

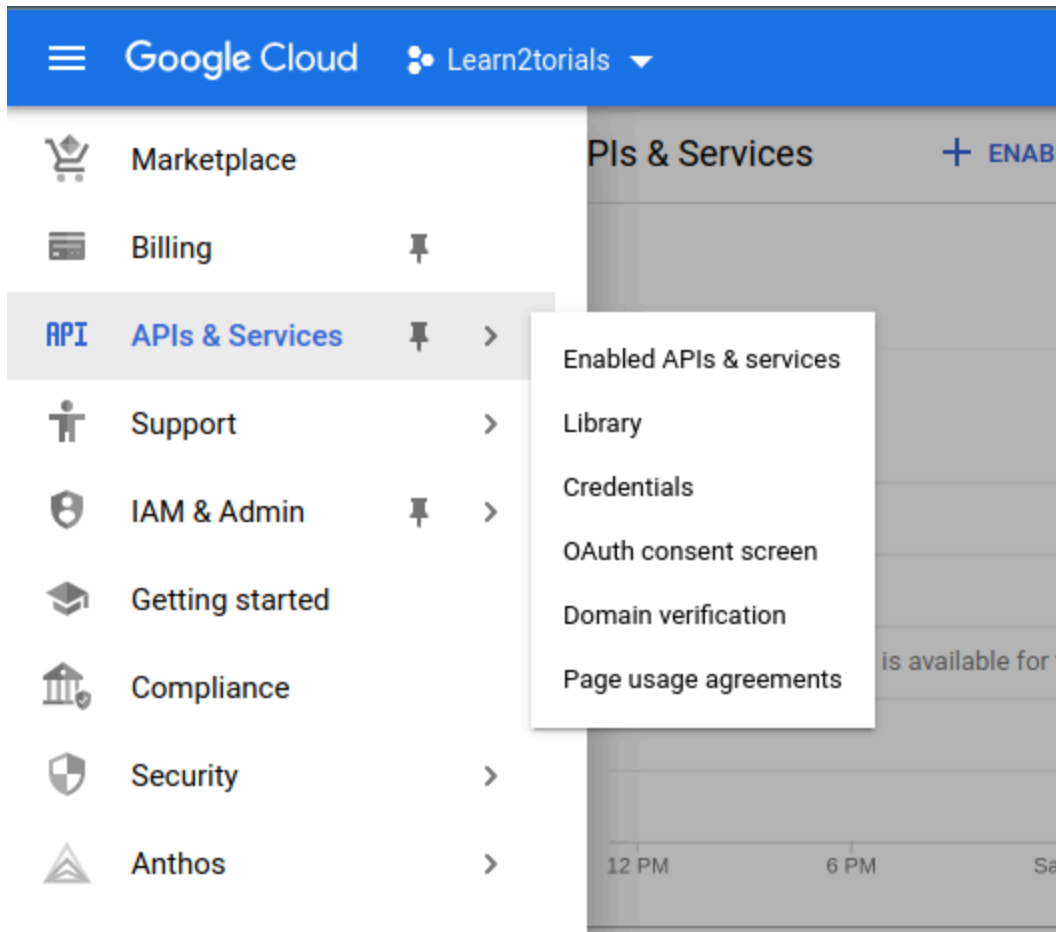
How to integrate Google login using your Gmail and connect it to a MySQL database in an app you're building with Eclipse and using Github.

1. Create OAuth Credentials

1. Go to Google Cloud Console
2. Create a project > Enable **Google+ API**
3. Go to "Credentials" > **Create OAuth client ID**
Choose **Web application**
Add a redirect URI (e.g., <http://localhost:8080/oauth2callback>)
4. Save **Client ID** and **Client Secret**

Step-1: Navigate to APIs & Services tab in google cloud

First step is to login to google cloud using the link provided above.
Soon as you log in click on the left navigation bar as shown in below screen and find APIs & Services menu item.



