



Rate Limiter,

Version: 1.0 R1

Released: Monday, June 04, 2018

[Objective](#)

[Deliverables](#)

OBJECTIVE

Identify a Developer's profile on GitHub, when following attributes are provided:

1. First Name
2. Last Name
3. Location

Once the profile has been identified, extract following attributes from the profile:

1. Public profile details
2. Name of repositories where they are contributing
3. No. of commits made by that developer on those repositories

The solution should be able to take a list of developer information, rather than just 1 developer at a time. The data should be gathered in the minimum amount of time without wasting resources and violating any terms & conditions mentioned by the host.

GitHub applies rate limits to the use their API, this rate limit differs between an anonymous user and a logged in user. The rate limits are in place to avoid any type of D/DOS attacks at the same time to avoid malicious scanning of user data. This rate limit needs to be honoured to avoid account lockouts, this could also lead to the IP getting flagged and blocked by the service provider. Hence we need to gather the data in a way that application operates in the given rate limits as specified by the host.

DELIVERABLES

1. Working solution. Provide source code with build and execution instructions.
2. Document the approach take, including (but not limited to) understanding of rate limits, your approach to manage it, information being processed and what it represents.
3. Choose ideal output format to store and represent the collected results
4. State any assumptions made
5. Suggest any additional parameters that could be additionally taken as input to improve the accuracy of the solution
6. State any improvements that you would like to make if more time was available for implementation.
7. Below is the sample input information. You need to attach the output of your program taking the below users as sample input
8. What

EVALUATION

Your submission will be evaluated on completing all the items specified in deliverables list. Once the submission is made, we will arrange a follow up discussion to understand from you the solution itself and ask any question that come out of the submission you make.

Important: To make evaluation accurate and less time consuming please take note of following.

1. **Code formatting:** Please indent the code (using tabs or space) and use camel case when defining variables and functions.
2. **Documentation:** Please document the code where necessary and document it just enough. Excessive documentation is worse than no documentation.
3. **Overall summary:** Please explain on how to build and execute the code. How and what inputs to be passed and where output can be seen. A general execution flow will also be good for the evaluator to

understand the solution that is being provided. As mentioned above, this needs to be just enough to make evaluation effective and avoid excessive documentation.

SAMPLE INPUT

First Name	Last Name	Location
Chunky	Garg	Gurgaon
Sharvi	Verma	India
Rahul	Gaur	India
Abhishek	Parikh	
Anubhav	Saxena	
Shiva	Tiwari	Bangalore
Sumit	Yadav	New Delhi
Varun	Arora	India
Anshika	Singh	India
Salman	Kagzi	
James	Golick	New York
Dave	Fayram	San Francisco, CA
Nishan	Perera	London