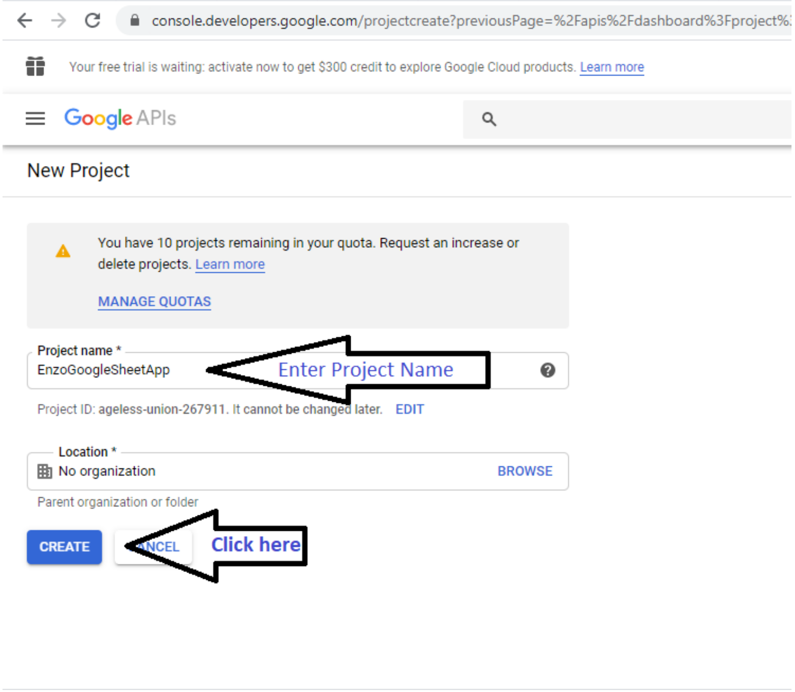
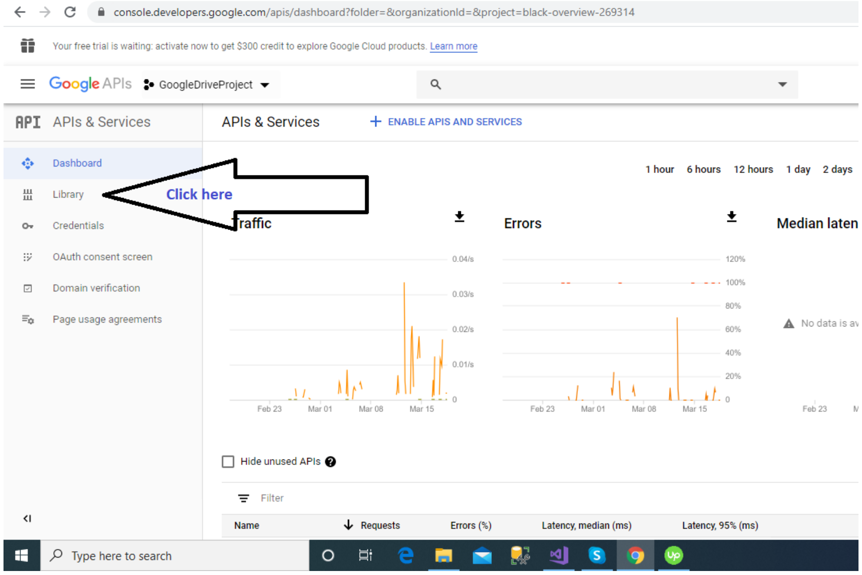
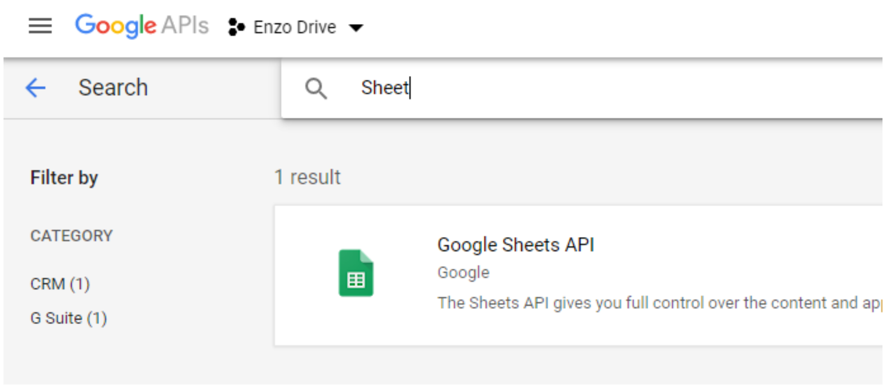
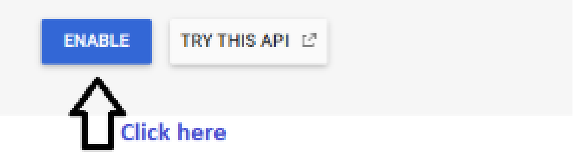
