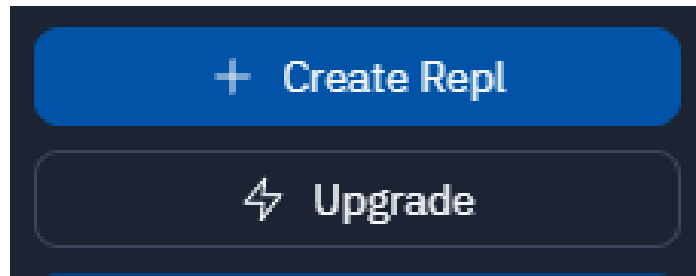


HOW TO

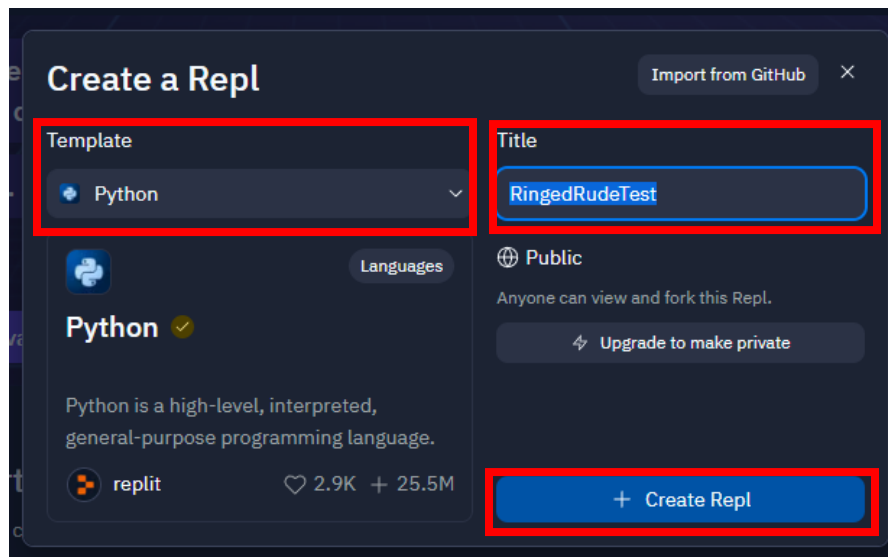
[CheapBotsDoneQuick-like python bot for Twitter](#)

Replit Hosting Version





- 1.) Go to replit.com and create an account if you don't have one.
- 2.) Click on "+ Create Repl" on the top left of your Repl account.



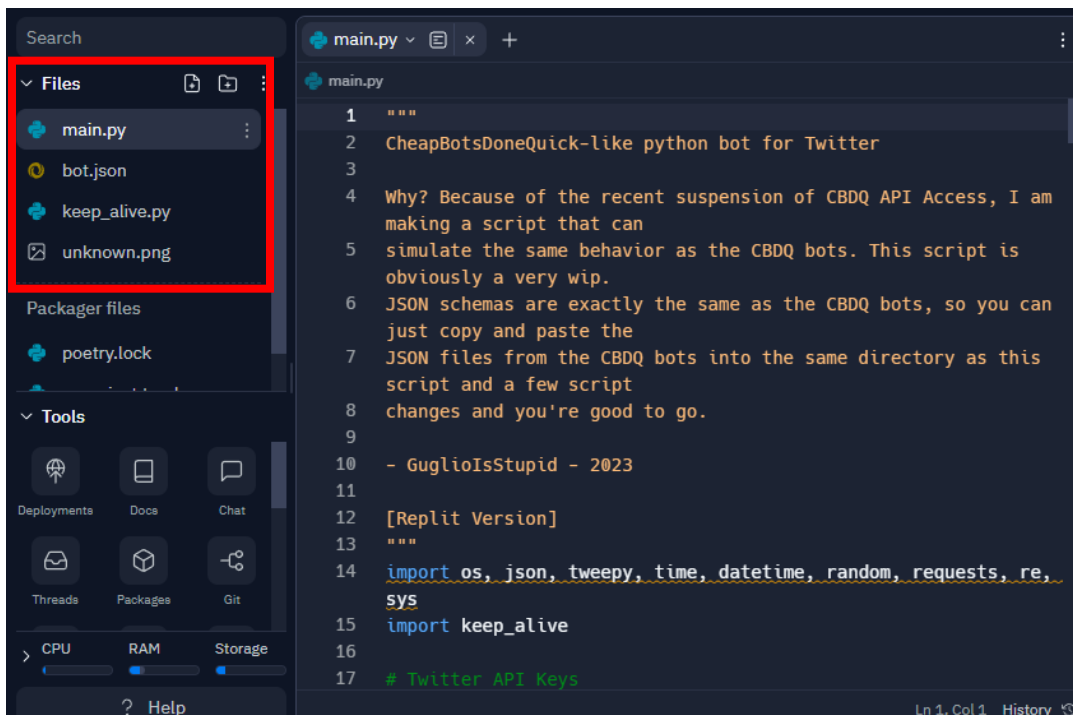
- 3.) Search for "Python" as the template and give it a title. Then click "Create Repl".



