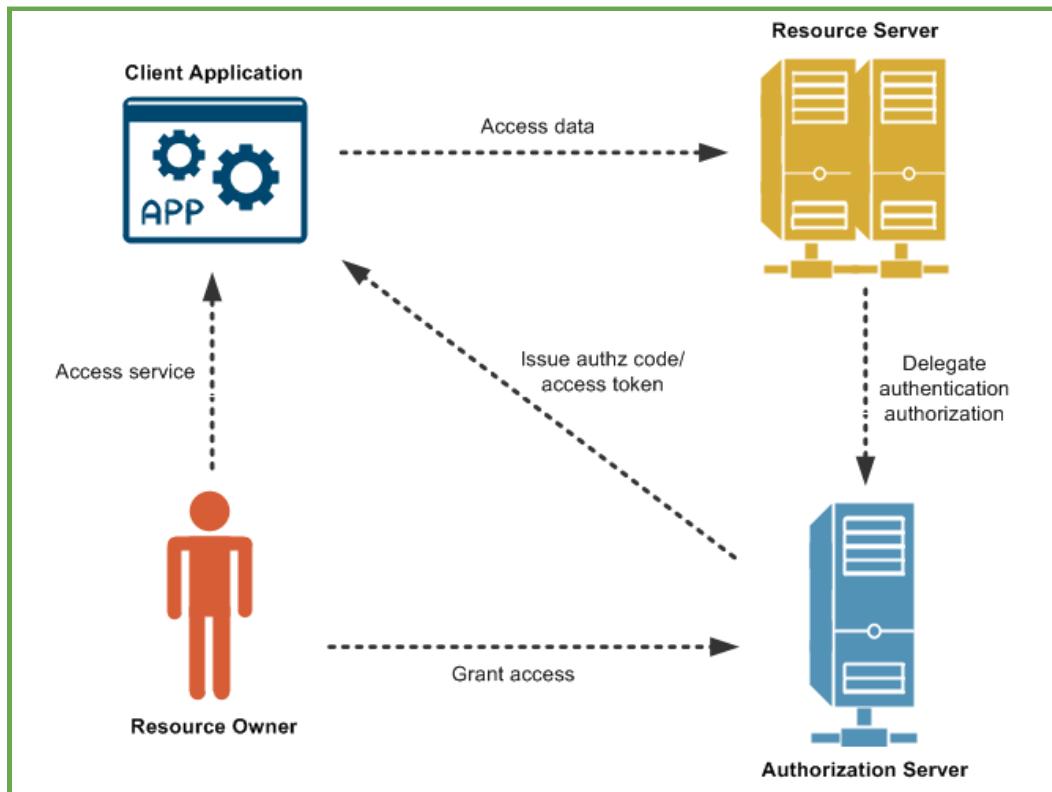




OAuth 2.0 Authorization using API (Gmail)



- What is OAuth 2.0
- OAuth2.0 Roles
 - Resource Owner
 - Client
 - Authorization Server
 - Resource Server
- How Does OAuth 2.0 Work?



Practical Section

Google APIs & Services :

<https://console.cloud.google.com/apis/dashboard?project=api-training-327501>

Step 01: Create New Project

- Click “Select a project” in the Google cloud platform dashboard.

A screenshot of the Google Cloud Platform dashboard. At the top, there is a banner with a gift icon and text about a free trial. Below the banner is a navigation bar with the text "Google Cloud Platform" and a dropdown menu labeled "Select a project". To the right of the dropdown is a search bar with the placeholder "Search products and resources". A red box highlights the "Select a project" dropdown menu.

- Select “New Project”

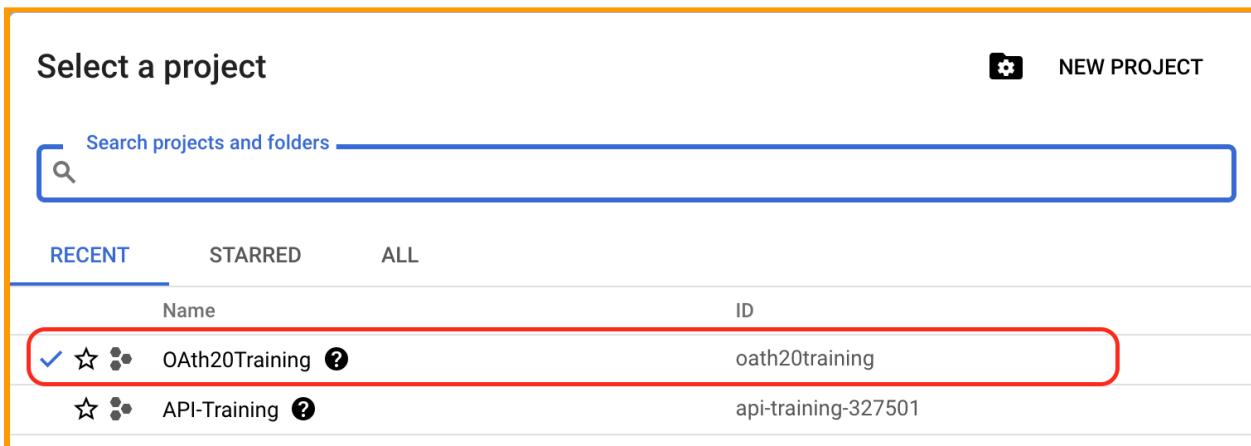
A screenshot of the "Select a project" dialog box. The title is "Select a project". On the right, there is a "NEW PROJECT" button with a gear icon. Below the title is a search bar with the placeholder "Search projects and folders" and a magnifying glass icon. At the bottom, there are three tabs: "RECENT" (which is selected), "STARRED", and "ALL".