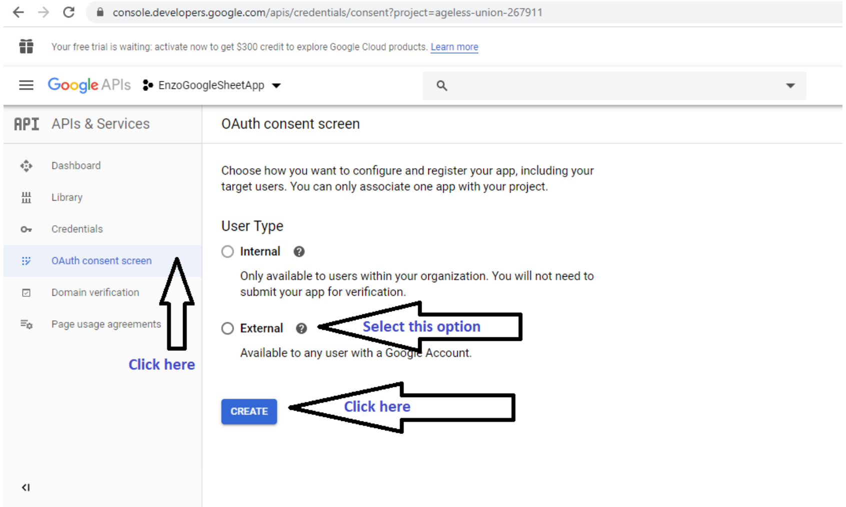
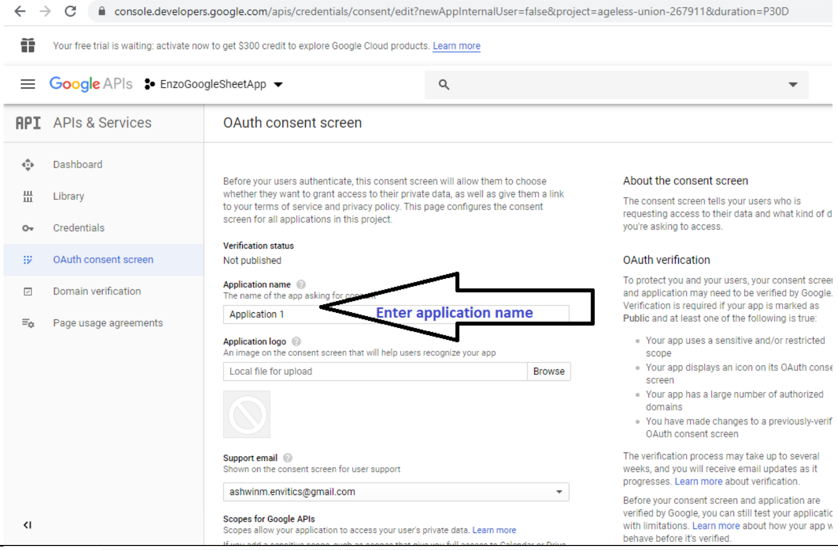
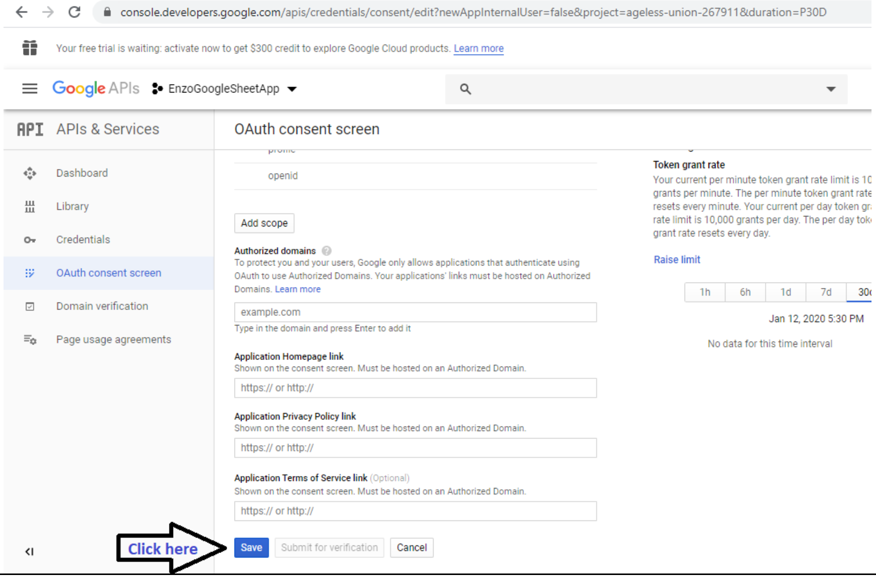
GoogleSheet: Configuration Steps