Step-2: Find API & Services menu item

Once you find the **APIs & Services** menu item click on **credentials** sub menu item. You will be brought to screen shown below:

API APIs & Services

Enabled APIs & services

Library

Credentials

OAuth consent screen

Domain verification

Page usage agreements

Credentials

+ CREATE CREDENTIALS

DELETE

Create credentials to access your enabled APIs. [Learn more](#)

Remember to configure the OAuth consent screen with information about your application.

API Keys

<input type="checkbox"/>	Name	Creation date	↓
No API keys to display			

OAuth 2.0 Client IDs

<input type="checkbox"/>	Name	Creation date	↓
No OAuth clients to display			

Service Accounts

<input type="checkbox"/>	Email	Name	↑
No service accounts to display			

Step-3: Configure Consent Screen

Next, click on the **CREATE CREDENTIALS** button as shown below and then click on OAuth client ID.

Google Cloud Learn2torials

Search Products, resources

APIs & Services

- Enabled APIs & services
- Library
- Credentials**
- OAuth consent screen
- Domain verification
- Page usage agreements

Credentials + CREATE CREDENTIALS DELETE

Create credentials to access your project

API Keys

☐ Name

No API keys to display

OAuth 2.0 Client IDs

☐ Name Creation date ↓

No OAuth clients to display

Service Accounts

☐ Email

No service accounts to display

API key
Identifies your project using a simple API key to check quota and access

OAuth client ID
Requests user consent so your app can access the user's data

Service account
Enables server-to-server, app-level authentication using robot accounts

Help me choose
Asks a few questions to help you decide which type of credential to use

Once you click on OAuth client ID for the first time it will ask you to configure the consent screen as shown below:

APIs & Services









← Create OAuth client ID

A client ID is used to identify a single app to Google's OAuth servers. If your app runs on multiple platforms, each will need its own client ID. See [Setting up OAuth 2.0](#) for more information. [Learn more](#) about OAuth client types.

⚠ To create an OAuth client ID, you must first configure your consent screen [CONFIGURE CONSENT SCREEN](#)

- Enabled APIs & services
- Library
- Credentials**
- OAuth consent screen
- Domain verification
- Page usage agreements

Click on Configure Consent Screen button and you will be brought to following screen:

API APIs & Services	OAuth consent screen
<div><div> Enabled APIs & services</div><div> Library</div><div> Credentials</div><div> OAuth consent screen</div><div> Domain verification</div><div> Page usage agreements</div></div>	<p>Choose how you want to configure and register your app, including your target users. You can only associate one app with your project.</p> <h3>User Type</h3> <div><input type="radio"/> Internal </div> <p>Only available to users within your organization. You will not need to submit your app for verification. Learn more about user type</p> <div><input checked="" type="radio"/> External </div> <p>Available to any test user with a Google Account. Your app will start in testing mode and will only be available to users you add to the list of test users. Once your app is ready to push to production, you may need to verify your app. Learn more about user type</p> <div><div>CREATE</div></div> <p>Let us know what you think about our OAuth experience</p>