- Give Project Name (Example: OAuth20Training)

A screenshot of the "New Project" creation dialog box. The title is "New Project". There is a warning message: "⚠ You have 10 projects remaining in your quota. Request an increase or delete projects. [Learn more](#)". Below the message is a "MANAGE QUOTAS" link. A red box highlights the "Project name *" input field, which contains "OAuth20Training". Below the input field is a note: "Project ID: oauth20training. It cannot be changed later. [EDIT](#)". At the bottom, there is a "Location *" section with "No organization" selected and a "BROWSE" button. A red box also highlights the "No organization" button.



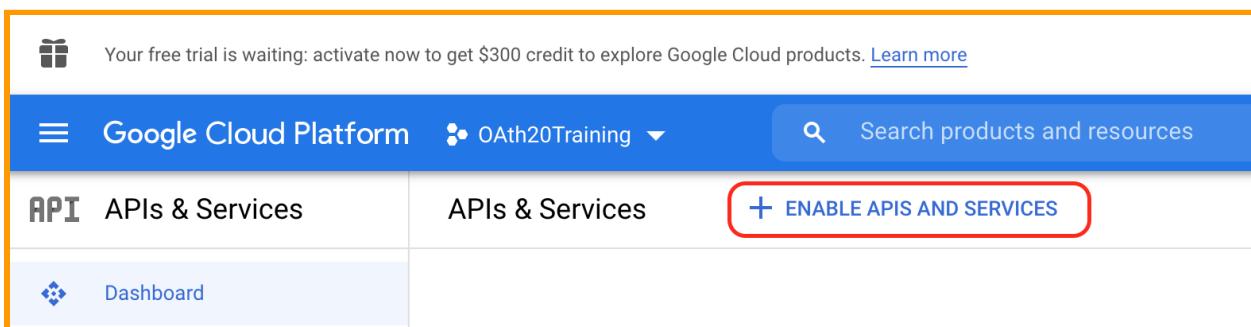
- Select your newly created project



A screenshot of the Google Cloud Platform 'Select a project' interface. At the top, there is a search bar labeled 'Search projects and folders'. Below it, three tabs are visible: 'RECENT' (which is selected), 'STARRED', and 'ALL'. A list of projects follows, each with a name, an icon, and an ID. The first project, 'OAuth20Training', has a checkmark next to it and is highlighted with a red rounded rectangle. Its ID is 'oath20training'. The second project, 'API-Training', is also highlighted with a red rounded rectangle and its ID is 'api-training-327501'.

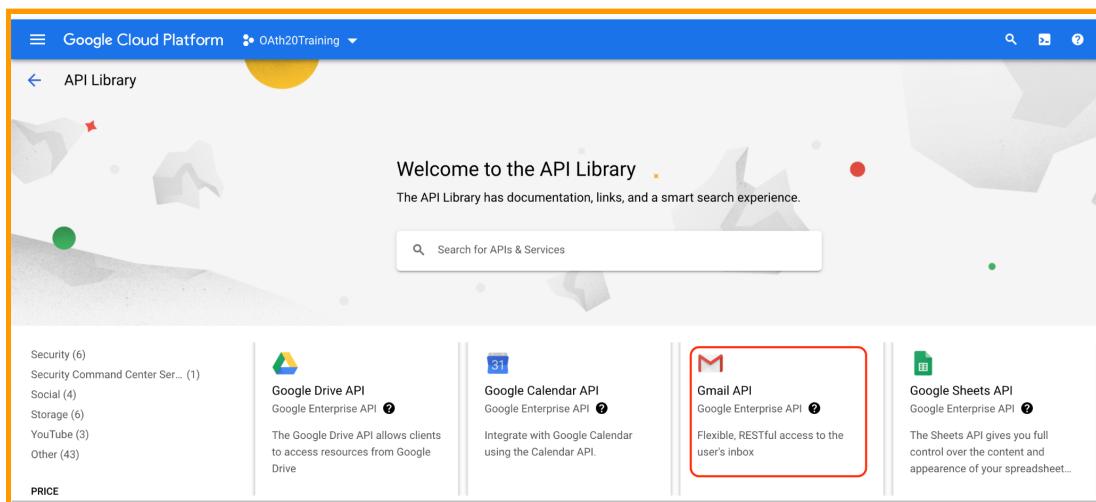
Step 02: Enable API and Services

- Click “ENABLE APIs AND SERVICES” in the dashboard



A screenshot of the Google Cloud Platform dashboard. At the top, there is a message about a free trial and a link to learn more. The navigation bar includes the 'Google Cloud Platform' logo, a dropdown for 'OAuth20Training', and a search bar. Below the navigation bar, there are two main sections: 'API APIs & Services' and 'APIs & Services'. In the 'APIs & Services' section, there is a prominent blue button with a plus sign and the text 'ENABLE APIs AND SERVICES', which is highlighted with a red rounded rectangle.

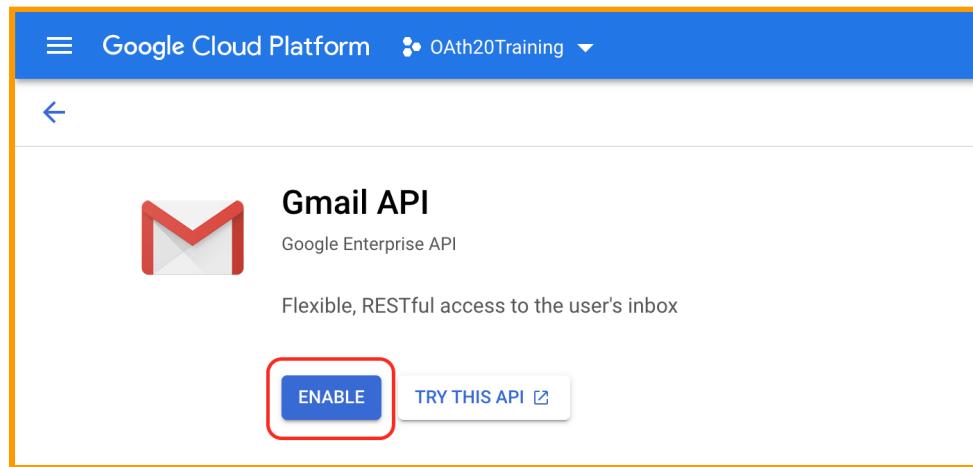
- Select “Gmail API” in the API Library



A screenshot of the Google Cloud Platform API Library. The page title is 'Welcome to the API Library'. On the left, there is a sidebar with categories like 'Security', 'Storage', 'YouTube', and 'Other'. The main area shows several API cards. One card for 'Gmail API' is highlighted with a red rounded rectangle. The card contains the text: 'Flexible, RESTful access to the user's inbox.' Below the card, there is a brief description of the Sheets API.