To complete the steps below, you must first install the GoogleSheet adapter from the Marketplace in Enzo Manager. Make sure you see the GoogleSheet adapter on the left menu of Enzo Manager before proceeding.

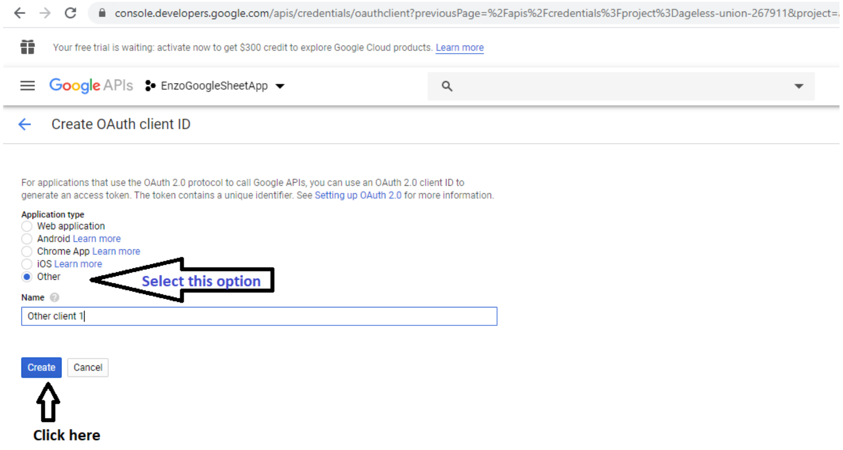
> *NOTE: Obtaining the necessary secrets you must already have a Google account with access to a GoogleDrive. Google requires OAuth authentication with a client flow. Once the steps are completed you will be able to use the GoogleSheet adapter.*

Logon to the Google Developer Console with your Google account: https://console.developers.google.com/	
Create a new Project (ex: EnzoGoogleSheetApp)	 console.developers.google.com/projectcreate?previousPage=%2Fapis%2Fdashboard%3Fproject% Your free trial is waiting: activate now to get $300 credit to explore Google Cloud products. Learn more New Project You have 10 projects remaining in your quota. Request an increase or delete projects. Learn more Project name * EnzoGoogleSheetApp Enter Project Name Project ID: ageless-union-267911. It cannot be changed later. Location * No organization Parent organization or folder Click here Google APIs MANAGE QUOTAS EDIT BROWSE CREATE NCEL

From the Library section, add the Google Drive API	 console.developers.google.com/apis/dashboard?folder=&organizationId=&project=black-overview-269314 Your free trial is waiting: activate now to get $300 credit to explore Google Cloud products. Learn more Google APIs GoogleDriveProject APIs & Services APIs & Services ENABLE APIS AND SERVICES Dashboard Library Click here Credentials OAuth consent screen Domain verification Page usage agreements 1 hour 6 hours 12 hours 1 day 2 days Traffic Errors Median laten No data is av Hide unused APIs Filter Name Requests Errors (%) Latency, median (ms) Latency, 95% (ms) Type here to search Google APIs Enzo Drive Search Sheet Filter by 1 result CATEGORY CRM (1) G Suite (1) Google Sheets API Google The Sheets API gives you full control over the content and ap
Enable the Google Sheet API	