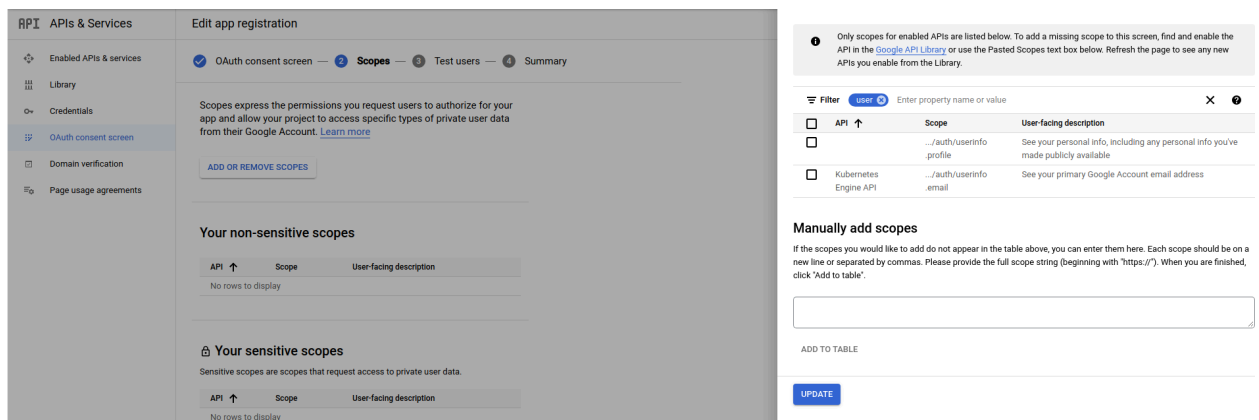
Click on the External option so that all google users can use this application. Click on create button you will be brought to OAuth consent screen as shown below:

API	APIs & Services	Edit app registration
<div> <div>✦</div> <div>Enabled APIs & services</div> </div> <div> <div>≡</div> <div>Library</div> </div> <div> <div>🔑</div> <div>Credentials</div> </div> <div> <div>🔗</div> <div>OAuth consent screen</div> </div> <div> <div>☑</div> <div>Domain verification</div> </div> <div> <div>⚙</div> <div>Page usage agreements</div> </div>	<div> <div>1</div> <div>OAuth consent screen</div> </div> <div> <div>2</div> <div>Scopes</div> </div> <div> <div>3</div> <div>Test users</div> </div> <div> <div>4</div> <div>Summary</div> </div>	<h3>App information</h3> <p>This shows in the consent screen, and helps end users know who you are and contact you</p> <div> <div>App name *</div> <div>The name of the app asking for consent</div> </div> <div> <div>User support email *</div> <div>For users to contact you with questions about their consent</div> </div> <div> <div>App logo</div> <div>BROWSE</div> <div>Upload an image, not larger than 1MB on the consent screen that will help users recognize your app. Allowed image formats are JPG, PNG, and BMP. Logos should be square and 120px by 120px for the best results.</div> </div> <h3>App domain</h3> <p>To protect you and your users, Google only allows apps using OAuth to use Authorized Domains. The following information will be shown to your users on the consent screen.</p> <div> <div>Application home page</div> <div>Provide users a link to your home page</div> </div> <div> <div>Application privacy policy link</div> <div>Provide users a link to your public privacy policy</div> </div> <div> <div>Application terms of service link</div> <div>Provide users a link to your public terms of service</div> </div> <div> <div>Authorized domains</div> <div>?</div> </div>

Fill out following important details in OAuth concent screen

- App Name
- User support email
- App Logo

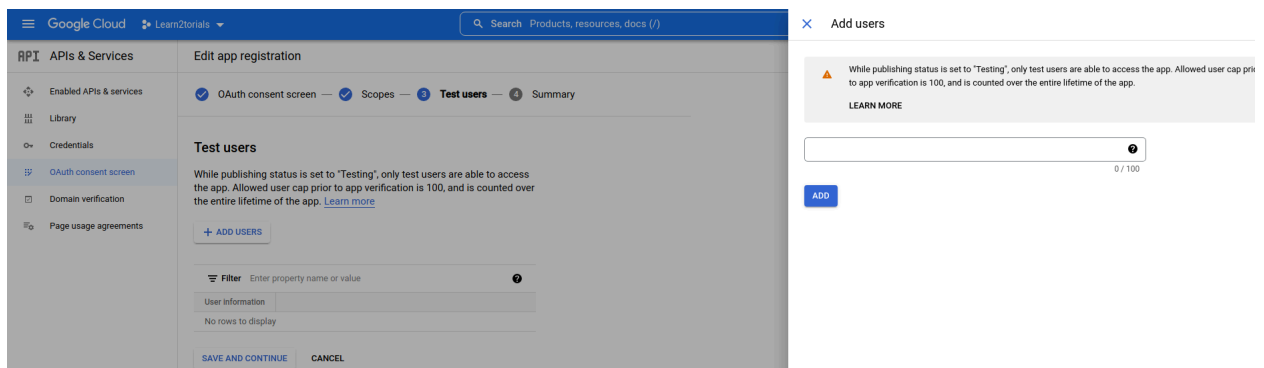
Next, click on save and continue and it will take you to Scopes screen. Click on Add or remove scopes button and it will pop open the screen on the right.