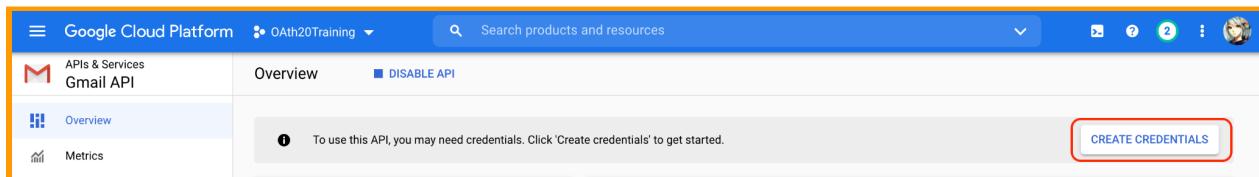


- After you select Gmail API, you have to enable Gmail API. For that, click the “ENABLE” button.

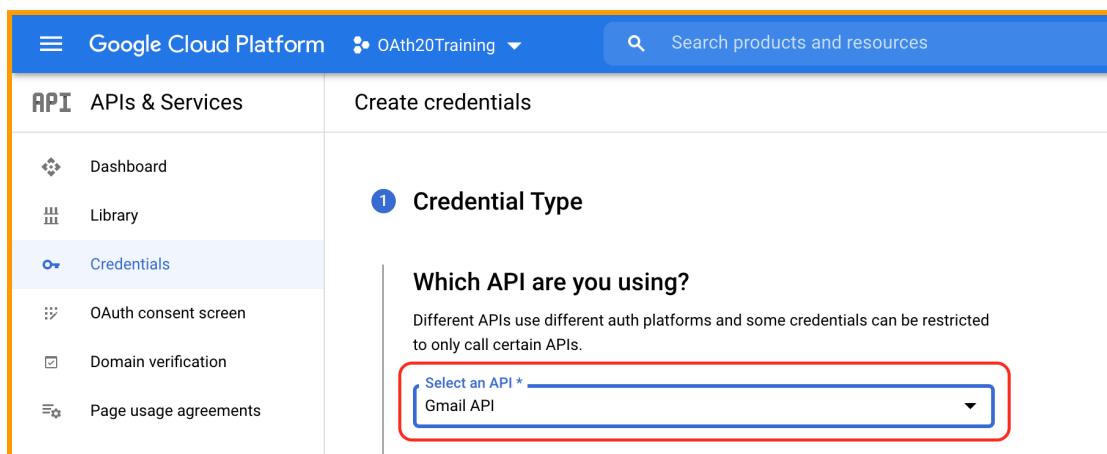


Step 03: Create Credential for Gmail API

- Select “CREATE CREDENTIALS” in Gmail API Overview



- You must choose which API you will use and select “Gmail API” in the “Select an API” drop-down list.





- Now you have to select which data you are going to access. Select “User data”

The screenshot shows the Google Cloud Platform API & Services Credentials page. On the left sidebar, 'Credentials' is selected. In the main area, under 'What data will you be accessing? *', the 'User data' radio button is selected and highlighted with a red box. The 'User data' section contains the text: 'Data belonging to a Google user, like their email address or age. User consent required. This will create an OAuth client.' Below it, the 'Application data' radio button is also present.

Step 04: Configure and Register your App

- Go to the “OAuth Consent screen” and select “User Type” as “External”

The screenshot shows the Google Cloud Platform API & Services OAuth consent screen page. On the left sidebar, 'OAuth consent screen' is selected. In the main area, under 'User Type', the 'External' radio button is selected and highlighted with a red box. The 'External' section contains the text: '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.' Below it, the 'Internal' radio button is also present.



- After you select the "External" user type, now we have to give our application details

- Type your app name (Example: OAuthApp)
 → Type User Support email (you can give your email address)

The screenshot shows the Google Cloud Platform interface for managing APIs & Services. The left sidebar has a 'OAuth consent screen' item selected. The main area is titled 'Edit app registration' and shows the 'OAuth consent screen' tab selected. It includes sections for 'App information' and 'Developer contact information'. The 'App information' section contains fields for 'App name *' (set to 'OAuthApp') and 'User support email *' (set to 'mmoonamaldeniya@gmail.com'). The 'Developer contact information' section contains a field for 'Email addresses *' (set to 'mmoonamaldeniya@gmail.com'). A red box highlights the 'User support email' field.

- Now you have to give the developer's contact information. Put your email address in "Developer contact information." Google will notify you of any changes that happen to your project.

The screenshot shows the Google Cloud Platform interface for managing APIs & Services. The left sidebar has a 'OAuth consent screen' item selected. The main area is titled 'Edit app registration' and shows the 'OAuth consent screen' tab selected. It includes sections for 'Authorized domains' and 'Developer contact information'. The 'Developer contact information' section contains a field for 'Email addresses *' (set to 'mmoonamaldeniya@gmail.com'). A red box highlights the 'Email addresses' field. At the bottom, there are 'SAVE AND CONTINUE' and 'CANCEL' buttons.