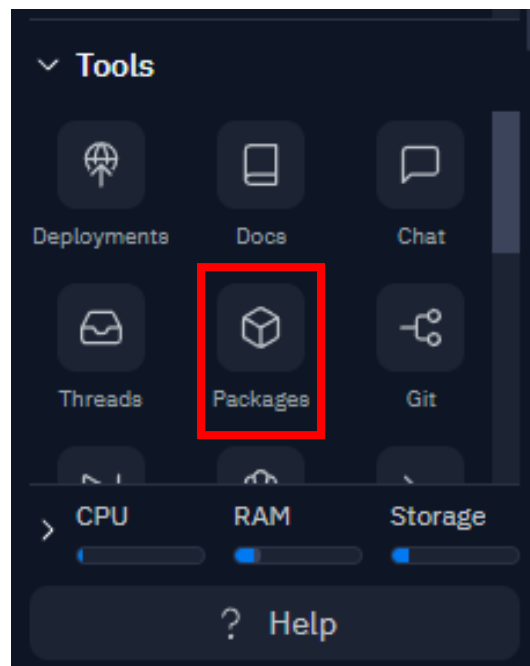
- 4.) Download the latest release of the Github repository [here](#) (specifically the **replit.zip** file). Unzip the file on your computer. The zip file should contain these files:

Downloads > replit (1).zip		
Name	Type	Compressed size
 bot.json	JSON File	2 KB
 keep_alive.py	Python File	1 KB
 main.py	Python File	3 KB
 unknown.png	PNG File	12 KB

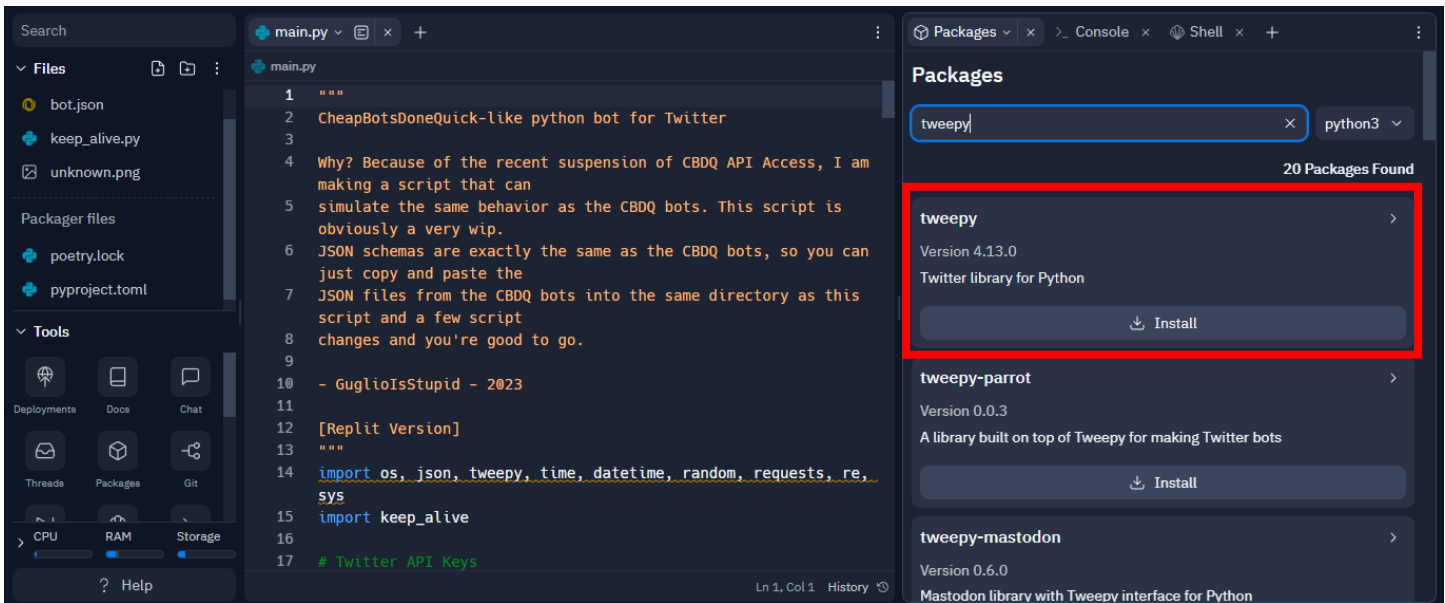
5.) Drag all files from step 4 to your Repl created on step 3. Drag it in the **Files** section. If it asks you to overwrite the main.py file, you **overwrite** it. The result should be this:



6.) Under the **Tools** section on the lower left side of your Repl, click **Packages**.



