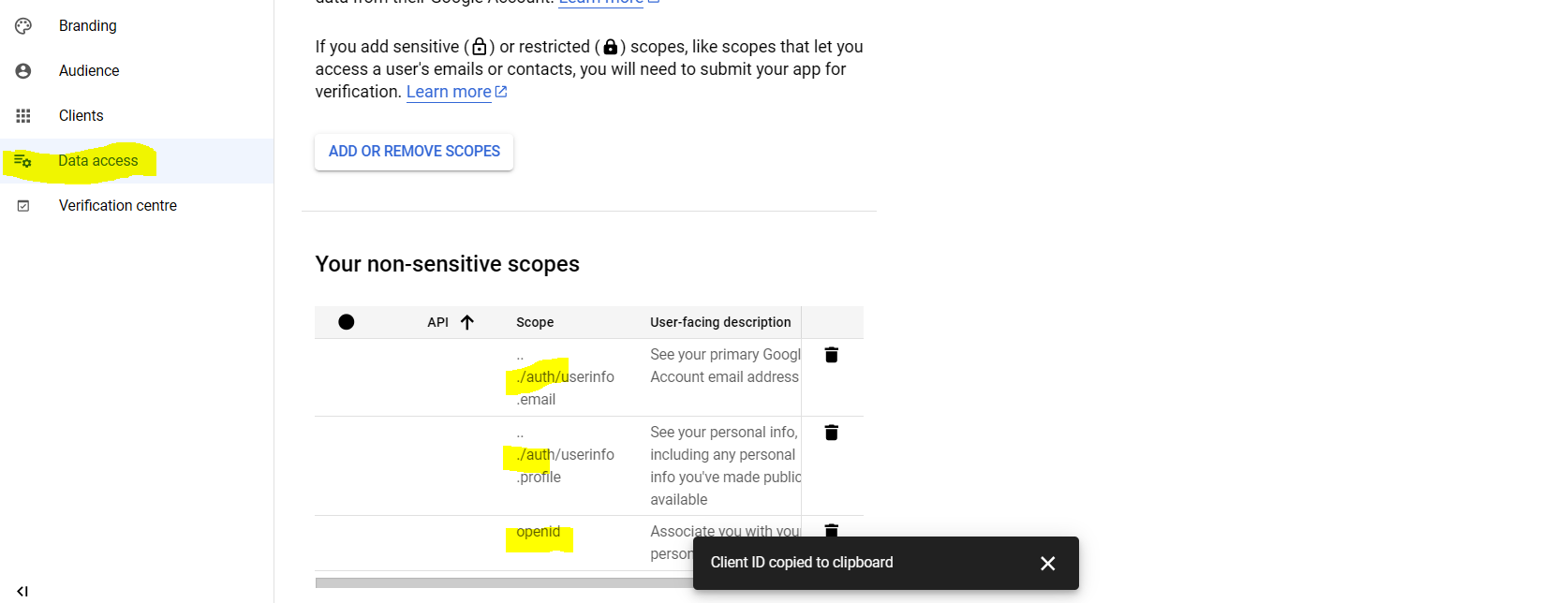
1. Go to new experience and click on **Get Started** button.