Create an Internal or External OAuth consent screen	 console.developers.google.com/apis/credentials/consent?project=ageless-union-267911 Your free trial is waiting: activate now to get $300 credit to explore Google Cloud products. Learn more Google APIs EnzoGoogleSheetApp APIs & Services OAuth consent screen Choose how you want to configure and register your app, including your target users. You can only associate one app with your project. User Type ○ Internal Only available to users within your organization. You will not need to submit your app for verification. ○ External Available to any user with a Google Account. Dashboard Library Credentials OAuth consent screen Domain verification Page usage agreements Click here Select this option CREATE Click here
Enter an application name	 console.developers.google.com/apis/credentials/consent/edit?newAppInternalUser=false&project=ageless-union-267911&duration=P30D Your free trial is waiting: activate now to get $300 credit to explore Google Cloud products. Learn more Google APIs EnzoGoogleSheetApp APIs & Services OAuth consent screen Before your users authenticate, this consent screen will allow them to choose whether they want to grant access to their private data, as well as give them a link to your terms of service and privacy policy. This page configures the consent screen for all applications in this project. Verification status Not published Application name The name of the app asking for consent Application 1 Application logo An image on the consent screen that will help users recognize your app Local file for upload Support email Shown on the consent screen for user support ashwinm.envitics@gmail.com Scopes for Google APIs Scopes allow your application to access your user's private data. Learn more About the consent screen The consent screen tells your users who is requesting access to their data and what kind of d you're asking to access. OAuth verification To protect you and your users, your consent scree and application may need to be verified by Google. Verification is required if your app is marked as Public and at least one of the following is true: - Your app uses a sensitive and/or restricted scope - Your app displays an icon on its OAuth conse screen - Your app has a large number of authorized domains - You have made changes to a previously-verif OAuth consent screen The verification process may take up to several weeks, and you will receive email updates as it progresses. Learn more about verification. Before your consent screen and application are verified by Google, you can still test your applicatic with limitations. Learn more about how your app w behave before it's verified. Dashboard Library Credentials OAuth consent screen Domain verification Page usage agreements Enter application name Browse
Change the Token grant rate if needed and click Save	 console.developers.google.com/apis/credentials/consent/edit?newAppInternalUser=false&project=ageless-union-267911&duration=P30D Your free trial is waiting: activate now to get $300 credit to explore Google Cloud products. Learn more Google APIs EnzoGoogleSheetApp APIs & Services OAuth consent screen openid Authorized domains To protect you and your users, Google only allows applications that authenticate using OAuth to use Authorized Domains. Your applications' links must be hosted on Authorized Domains. Learn more example.com Type in the domain and press Enter to add it Application Homepage link Shown on the consent screen. Must be hosted on an Authorized Domain. https:// or http:// Application Privacy Policy link Shown on the consent screen. Must be hosted on an Authorized Domain. https:// or http:// Application Terms of Service link (Optional) Shown on the consent screen. Must be hosted on an Authorized Domain. https:// or http:// Token grant rate Your current per minute token grant rate limit is 1 grants per minute. The per minute token grant rate resets every minute. Your current per day token gr rate limit is 10,000 grants per day. The per day tok grant rate resets every day. Jan 12, 2020 5:30 PM No data for this time interval Dashboard Library Credentials OAuth consent screen Domain verification Page usage agreements Add scope Raise limit 1h 6h 1d 7d 30 Click here Save Submit for verification Cancel
Select Credentials	

Create an OAuth Client ID (choose Other or TV and Limited Input as the application type)



console.developers.google.com/apis/credentials/oauthclient?previousPage=%2Fapis%2Fcredentials%3Fproject%3Dageless-union-267911&project=

Your free trial is waiting: activate now to get $300 credit to explore Google Cloud products. Learn more

Create OAuth client ID

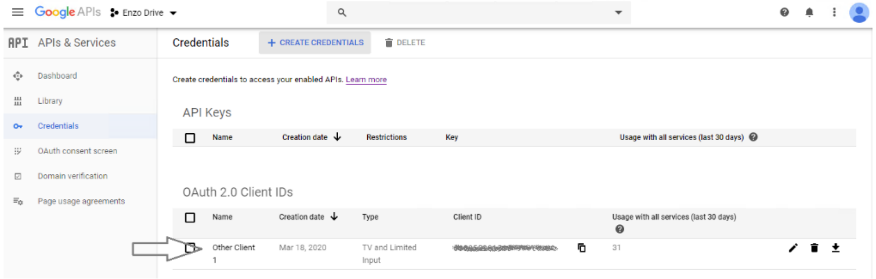
For applications that use the OAuth 2.0 protocol to call Google APIs, you can use an OAuth 2.0 client ID to generate an access token. The token contains a unique identifier. See Setting up OAuth 2.0 for more information.