7.) On the Packages window, search for **tweepy**. Click Install. Wait for the installation to finish.



8.) This part is where we get our Twitter Developer Account. Register your bot account in <https://developer.twitter.com/>. Sign up for **a Free account only**. Fill out the use case form and click submit.

Developer agreement & policy

Describe all of your use cases of Twitter's data and API:
We need this information for data protection. [Learn more](#)

⚠ Must be 250 characters or longer

☐ You understand that you may not resell anything you receive via the Twitter APIs

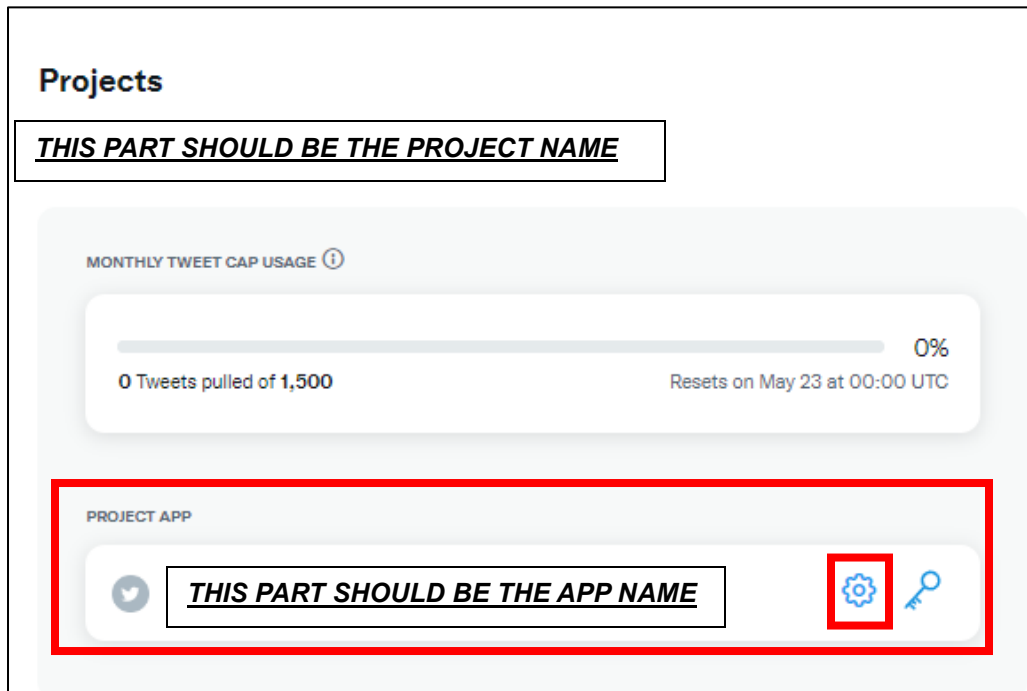
☐ You understand your Developer account may be terminated if you violate the Developer Agreement or any of the Incorporated Developer Terms

☐ You accept the Terms & Conditions

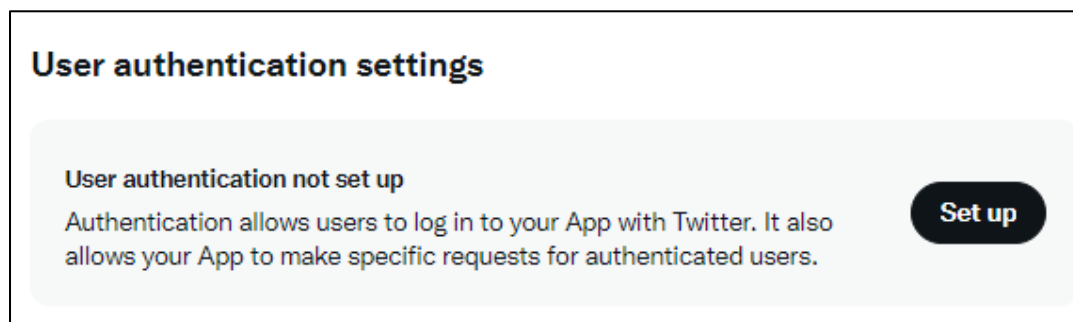
By clicking on the box, and by otherwise accessing or using any Licensed Material, you indicate that you have read and agree to this [Developer Agreement](#) and the [Twitter Developer Policy](#)