Search for user in the search box as shown below and add:

- /auth/userinfo.profile
- /auth/userinfo.email

select above scopes and hit update button. Click on save and continue button again. You will be taken to Test users screen as shown below: Click on Add users button.



Add user email which you can use for testing purposes of your application. You will not be able to use any other user until your app is in testing mode.

Add email address and then click on save and continue button. It will present a summary page of your application.

Verify the summary and click on back to the dashboard button shown at the end of summary screen.

Step-4: Create OAuth client ID

Now, once you are back on the dashboard screen click on credentials menu item and then click on create credentials button.

The screenshot shows the Google Cloud console interface. The top navigation bar includes the Google Cloud logo and a search bar. The left sidebar is titled 'APIs & Services' and contains a list of options: 'Enabled APIs & services', 'Library', 'Credentials' (which is highlighted), 'OAuth consent screen', 'Domain verification', and 'Page usage agreements'. The main content area is titled 'Credentials' and features a '+ CREATE CREDENTIALS' button and a 'DELETE' button. A dropdown menu is open, showing four options: 'API key' (Identifies your project using a simple API key to check quota and access), 'OAuth client ID' (Requests user consent so your app can access the user's data), 'Service account' (Enables server-to-server, app-level authentication using robot accounts), and 'Help me choose' (Asks a few questions to help you decide which type of credential to use). Below the dropdown, there are three sections: 'API Keys' with a table header 'Name' and a message 'No API keys to display'; 'OAuth 2.0 Client IDs' with a table header 'Name' and 'Creation date' (with a downward arrow) and a message 'No OAuth clients to display'; and 'Service Accounts' with a table header 'Email' and a message 'No service accounts to display'.

This time you will be brought to Create OAuth client ID screen as shown below. Select Web application in Application type dropdown menu provide name of your application and add authorized redirect URIs.

API APIs & Services

← Create OAuth client ID

Enabled APIs & services

Library

Credentials

OAuth consent screen

Domain verification

Page usage agreements

A client ID is used to identify a single app to Google's OAuth servers. If your app runs on multiple platforms, each will need its own client ID. See [Setting up OAuth 2.0](#) for more information. [Learn more](#) about OAuth client types.

Application type *
Web application

Name *
Web client 1

The name of your OAuth 2.0 client. This name is only used to identify the client in the console and will not be shown to end users.

i The domains of the URIs you add below will be automatically added to your [OAuth consent screen](#) as [authorized domains](#).

Authorized JavaScript origins ?

For use with requests from a browser

+ ADD URI

Authorized redirect URIs ?

For use with requests from a web server

+ ADD URI

Note: It may take 5 minutes to a few hours for settings to take effect

CREATE CANCEL

Once you click on the create button it will present a popup with OAuth client information. It will contain following information:

- Your Client ID

■ Your Client Secret


You can download the json file because it won't give you this option again. Save it in a secure place.

Creation date ↓

Restrictions

OAuth client created


The client ID and secret can always be accessed from Credentials in APIs & Services



OAuth is limited to 100 [sensitive scope logins](#) until the [OAuth consent screen](#) is verified. This may require a verification process that can take several days.


Your Client ID


449951191942-ijna0vroflo79n6s935bdtk4bn80adqs.apps.gc



Your Client Secret

G0CSPX-IB3bL_smif3BhNhiqzPEwEPaCBF1



 DOWNLOAD JSON

OK