Step 05: Create OAuth client ID

- Go to the “Credentials”, click “CREATE CREDENTIAL” and select “OAuth client ID”

The screenshot shows the Google Cloud Platform API & Services Credentials page. The left sidebar is titled 'API APIs & Services' and includes 'Dashboard', 'Library', 'Credentials' (which is selected), 'OAuth consent screen', 'Domain verification', and 'Page usage agreements'. The main area is titled 'Credentials' and contains a 'CREATE CREDENTIALS' button and a 'DELETE' button. A red box highlights the 'OAuth client ID' section, which is described as 'Requests user consent so your app can access the user's data'. Below it are 'API key' (described as 'Identifies your project using a simple API key to check quota and access') and 'Service account' (described as 'Enables server-to-server, app-level authentication using robot accounts'). At the bottom, there is a 'Help me choose' section and a table for 'OAuth 2.0 Client'. The table has columns for 'Name' and 'Creation date'. A red box highlights the 'Name' column of the first row.

- Now you have to select “Application type”. Select “Web application”

The screenshot shows the 'Create OAuth client ID' page. The left sidebar is identical to the previous one. The main area has a back arrow and the title 'Create OAuth client ID'. It explains that a client ID identifies a single app to Google's OAuth servers. Below this is a 'Application type *' dropdown menu. A red box highlights the 'Web application' option, which is the selected choice. Other options listed are 'Android', 'Chrome app', 'iOS', 'TVs and Limited Input devices', 'Desktop app', and 'Universal Windows Platform (UWP)'.



- You have to put 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. you can give any name you like. (Example: Web Client 1)

The screenshot shows the Google Cloud Platform API & Services Credentials page. On the left sidebar, 'Credentials' is selected. In the main area, it says 'Create OAuth client ID'. Under 'Application type *', 'Web application' is chosen. The 'Name *' field contains 'Web client 1', which is highlighted with a red box. A tooltip below the input field states: '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.' At the bottom, a note says: 'The domains of the URIs you add below will be automatically added to your [OAuth consent screen](#) as [authorized domains](#)'.

- After you enter application type and OAuth 2.0 client name, you'll get “Your Client ID” and “Your Client Secret”

The screenshot shows a confirmation dialog titled 'OAuth client created'. It states: 'The client ID and secret can always be accessed from Credentials in APIs & Services'. A note says: 'OAuth access is restricted to the [test users](#) listed on your [OAuth consent screen](#)'. Below this, 'Your Client ID' is listed as '629750194203-d7nc8b0ti8qq08fdhcqpsl6gd02rg0e4.apps.googleusercontent.com' with a copy icon. 'Your Client Secret' is listed with a redacted value and a copy icon. At the bottom, there is a 'DOWNLOAD JSON' button, an 'OK' button, and a small 'x' button.



Step 06: Get Access Token using the Postman

You don't need to download Postman. Try Postman Web version. You can access the Postman web version using the following link.

Postman: <https://www.postman.com/downloads/>

- After you log in to the Postman, first "Create new Collection" in your Workspace.

The screenshot shows the Postman interface. At the top, there are navigation tabs: Home, Workspaces, API Network, Reports, and Explore. Below these, the 'My Workspace' section is visible. On the left, there's a sidebar with 'Collections' (highlighted with a red box), 'APIs', and other options like 'Environments' and 'Mock Servers'. In the main workspace area, there is a collection named 'GMAIL / Get Specific msg from ID'. Below it, a request card for a 'GET' method is shown with fields for 'Enter request URL', 'Params', 'Authorization', 'Headers (5)', 'Body', and 'P'. A large red box highlights the 'Create new Collection' button in the 'Collections' sidebar.

- You can give a name for your new connection (Example: OauthTraining)

The screenshot shows the Postman interface after creating a new collection. The 'Collections' sidebar now lists 'GMAIL' and 'OauthTraining' (highlighted with a red box). The main workspace shows the 'OauthTraining' collection details. It includes tabs for 'Authorization' (which is currently selected), 'Pre-request Script', 'Tests', and 'Variables'. A note says, "This authorization method will be used for every request in this collection. You can override this by specifying one in the request." Below this, there's a 'Type' dropdown set to 'No Auth'. At the bottom, a note states, "This collection does not use any authorization. Learn more about authorization." A red box highlights the collection name 'OauthTraining' in the sidebar.



- Now “Add request” to the new collection you created earlier.

The screenshot shows the Postman application interface. In the left sidebar, under 'My Workspace', there is a 'Collections' section. A context menu is open over the 'OauthTraining' collection, with the 'Add request' option highlighted by a red box. The main workspace shows a collection named 'OauthTraining' with tabs for 'Authorization', 'Pre-request Script', 'Tests', and 'Variables'. A note below says: 'This authorization method will be used for every request in this collection unless one is specified in the request.' The 'Type' dropdown is set to 'No Auth'.

- In here also you can give a name to your newly added request
(Example: Get Massage)

The screenshot shows the Postman application interface. In the left sidebar, under 'My Workspace', there is a 'Collections' section. A new request named 'Get Massage' has been added to the 'OauthTraining' collection. The 'Name' field for this request is highlighted by a red box and contains the text 'Get Massage'. Below it, the 'Method' is set to 'GET' and the 'Enter request URL' field is empty. The 'Params' tab is selected. Other tabs include 'Authorization', 'Headers (5)', 'Body', and 'Pre-request'.

- After you enter the request name, you have to Enter the request URL for that you need to go to the Gmail API > References > REST Resources you can use the following link to access REST resources in Gmail API

Gmail API REST Resources Link :

<https://developers.google.com/gmail/api/reference/rest/v1/users.messages/get>



- After you access REST resources in Gmail API to get the GET HTTP Request :

1. Select `users.messages`
2. Under `users.messages` select `list`
3. Copy HTTP Request in `users.messages.list`

The screenshot shows the Gmail for Developers API Reference. On the left, there's a sidebar with a tree view of REST Resources under the 'users' category. The 'messages' node is highlighted with a red box and labeled '1'. Below it, the 'list' node is also highlighted with a red box and labeled '2'. The main content area shows the 'Method: users.messages.list' with a description: 'Lists the messages in the user's mailbox.' It includes an 'HTTP request' section with a code block: 'GET https://gmail.googleapis.com/gmail/v1/users/{userId}/messages' (labeled '3') and a note: 'The URL uses gRPC Transcoding syntax.' Below that is a 'Path parameters' section with a table for 'Parameters' and a 'userId' entry. The 'userId' column is 'string' and the description says: 'The user's email address. The special value me can be used to indicate th...'. At the bottom is a 'Query parameters' section.