BackSubmit

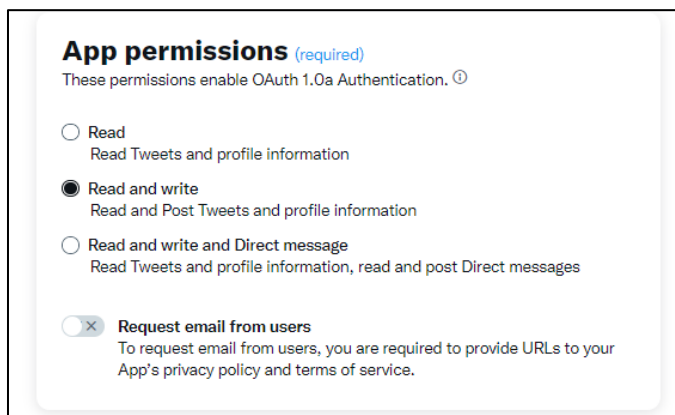
- 9.) After registering, you will be redirected to the Twitter developer dashboard. Under **Projects**, you will see the **Project App** that was created by default. Click the **gear icon** beside its app name.



- 10.) Scroll down and click **Set Up** under **User Authentication Settings**.



11.) Select “**Read and write**” under **App Permissions**. Next, Select “**Web App, Automated App or Bot**” under **Type of App**. For the Callback URI and Website URL under **App Info**, just put the link of the GitHub repository (<https://github.com/GugliolsStupid/CBDQ-Python>). Click **Save** at the end of the page. If the “Changing permissions might affect your App” popup shows up, click **Yes**.



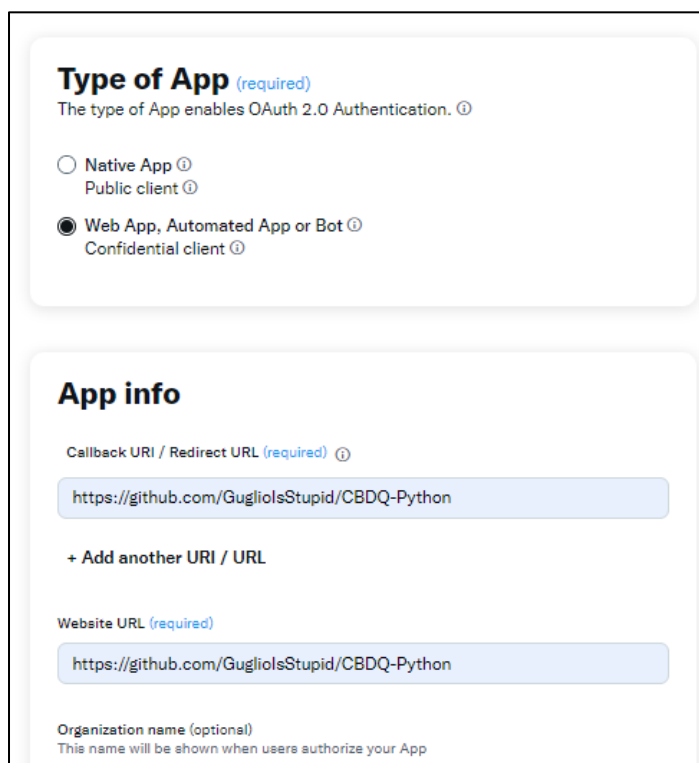
App permissions (required)
These permissions enable OAuth 1.0a Authentication. ⓘ

☐ Read
Read Tweets and profile information

☒ **Read and write**
Read and Post Tweets and profile information

☐ Read and write and Direct message
Read Tweets and profile information, read and post Direct messages

☐ Request email from users
To request email from users, you are required to provide URLs to your App's privacy policy and terms of service.