Application type

○ Web application
○ Android Learn more
○ Chrome App Learn more
○ iOS Learn more
● Other

Name: Other client 1

Once the OAuth credentials have been created, get all the necessary secrets (will be needed by Enzo) - click on the OAuth credential and copy the **ClientID**, **Client Secret**.



Create credentials to access your enabled APIs. Learn more

API Keys

Name	Creation date ↓	Restrictions	Key	Usage with all services (last 30 days)

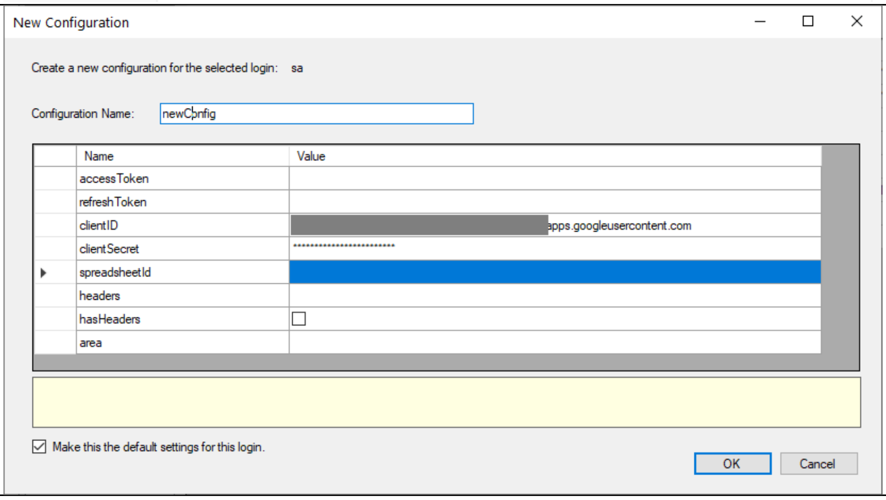
OAuth 2.0 Client IDs

Name	Creation date ↓	Type	Client ID	Usage with all services (last 30 days)
Other Client 1	Mar 18, 2020	TV and Limited Input	[redacted]	31

Go back to Enzo Manager and choose the GoogleSheet adapter; then Add a new Configuration.

Enter the ClientID and ClientSecret values. Click OK.

The accessToken and refreshToken values will be entered later.



New Configuration

Create a new configuration for the selected login: sa

Configuration Name: newConfig

Name	Value
accessToken	
refreshToken	
clientID	[redacted] apps.googleusercontent.com
clientSecret	************************
spreadsheetId	
headers	
hasHeaders	□
area	

☑ Make this the default settings for this login.