- After you copy HTTP Request from `users.messages.list` past it in "GET" tab in the newly created request.

The screenshot shows the Postman application interface. At the top, there are tabs for Home, Workspaces, API Network, Reports, and Explore, along with a search bar. In the center, there's a 'My Workspace' section with a 'Collections' tab selected. A new request is being created, with the method set to 'GET' and the URL field containing 'https://gmail.googleapis.com/gmail/v1/users/{userId}/messages'. This URL is highlighted with a red box. The request is titled 'Get Massage' and is associated with the 'OauthTraining' collection.



- Now we need to declare `{userId}` in GET HTTP Request for that first we need to create an environment for that click “Add” in “Environment”

The screenshot shows the Postman interface. At the top, there's a search bar and various icons. Below it, a navigation bar shows 'GET Get Massage'. To the right, it says 'No Environment' with a dropdown arrow. On the left, under 'Environment', it says 'No active Environment' and provides a brief description: 'An environment is a set of variables that allow you to switch the context of your requests.' A red box highlights the 'Add' button at the top right of this section.

- You can give a name to the new environment you're going to create (Example: `Oauthtrain`)
- Enter VARIABLE: `userId`
- For “INITIAL VALUE” and “CURRENT VALUE” Put your Gmail address

The screenshot shows the Postman interface with 'My Workspace' selected. In the center, there's a table for environment variables. One row is highlighted with a red box, showing 'VARIABLE' as `userId`, 'INITIAL VALUE' as `mmoonamaldeniya@gmail.com`, and 'CURRENT VALUE' as `mmoonamaldeniya@gmail.com`. A red box also highlights the 'Oauthtrain' environment name in the workspace list.

- After creating an environment, you can select your scope (environment) from the “Type to filter” Dropdown list like the following image.

The screenshot shows a dropdown menu titled 'Type to filter'. It contains three items: 'No Environment', 'GMAIL', and 'Oauthtrain'. The 'Oauthtrain' item is highlighted with a red box.



- After you select your environment, you need to put your Environment Variable name in to GET HTTP Request `{userId}` path parameter.

The screenshot shows a Postman request configuration for a GET request to `https://gmail.googleapis.com/gmail/v1/users/{userId}/messages`. The 'Params' tab is selected, showing a variable named 'userId' with three entries: 'INITIAL' (mmoonamaldeniya@gmail.com), 'CURRENT' (mmoonamaldeniya@gmail.com), and 'SCOPE' (Environment). A red box highlights this section.

- Now we are going to give the authorization to request that we created

→ Click the “Authorization” tab in the request, click the “Type” drop-down list, and select “OAuth 2.0”.

The screenshot shows a Postman request configuration for a GET request to `https://gmail.googleapis.com/gmail/v1/users/{userId}/messages`. The 'Authorization' tab is selected. A dropdown menu for 'Type' is open, showing options like 'Inherit auth...', 'No Auth', 'API Key', 'Bearer Token', 'Basic Auth', 'Digest Auth', 'OAuth 1.0', and 'OAuth 2.0'. The 'OAuth 2.0' option is highlighted with a red box. A red box also highlights the 'Authorization' tab in the request header bar.

**Step 07: Configure New Token**

- Before going to Configure New Token, you have to copy “Callback URL”. The callback URL cannot be modified when authenticating via the browser.

Configure New Token

Configuration Options Advanced Options

Token Name

Grant Type

Callback URL ⓘ

The callback URL cannot be modified when authenticating via the browser.

Auth URL ⓘ

Access Token URL ⓘ

- After you copy “Callback URL” from Configuration options in the Postman request that we create, you must paste the Callback URL in Client ID authorized redirect URIs.

Google API & SERVICES > Credentials > OAuth 2.0 Client IDs > Authorized redirect URIs

In OAuth 2.0 Client IDs, you have to select the Client you created earlier; then, you can paste the Callback URL in Client ID authorized redirect URIs.

Google Cloud Platform OAuth2Training Search products and resources

API APIs & Services Client ID for Web application DOWNLOAD JSON RESET SECRET

Dashboard Library Credentials OAuth consent screen Domain verification Page usage agreements

Authorized JavaScript origins ⓘ

For use with requests from a browser

+ ADD URI

Authorized redirect URIs ⓘ

For use with requests from a web server

URIs *

+ ADD URI



- After you enter a web client redirect URIs now you have to fill Configuration Options to Configure New Token.
 - Token Name: `AccToken` (you can give any name)
 - Grant Type: `Authorization Code`
 - Auth URL: `https://accounts.google.com/o/oauth2/auth`
 - Access Token URL: `https://accounts.google.com/o/oauth2/token`

The screenshot shows the 'Configure New Token' section in Postman. A red box highlights the configuration fields:

- Token Name: `AccToken`
- Grant Type: `Authorization Code`
- Callback URL: `https://oauth.pstmn.io/v1/browser-callback`
- Auth URL: `https://accounts.google.com/o/oauth2/auth`
- Access Token URL: `https://accounts.google.com/o/oauth2/token`

- After you enter the above details, you have to end your client ID and Client Secret. After that, you have to enter the scope. To get the scope, you have to go back to the `Gmail API > References > REST Resources` then to get the authorization scopes

1. Select `users.messages`
2. Under `users.messages` select `list`
3. Go to the "Authorization Scopes"
4. Copy `https://www.googleapis.com/auth/gmail.modify`

The screenshot shows the 'Authorization Scopes' section for the 'list' endpoint in the Gmail API Reference. A red box highlights the scope `https://www.googleapis.com/auth/gmail.modify`.