Type of App (required)
The type of App enables OAuth 2.0 Authentication. ⓘ

☐ Native App ⓘ
Public client ⓘ

☒ **Web App, Automated App or Bot** ⓘ
Confidential client ⓘ

App info

Callback URI / Redirect URL (required) ⓘ

<https://github.com/GugliolsStupid/CBDQ-Python>

+ Add another URI / URL

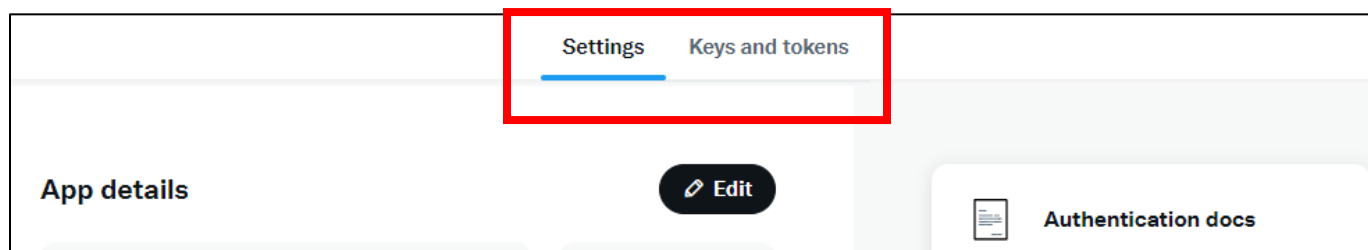
Website URL (required)

<https://github.com/GugliolsStupid/CBDQ-Python>

Organization name (optional)
This name will be shown when users authorize your App

12.) The next page should show your **OAuth 2.0 Client ID** and **Client Secret**. **Copy these** (in a notepad or Word doc) and **save it** somewhere in your PC.

13.) Going back to your default project menu settings, click **Keys and Tokens**.

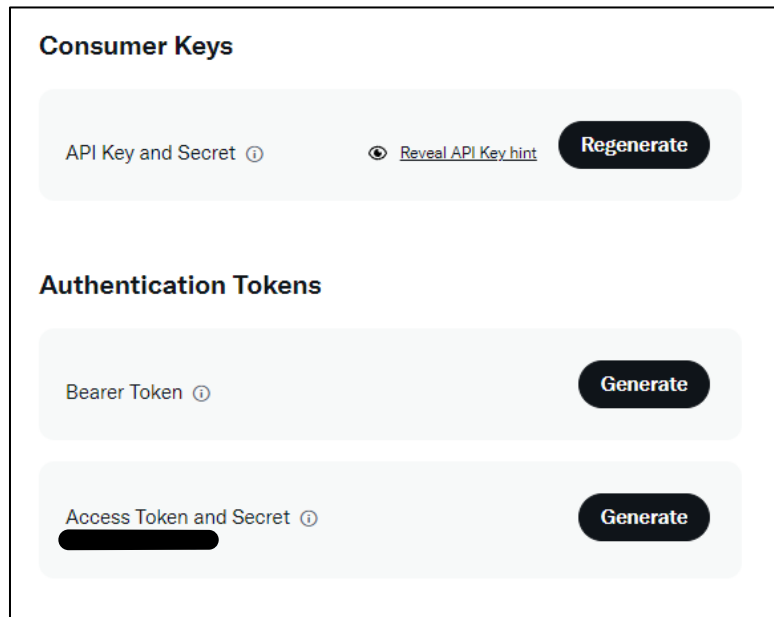


Settings **Keys and tokens**

App details [Edit](#)

Authentication docs

14.) This part will get the API keys that will be needed for our Repl project. Click **Regenerate** for the **Consumer Keys** and click **Generate** for the **Bearer Token** and **Access Token and Secret**. Copy these somewhere (like a notepad/Word doc) and save it in a secure location.



The screenshot shows a web interface for generating API keys. It is divided into two main sections: "Consumer Keys" and "Authentication Tokens".

Consumer Keys

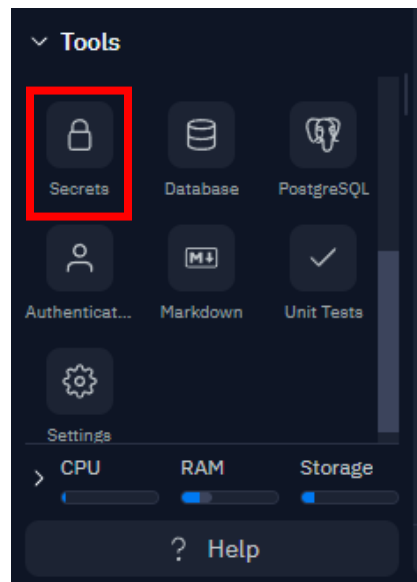
API Key and Secret ⓘ [Reveal API Key hint](#) **Regenerate**