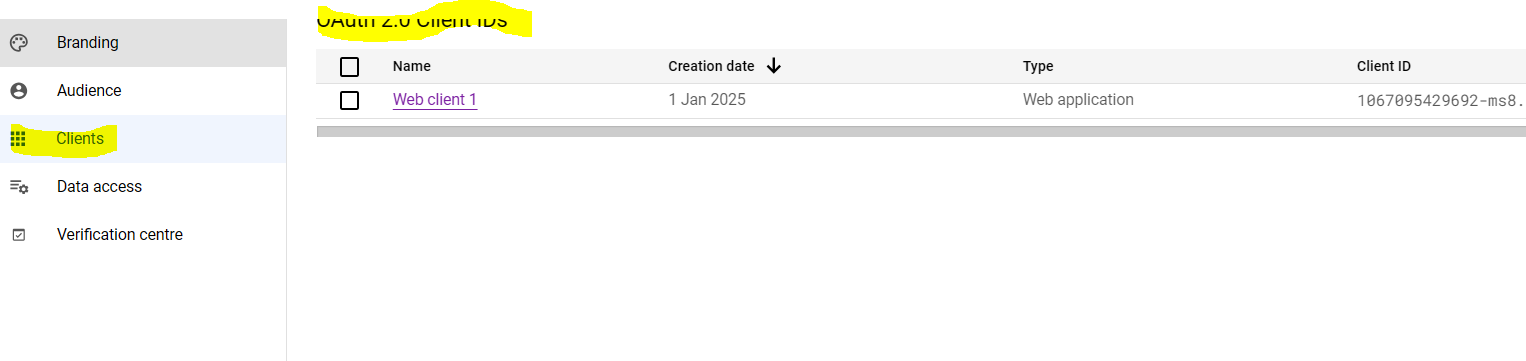
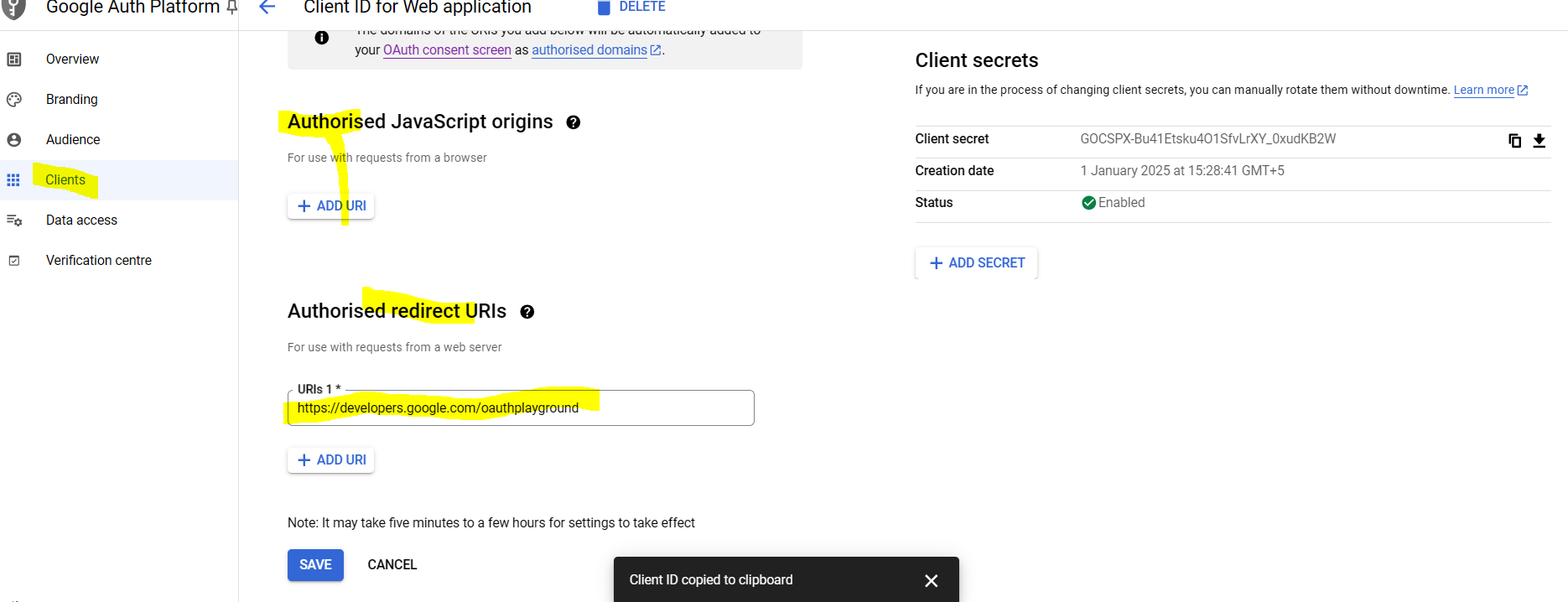