Estimated total number of results: [redacted]

Requires one of the following OAuth scopes:

- `https://mail.google.com/`
- `https://www.googleapis.com/auth/gmail.modify` (highlighted with a red box)
- `https://www.googleapis.com/auth/gmail.readonly`
- `https://www.googleapis.com/auth/gmail.metadata`

For more information, see the [OAuth 2.0 Overview](#).



- After you copy the authorization scope, you have to paste it to configure the new token's configuration options scope and select Client Authentication as "Send as Basic Auth header" like the following image.

The screenshot shows the Postman interface for generating an OAuth access token. The 'Body' tab is selected. The 'Client ID' field contains a long string of characters, and the 'Client Secret' field is redacted. The 'Scope' field contains 'https://www.googleapis.com/auth/gmail.modify'. The 'Client Authentication' dropdown is set to 'Send as Basic Auth header'. A red box highlights the 'Client ID', 'Client Secret', and 'Scope' fields.

Body	Pre-request Script	Tests	Settings
Client ID ⓘ Client Secret ⓘ Scope ⓘ State ⓘ Client Authentication			
Send as Basic Auth header			
Get New Access Token			

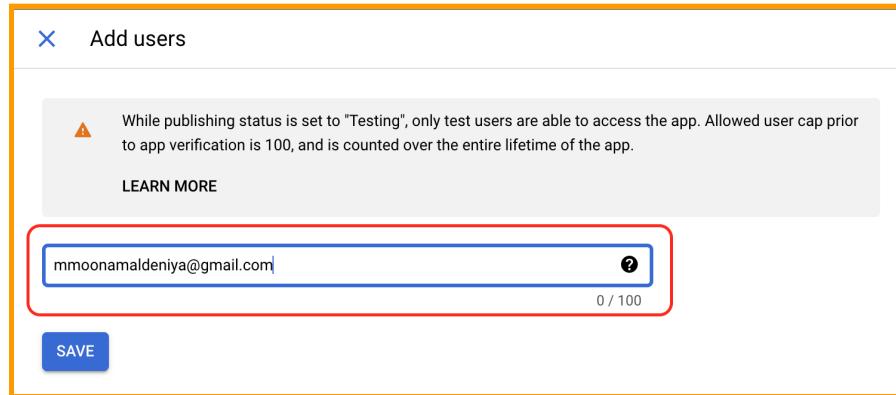
- Before you click Get New Access Token, you need to add Test Users to your app. Only test users can access the app you create.

Google API & SERVICES > OAuth consent screen and go to the **Test Users** and click "**ADD USERS**"

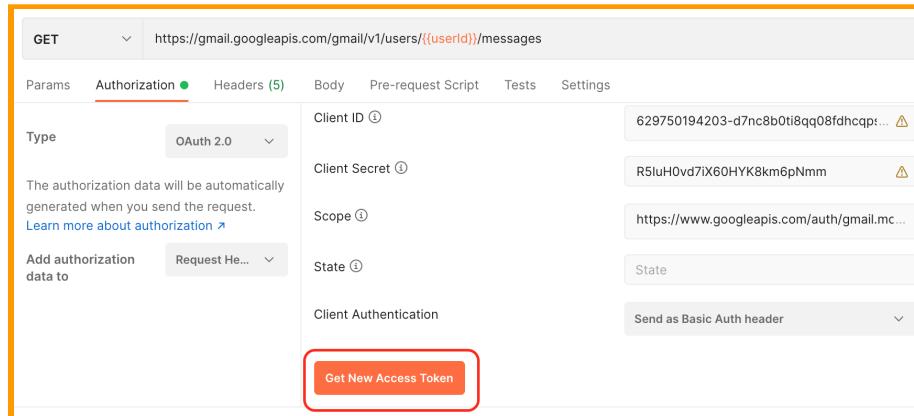
The screenshot shows the Google Cloud Platform OAuth consent screen. The left sidebar has 'APIs & Services' selected. Under 'OAuth consent screen', the 'Test users' section is highlighted with a red box. It shows a progress bar at 0 users (0 test, 0 other) / 100 user cap and a '+ ADD USERS' button.



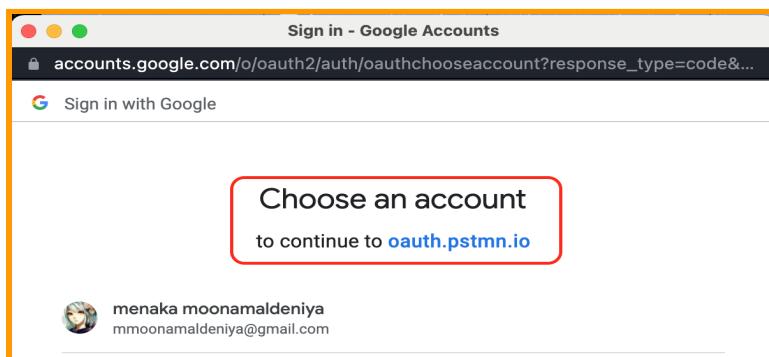
- After you select **ADD USER** you can enter your email address, and not only that, and you can add 100 users if you want. That is the maximum user cap before app verification.