Authentication Tokens

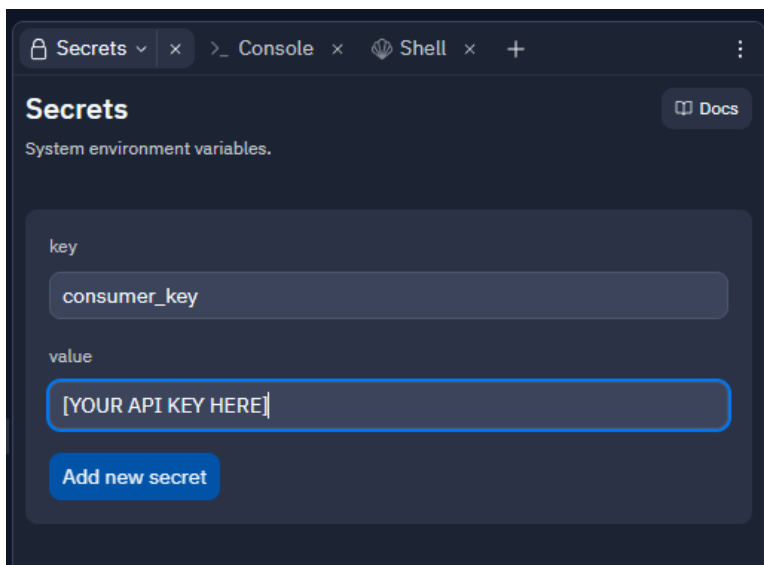
Bearer Token ⓘ **Generate**

Access Token and Secret ⓘ **Generate**

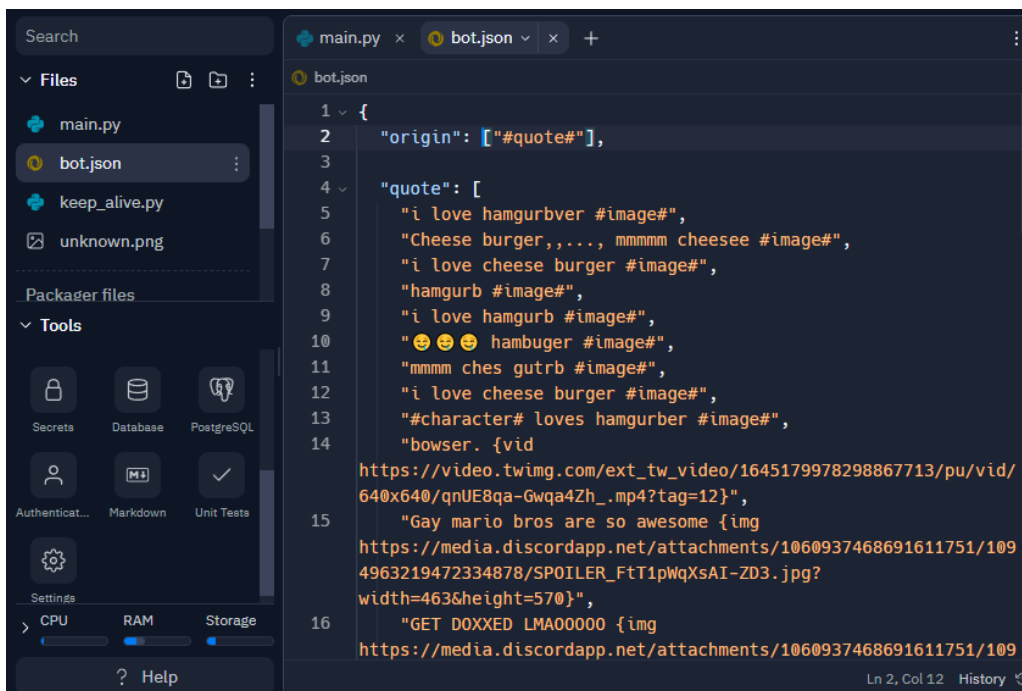
15.) Go back to the Repl Project from step 7. Under the **Tools** section, Find the “Secrets” button. Click it.



16.) On the Secrets section, create the key titles **consumer_key**, **consumer_secret**, **access_token**, **access_token_secret**, and **bearer_token**. Put the values that correspond to the keys that you got from step 14. (Note: *consumer_key* value is the API key, and *consumer_secret* is the API key secret.) Click “Add new secret” for each key.