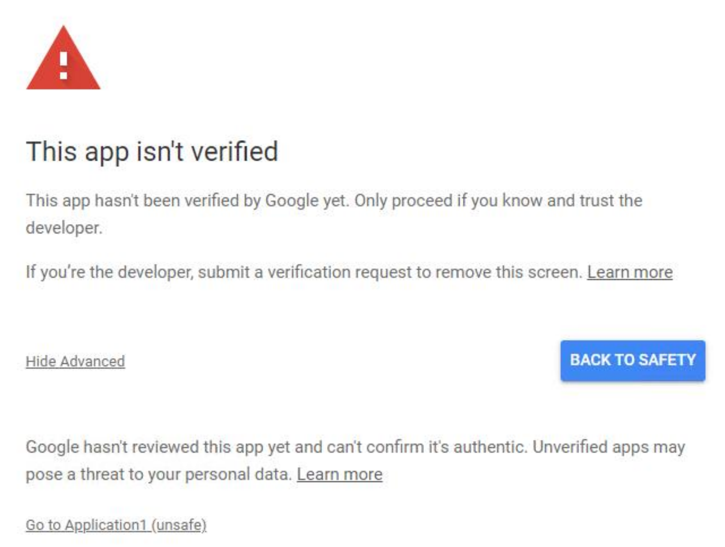
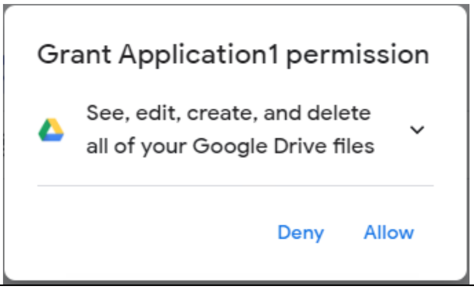
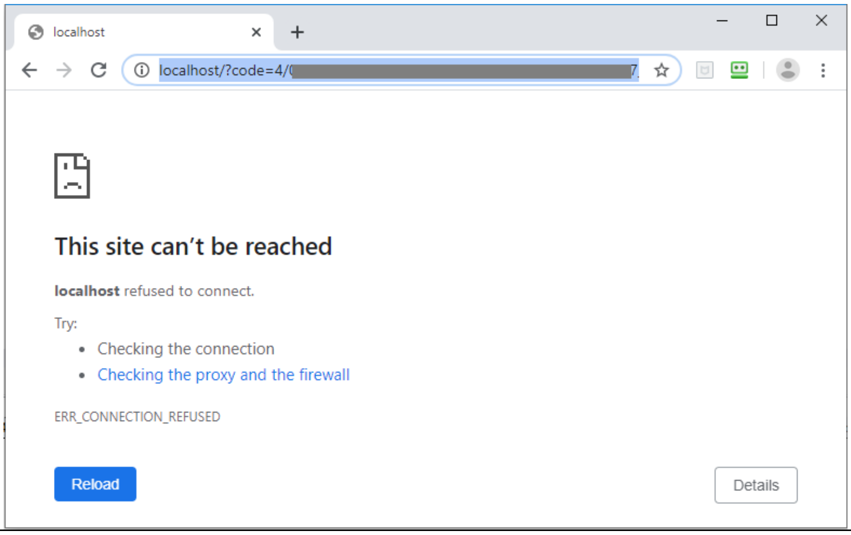
Connect to Enzo directly from SQL Server Management Studio and execute the following command to obtain the URL to use for Google.

NOTE: Execute the SQL Command provided on the right. For this command to work, you must connect to ENZO Server **directly** from SSMS (localhost,9550 when running locally using default configuration settings).

```
EXEC GoogleSheet.getAuthURL
```

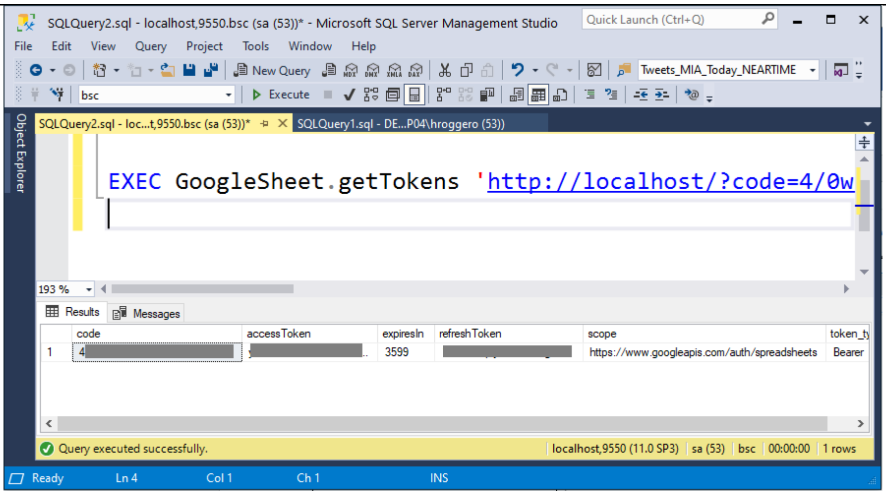
The above command will return something like this: https://accounts.google.com/o/oauth2/v2/auth?......