- After you add test users now, you can create a new access token, go back to the Postman and Click **"Get New Access Token"**



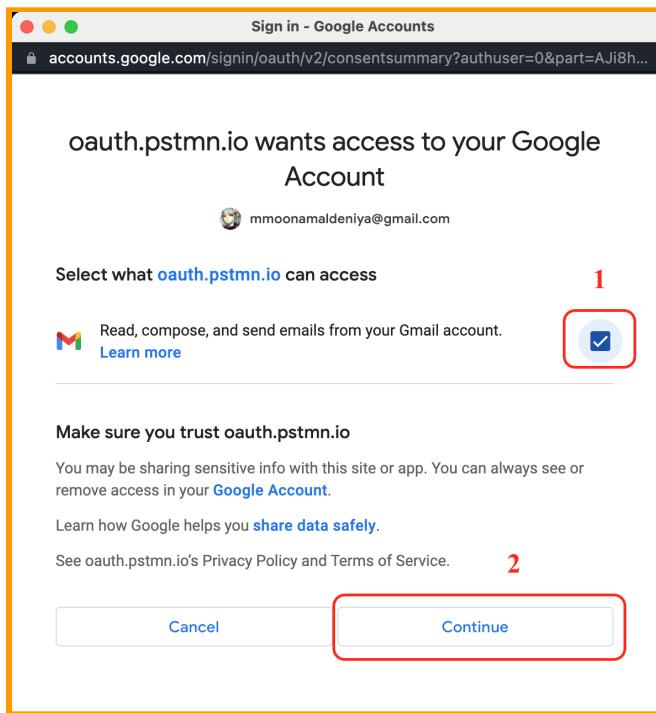
- You need to choose an account to continue; furthermore, select your test user's email address.





- You (Test User) need to give access to your (Test User) google Account.

1. Click the checkbox to give the access
2. Select “Continue” to get the access token



- Now you got your new access token.

A screenshot of a "MANAGE ACCESS TOKENS" interface. It shows a table with columns for Token Name, Access Token, Token Type, expires_in, and scope. One row is selected, showing "AccessToken" as the token name, an access token value, "Bearer" as the token type, "3599" as the expires_in value, and "https://www.googleapis.com/auth/gmail.modify" as the scope. The "Access Token" column is highlighted with a red rectangle. The entire row for the selected token is also highlighted with a red rounded rectangle.



- You can use a new access token in your request. Please select your request and choose “Authorization” and go to the “Access Token” drop-down list and select your new access token like the following image.

The screenshot shows the Postman interface for a GET request to `https://gmail.googleapis.com/gmail/v1/users/{{userId}}/messages`. The 'Authorization' tab is selected. In the 'Access Token' dropdown, a new token has been selected, highlighted with a red box. The token value is:

```
ya29.aOARdaM_GUT0qd7RIP_LcL01S6
LB4dzhMOFvqu4QRoeffgRUMF-
bc6JW2XpvpgE8t8FVGectuah5EMcyBJ
Ld0ej7ab2BG9R3MaeHMtC8CDTTroBJ
8bUsNf2dRpRJ3N8fvk-
vPThYbgSWLnRKM7TimGzGW5-
DBwbuD
```

- You have an access token, and now you can click “Send” to send a request.

The screenshot shows the Postman interface for the same request. The 'Send' button is highlighted with a red box. The status bar at the bottom right indicates: Status: 200 OK Time: 607 ms Size: 8.61 KB.

- After you send the request, you'll get a list of all messages with “id”, “threadId”, “Status”, “Time” and “Size”.

The screenshot shows the JSON response body from the request. It contains three message objects, each with an 'id' and a 'threadId'. The third message object is highlighted with a red box.

```

1
2   "messages": [
3     {
4       "id": "17c31c1a6d85f7ec",
5       "threadId": "17c31c1a6d85f7ec"
6     },
7     {
8       "id": "17c31b986d9eb094",
9       "threadId": "17c31b986d9eb094"
10    },
11    {
12      "id": "17c31a04d6eb3c31",
13      "threadId": "17c31a04d6eb3c31"
14    }
15  ]

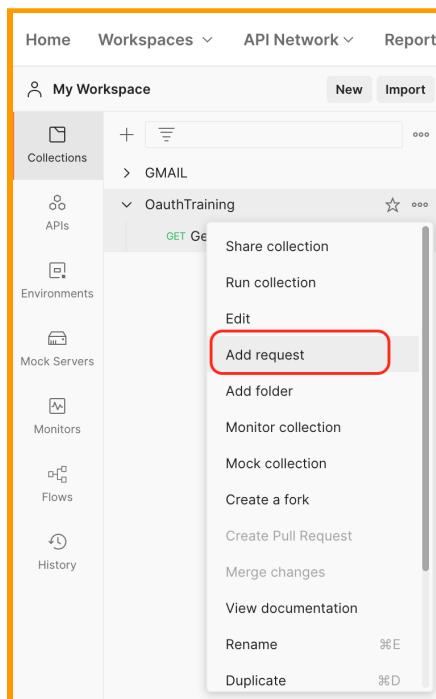
```



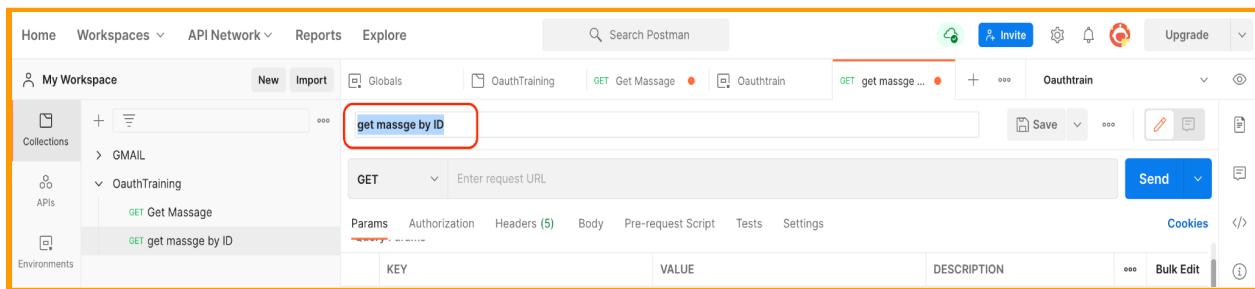
Step 08: Get All details from a specific message

Previously, we used “users.messages.list” to List the messages in the user's mailbox. Now we are going to use “users.messages.get” to gets the specified message.