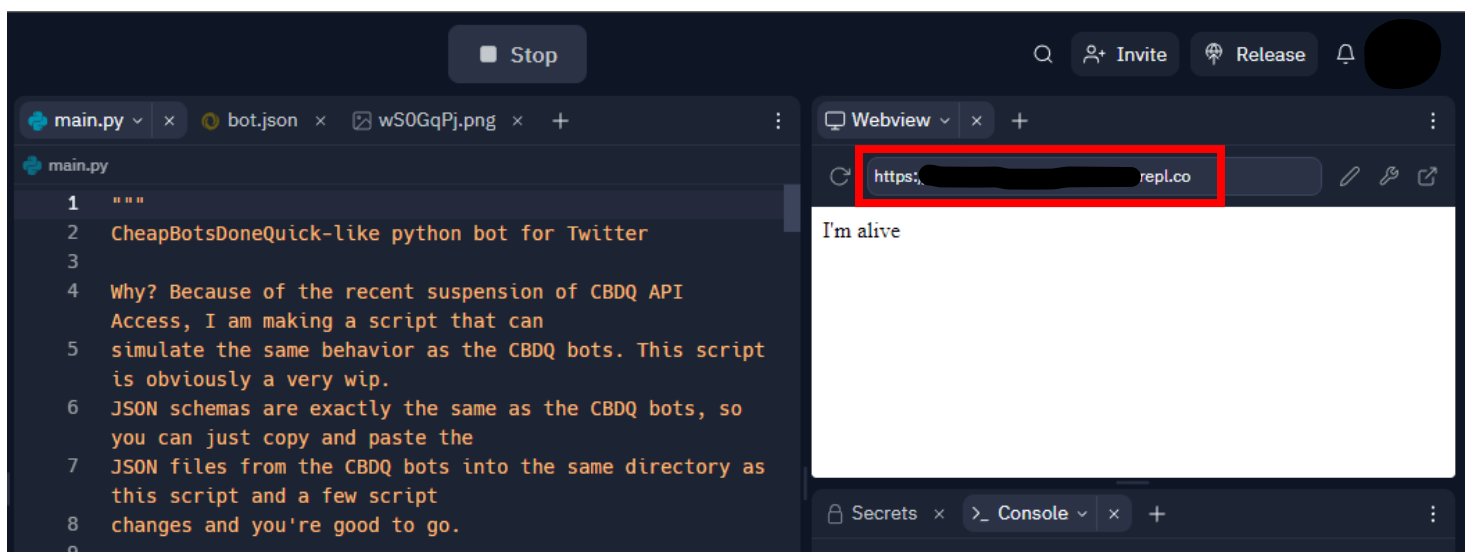
17.) API stuff part is done! Now go to your **bot.json** file. Put your CBDQ source in this file!



18.) Go to your main.py file. Search (ctrl + F) for **time_between_tweets**. Change the time interval for your bot to tweet! The unit used here is seconds, so you have to convert hours/minutes to seconds (e.g. 1 hour = 3600 seconds) for the bot to work.

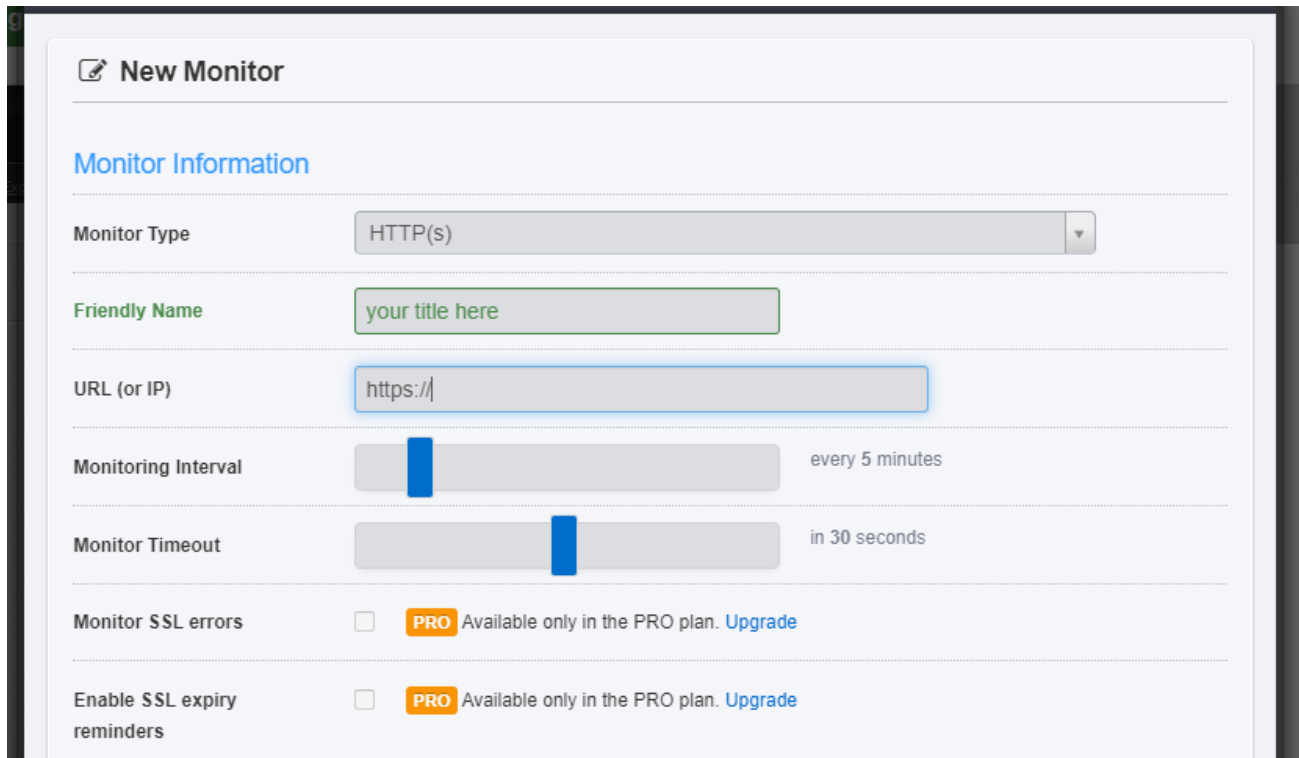
```
51
52 # The time between every new tweet
53 # This is in seconds, so 3600 is 1 hour
54 # Default is 30 minutes
55 # You can change this to whatever you want, but I recommend
  keeping it at 30-60 minutes
56 time_between_tweets = 1800
57
```

