Machine Learning/Data Science Hiring Challenge

Thank you for applying to Verloop!

This is to help us understand your programming experience. We understand that this does not capture your Machine Learning/Data Science expertise. We will discuss that separately with a ML task to guide our conversation.

Getting Started

For this round, you have to write a simple API server, which fetches data from Github, serializes in a certain format.

It involves few steps:

1. You need to get a unique ID, which identifies your submission
2. You need to make a REST API call to Github's API service and fetch the data
3. Measure the response time for your endpoint

Get the Question ID

Send a POST request to `https://hiring.verloop.io/api/github-challenge/start/` with a custom header and body (in JSON).

Header

Your POST request should have a header `x-verloop-password`, which is md5 digest (in hexadecimal) of your email.

hint: If your email is `avinash@verloop.io`, then value of `x-verloop-password` will be `d68aa169f74a4af3e78a12690aecc071`.

Body

Your POST body should be in JSON with appropriate headers. It should have following fields:

- email (string)
- name (string)

- angel_list (string)
- github (string)

The field `github` here is optional.

Here is an example:

```
{
    "email": "youremail@gmail.com",
    "name": "your name here",
    "angel_list": "https://angel.co/your-profile-id",
    "github": "https://github.com/your-profile-id"
}
```

Response Time

Figure out how to measure response time for your end point

On successful request, you will receive a response as following:

```
{
    "response": {
        "question_id": "48b4e5074bad4463b641e60934c839fc",
        "github_id": "avinassh"
    }
}
```

Make note of the `question_id` , this will be used to identify the submissions we receive at Verloop. You can ignore the `github_id` field for now.

Github API

You will need to write a simple API which returns top 3 repositories of an organisation in Github by stars.

API Details

We will send a `POST` request to your API at `/repos` with payload (in JSON):

```
{
      "org": "github-organization-id"
}
```

and the API should return a response in following format (JSON):

```
{
    "results": [
        {"name":"rick", "stars": 100},
        {"name":"morty", "stars": 98},
        {"name":"jerry", "stars":2},
    ]
}
```

Metrics

Response Time: Share your findings on response time and the code that you use for the same

Extras

1. There's no restriction on the programming language, libraries or the web framework
2. It is okay to Google or Bing (if that's your thing)
3. Feel free to ask us any questions you have and clarifications you need
4. Think through the edge cases
5. Write good, useful docstrings and comments

Bonus

1. Write functional and units tests
2. Deploy this as a microservice on a cloud and share link
3. Profile your code. What are the bottlenecks?
4. In addition to response time, measure requests/second that your endpoint can handle
5. Add logging

Submission

Once you are done, email us the solution and findings/notes/logs in a zip file titled `<your-name>-hiring-challenge.zip` Don't forget to mention the `question_id` in the email id.