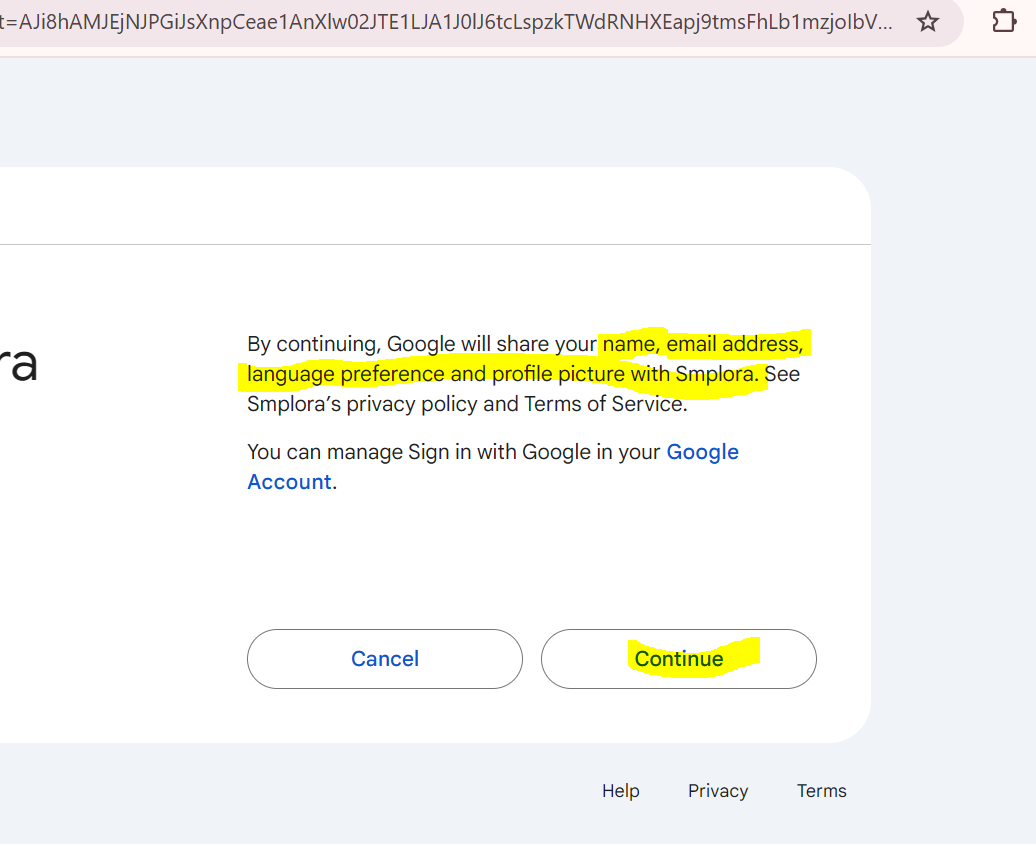
1. Provide all the information on the dialog box and create.



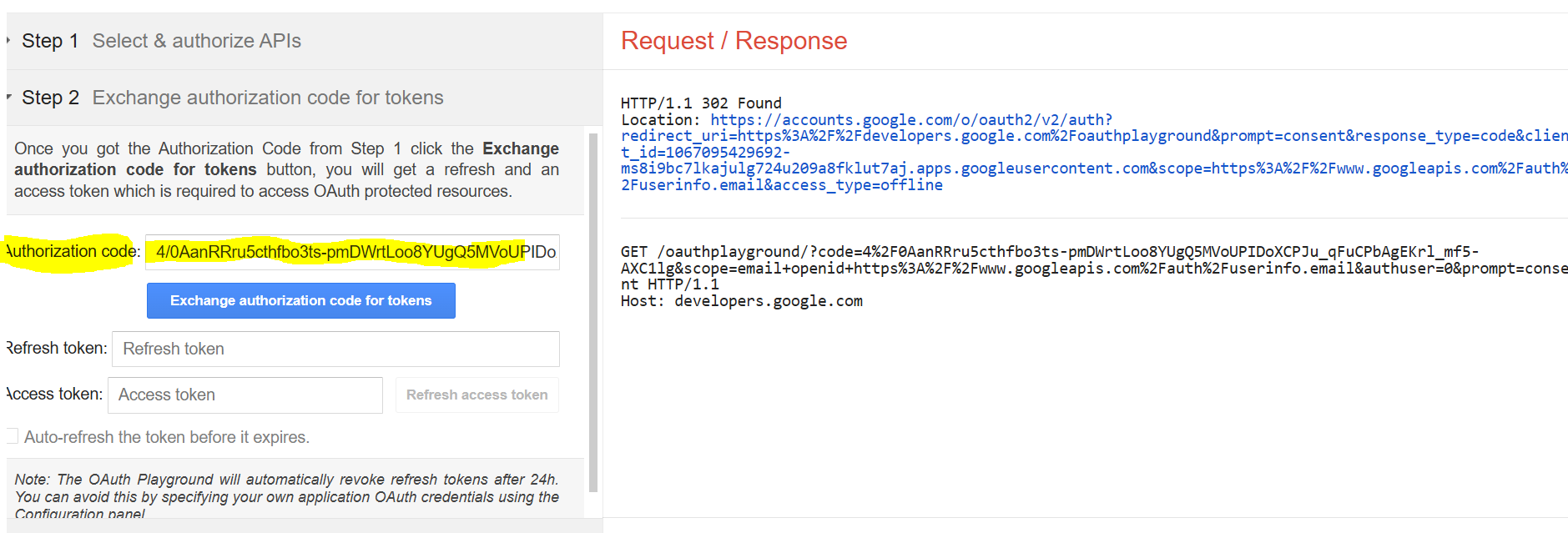
1. Go to under **Data Access** section and provide all the three or required scopes.