Copy and paste the entire URL in a browser

In the browser where you pasted the URL obtained previously, follow the prompts to grant access to the Project you created earlier. You may be prompted with a screen that indicates the application is not verified by Google. This is normal since we are using External OAuth, but we haven't published the project for public use. Click on Advanced, and choose the link **Go to (application) (unsafe)** at the bottom.	 This app isn't verified This app hasn't been verified by Google yet. Only proceed if you know and trust the developer. If you're the developer, submit a verification request to remove this screen. Learn more Google hasn't reviewed this app yet and can't confirm it's authentic. Unverified apps may pose a threat to your personal data. Learn more Go to Application1 (unsafe)
Additional screens like the one shown here will prompt you to Allow the application to access your Google Drive. Click **Allow** and continue the flow until a page shows up (next step).	 Grant Application1 permission See, edit, create, and delete all of your Google Drive files
You will then be redirected to a web page with a URL that cannot be reached; this is normal. Copy the ENTIRE URL from the browser URL bar.	 localhost/?code=4/[redacted] This site can't be reached **localhost** refused to connect. Try: • Checking the connection • Checking the proxy and the firewall ERR_CONNECTION_REFUSED

Back in SQL Server Management Studio, execute the following SQL Statement to obtain the access tokens, by pasting the URL as the parameter.	`EXEC GoogleSheet.getTokens`
	`'http://localhost/?code=4/... '`

You will need the tokens returned by this SQL in the next step.



SQLQuery2.sql - localhost,9550.bsc (sa (53))* - Microsoft SQL Server Management Studio
Quick Launch (Ctrl+Q)
File Edit View Query Project Tools Window Help
New Query
Tweets_MIA_Today_NEARTIME
bsc
Execute
SQLQuery2.sql - loc...t,9550.bsc (sa (53))*
SQLQuery1.sql - DE...P04\hroggero (53))
Object Explorer
193 %
Results
Messages

```
EXEC GoogleSheet.getTokens 'http://localhost/?code=4/0w
```

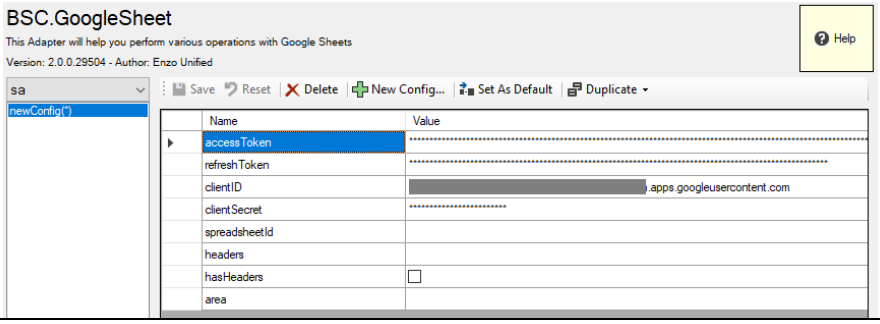
	code	accessToken	expiresIn	refreshToken	scope	token_ty
1	4[redacted]	[redacted]	3599	[redacted]	https://www.googleapis.com/auth/spreadsheets	Bearer

Query executed successfully.

localhost,9550 (11.0 SP3) | sa (53) | bsc | 00:00:00 | 1 rows
Ready Ln 4 Col 1 Ch 1 INS

Back in Enzo Manager, copy and paste the accessToken and the refreshToken values in the configuration setting, and click Save.

The GoogleSheet is now ready for configuration.



BSC.GoogleSheet

This Adapter will help you perform various operations with Google Sheets

Version: 2.0.0.29504 - Author: Enzo Unified

Name	Value
accessToken	••
refreshToken	••
clientID	[redacted].apps.googleusercontent.com
clientSecret	••••••••••••••••••••••••
spreadsheetId	
headers	
hasHeaders	□
area	

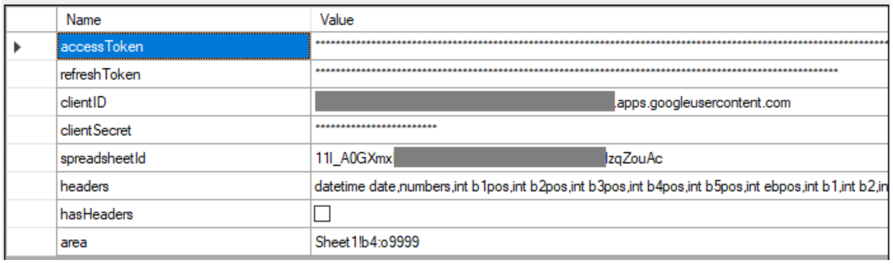
Each configuration setting points to a specific GoogleSheet as specified in the googleSheetId configuration setting.