- Again you have to create a new request. Go to the Collection you previously created and select “Add Request”



- Give a name to the request (Example: get message by ID)





- Now we need to add a GET HTTP request for our new request in. Here we need to go to the **Gmail API > References > REST Resources** to get the `users.messages.get` HTTP Request.

- Select `users.messages`
- Under `users.messages` select `get`
- Copy GET HTTP Request in `users.messages.get`

The screenshot shows the Gmail for Developers API Reference page. The left sidebar has a red box around the 'users.messages' item under 'REST Resources'. A red number '1' is placed next to it. Below it, the 'get' item is highlighted with a blue box, and a red number '2' is placed next to it. The main content area shows the 'Method: users.messages.get' with a description: 'Gets the specified message.' To the right of the method name is a red number '3'. Below the description is the HTTP request URL: `GET https://gmail.googleapis.com/gmail/v1/users/{userId}/messages/{id}`. A red box surrounds this URL, and a red number '3' is placed to its right. Below the URL, a note says: 'The URL uses gRPC Transcoding syntax.'

- Now you can past `users.messages.get` HTTP Request in GET tab in new request

The screenshot shows the Postman interface with a red box around the 'Params' tab. The 'Method' dropdown is set to 'GET' with a red box around it, and a red number '1' is placed next to it. The 'URL' field contains the value `https://gmail.googleapis.com/gmail/v1/users/{userId}/messages/{id}`, which is also surrounded by a red box, and a red number '2' is placed next to it. Below the URL are tabs for 'Params', 'Authorization', 'Headers (5)', 'Body', 'Pre-request Script', 'Tests', and 'Settings'.



- Now you need to declare “userId” and “id” path parameters,

- you can use the previous environment to declare “userId” and “id” you can choose any “id” in the messages list that we got in Step 07.

```

Body Cookies Headers (13) Test Results
Pretty Raw Preview Visualize JSON ↻
1 {
2   "messages": [
3     {
4       "id": "17c31c1a6d85f7ec",
5       "threadId": "17c31c1a6d85f7ec"
6     },
7     {
8       "id": "17c31b986d9eb094",
9       "threadId": "17c31b986d9eb094"
10    },
11    {
12      "id": "17c31a04d6eb3c31",
13      "threadId": "17c31a04d6eb3c31"
}

```

- After you get the Environment variable(userId) and id from the messages list, you can put those into the GET HTTP Request “userId” and “id” path parameters, respectively.

Explore Search Postman

Globals OauthTraining GET Get Massage OAuthtrain GET get massge ... + 1 Oauthtrain 1

OauthTraining / get massge by ID

GET https://gmail.googleapis.com/gmail/v1/users/{{userId}}/messages/17c31c1a6d85f7ec 2

Params Authorization Headers (5) Body Pre-request Script Tests Settings Cookies

- Now we are going to give the authorization to request that we created

→ Click the “Authorization” tab in the request, click the “Type” drop-down list, and select “OAuth 2.0”.

Params	Authorization	Headers (5)	Body
	Type Inherit auth... The authorization header will automatically generate the required headers for the request. Learn more about this. Response		
	Inherit auth from pa... No Auth API Key Bearer Token Basic Auth Digest Auth OAuth 1.0 OAuth 2.0 Hawk Authentication		



- After selecting the authorization type, you can choose the access token. For access token, you can use the previous token you configured. Go to the “Access Token” drop-down list and select the access token that you previously created.

OauthTraining / get massege by ID

GET https://gmail.googleapis.com/gmail/v1/users/{{userId}}/messages/17c31c1a6d85f7ec

Params Authorization **Headers (5)** Body Pre-request Script Tests Settings

Type OAuth 2.0

The authorization data will be automatically generated when you send the request.
Learn more about authorization ↗

Add authorization data to Request He... ▾

Current Token

Access Token Available Tokens

Header Prefix ⓘ

Manage Tokens

AccToken

- Now you can send the request to get the specific message details for that click “Send”

OauthTraining / get massege by ID

GET https://gmail.googleapis.com/gmail/v1/users/{{userId}}/messages/17c31c1a6d85f7ec

Params Authorization **Headers (6)** Body Pre-request Script Tests Settings

Type OAuth 2.0

The authorization data will be automatically generated when you send the request.
Learn more about authorization ↗

Add authorization data to Request He... ▾

Current Token

This access token is only available to you. Sync the token to let collaborators on this request use it.

Access Token Available Tokens

ya29.a0ARrdaM_GUT0qd7RIP_cL01S6L

Header Prefix ⓘ

Bearer

Send



- After sending the request, you can get the particular message id's all details (labels, payload, snippet, header, etc.).

The screenshot shows a POSTMAN interface with the following details:

- Method:** GET
- URL:** https://gmail.googleapis.com/gmail/v1/users/{{userId}}/messages/17c31c1a6d85f7ec
- Authorization:** OAuth 2.0 (selected)
- Headers:** (6)
- Body:** (empty)
- Tests:** (empty)
- Settings:** (empty)
- Cookies:** (empty)
- Type:** OAuth 2.0
- Buttons:** Save, Send, Get New Access Token
- Status:** 200 OK, Time: 989 ms, Size: 109.69 KB
- Response View:** Pretty (selected), Raw, Preview, Visualize, JSON
- Content of Response (Pretty Print):**

```
1 {
2   "id": "17c31c1a6d85f7ec",
3   "threadId": "17c31c1a6d85f7ec",
4   "labelIds": [
5     "UNREAD",
6     "CATEGORY_SOCIAL",
7     "INBOX"
8   ],
9   "snippet": "Your job recommendations from LinkedIn",
10  "payload": {
11    "partId": "",
12    "mimeType": "multipart/alternative",
13  }
14}
```