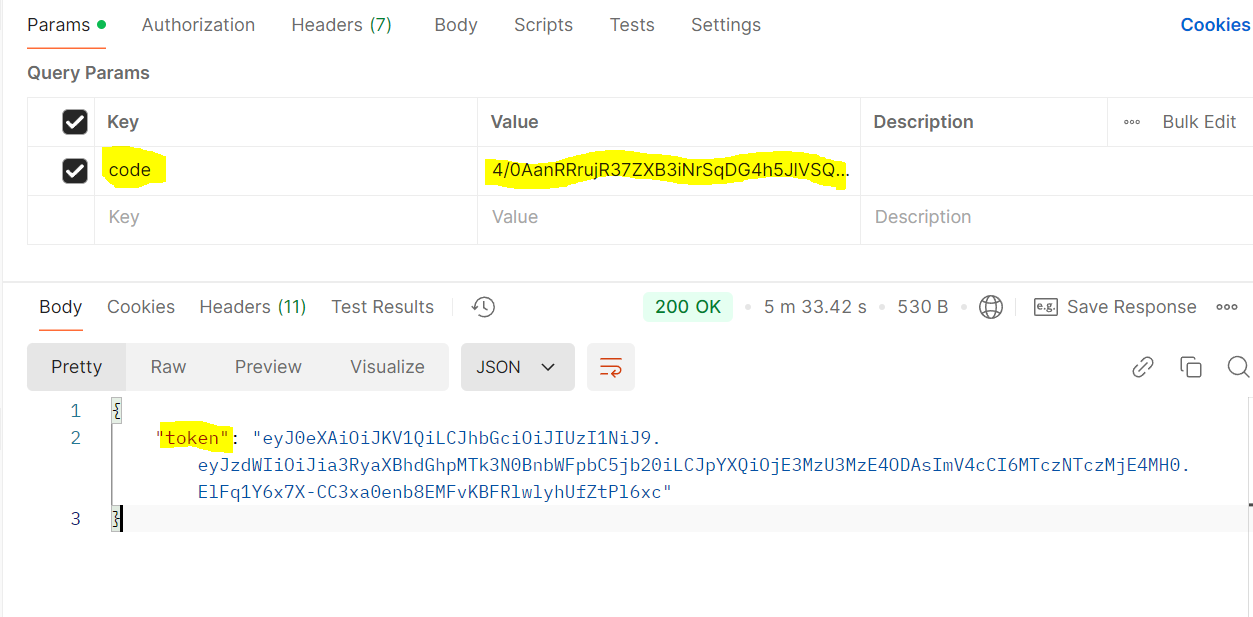
1. Under Client section provide all the necessary information 
2. 
3. Go to <https://developers.google.com/oauthplayground> page and choose scope as <https://www.googleapis.com/auth/userinfo.email>
4. Click on **Authorize API** and you will be landing google login page where you need to confirm the consent.



1. After clicking on **Continue**  you will be receiving the **Auth Code.**



1. You can use this **Auth Code,** along with Secrets to trigger the google token api to receive the **Access Token**.



1. And Now this token will be used to for communicating the required endpoints.