19.) Click **Run** on top of your Repl. If nothing goes wrong, a Webview tab will appear! A URL with **.repl.co** will also appear in the webview tab. Copy that URL.



20.) Now go to <https://uptimerobot.com/> and register for an account. After registering, go to the website's dashboard and click the "+ Add New Monitor" green button at the top left of the page.

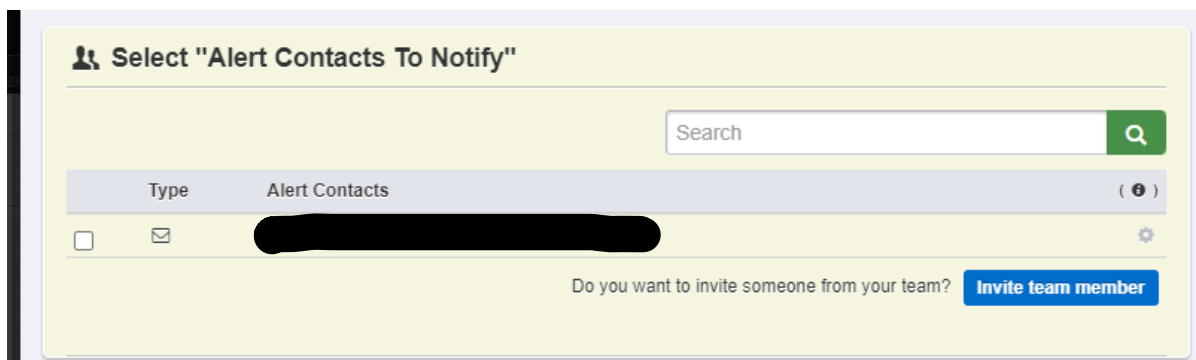
21.) Set the **Monitor Type** to **HTTP(s)**. Give it a title, and for the **URL (or IP)**, paste the **URL that you got from step 19**.




The screenshot shows the 'New Monitor' configuration page. It includes a 'Monitor Information' section with the following fields:

- Monitor Type:** A dropdown menu set to 'HTTP(s)'.
- Friendly Name:** A text input field containing 'your title here'.
- URL (or IP):** A text input field containing 'https://'.
- Monitoring Interval:** A slider set to 'every 5 minutes'.
- Monitor Timeout:** A slider set to 'in 30 seconds'.
- Monitor SSL errors:** A checkbox that is unchecked, with a note 'PRO Available only in the PRO plan. Upgrade'.
- Enable SSL expiry reminders:** A checkbox that is unchecked, with a note 'PRO Available only in the PRO plan. Upgrade'.

22.) (Optional) if you want to be notified if the bot goes down, check the checkbox with your email in the **Select "Alert contacts to notify"** section.



The screenshot shows the 'Select "Alert Contacts To Notify"' section. It includes a search bar and a table of alert contacts:

Type	Alert Contacts
<input type="checkbox"/>	 [Redacted email address]

At the bottom, there is a prompt: 'Do you want to invite someone from your team?' with an 'Invite team member' button.

23.) Click the “save” at the bottom of the page, and a monitor with the title that you set will appear on the dashboard! Start that monitor. It shows up as 0.00% at the start, but it will be 100% if you let it be. Like this:

