Getting Started with *GitHub Commit and 2FA Data Fetch* Automation

This automation was created to help you find data in a user-friendly fashion. With the help of this tool, you can easily:

- Find the latest commit to the given GitHub repository.
- Check if the user above is using 2-factor authentication.
- Understand which author wrote the highest word count along with the word count.

The step-by-step to how to use the automation and how it was created is as follows.

Step 1: List Commits

Important: Before getting started you must have a specific GitHub repository to reference. This is the foundation this automation is built upon.

1. The *owner* for this specific repository is found by clicking here. The name on the top left corner is owner of the repository.
2. Where it says *Repo* the repository name is entered. This is done so the automation knows which file to look at and where to find the information (in this case it is a name).
3. The number of pages at the bottom is set to 1 for simplicity.

Step 2: Print Latest Commit User

When the automation is run the end goal is to have the name of the latest commit author.

1. The word “commits” in this context becomes the automation’s code word for finding all the authors for this repository.
2. If the program doesn’t see any authors, the output will read “No commits found”. This is an off ramp for the automation to tell you “Something is wrong, the data asked for doesn’t exist”.
3. By writing “print” we are telling the automation what we want to see in the output. Here, we are asking the automation to find the author of the “commit” function and print their name in plain language.

Step 3: Fetch GitHub Authenticated User

Important: For this next step a GitHub account is necessary. If you do not have an account click here to go to the GitHub website and follow their steps. If a step-by-step guide is needed to create a GitHub account, one can be found here.

A personal access token on GitHub is designed to be used as an alternative to a password for authentication to GitHub when using GitHub’s services. These tokens are created to access GitHub resources on your behalf. The following steps can be taken to obtain this token for this automation:

- Sign into GitHub
- Top right corner, click on **profile photo**
- Click **settings**
- Left sidebar, click **developer settings**
- Left sidebar under personal tokens, click **Tokens (classic)**
- Select **Generate new token**

Next an HTTP request is executed. This is a request for a response protocol between a client and server. This means the automation is asked to communicate with a server to answer questions asked. In this case the question is “does the selected user have 2-factor authentication set up?”

Step 4: Check User 2FA

This step tells the automation to look at the above information, when we asked it a question, and give an answer. The answer comes in the form of a print statement. The statement will either verify or deny the use of the 2-factor authentication function.

Step 5: Fetch Users

The goal here is to use the same HTTP method as the previous step while asking a different question. The next task asked of the automation is to fetch the users of a given data sheet. The data sheet is in the form of a JSON file and by calling it, it can be referenced and parsed.

Step 6: Fetch Posts

As done previously, the method of HTTP stays the same, however, this time the goal is to reference the posts for a word count. The automation is going to be asked to find the author of the longest post along with their word count.

Step 7: Output Author with Highest Word Count

The automation executes the following steps using the data from steps 5 and 6. With the data the automation will parse and output the answer to the question which user wrote the lengthiest post and how many words were written.

Below are some important things to note when looking at the code.

- Line 1: “users” is referenced while calling the data from the output of step 5.
- Line 5: “posts” is referenced while calling the data from the output of step 6.
- Line 9: The automation is asked to count the words of the key word “posts”.
- Line 19: The goal is to sort the users by numerical wordcount from greatest to least.
- Line 24: Lastly, the automation is asked to print the name with their wordcount.

This is the end of the automation.