The optional area value is used to specify a default area of a Sheet, including the SheetId where the data is located. If not specified the A1:Z400000 will be used.

When hasHeaders is specified, the first row of the area provided is assumed to have column names; the column names will be used when returning data.

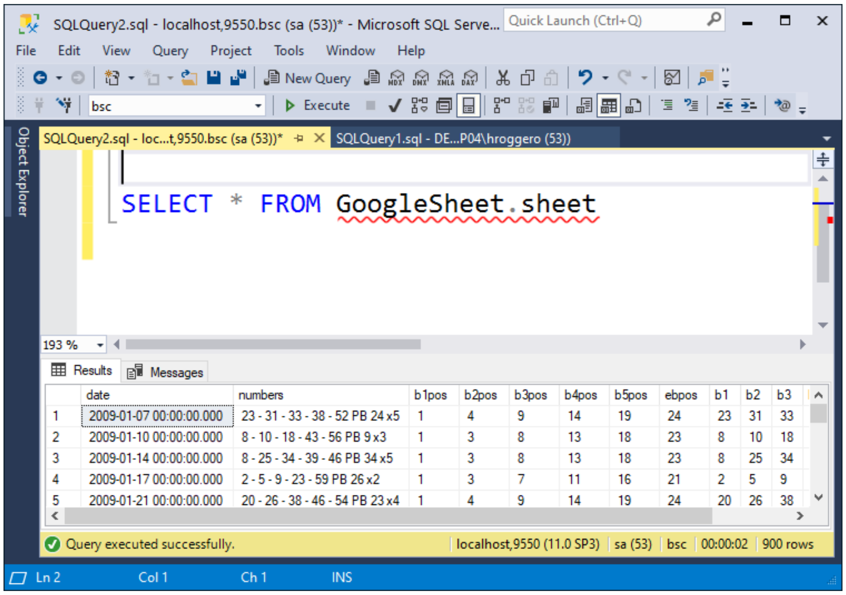
The optional headers value indicates the list of columns to be returned by Enzo and include a C# data type (ex: int id,string name,double age). If not specified all columns will be treated as a string.

Example configuration for a SpreadSheet:



Name	Value
accessToken	••
refreshToken	••
clientID	[redacted].apps.googleusercontent.com
clientSecret	••••••••••••••••••••••••
spreadsheetId	11I_A0GXmx[redacted]IzqZouAc
headers	datetime date,numbers,int b1pos,int b2pos,int b3pos,int b4pos,int b5pos,int ebpos,int b1,int b2,in
hasHeaders	□
area	Sheet1!b4:o9999

You can now execute SQL commands against the GoogleSheet from SQL Server Management Studio


SQLQuery2.sql - localhost,9550.bsc (sa (53))* - Microsoft SQL Serve...
Quick Launch (Ctrl+Q)
File
Edit
View
Query
Project
Tools
Window
Help
New Query
bsc
Execute
Object Explorer
SQLQuery2.sql - loc...t,9550.bsc (sa (53))*
SQLQuery1.sql - DE...P04\hroggero (53))
193 %
Results
Messages

```sql
SELECT * FROM GoogleSheet.sheet
```

	date	numbers	b1pos	b2pos	b3pos	b4pos	b5pos	ebpos	b1	b2	b3
1	2009-01-07 00:00:00.000	23 - 31 - 33 - 38 - 52 PB 24 x5	1	4	9	14	19	24	23	31	33
2	2009-01-10 00:00:00.000	8 - 10 - 18 - 43 - 56 PB 9 x3	1	3	8	13	18	23	8	10	18
3	2009-01-14 00:00:00.000	8 - 25 - 34 - 39 - 46 PB 34 x5	1	3	8	13	18	23	8	25	34
4	2009-01-17 00:00:00.000	2 - 5 - 9 - 23 - 59 PB 26 x2	1	3	7	11	16	21	2	5	9
5	2009-01-21 00:00:00.000	20 - 26 - 38 - 46 - 54 PB 23 x4	1	4	9	14	19	24	20	26	38

Query executed successfully. | localhost,9550 (11.0 SP3) | sa (53) | bsc | 00:00:02 | 900 rows

Ln 2 Col 1 Ch 1 INS