**CenturyLink Coding Challenge**

**Overview**: The purpose of this coding challenge is to test your ability to develop software for the CenturyLink Cloud organization. For this coding challenge, you’re free to use any coding language you’d like.

**Requirements**:

1. Write an API endpoint that accepts a GitHub ID and returns Follower GitHub ID’s (up to 5 Followers total) associated with the passed in GitHub ID. Retrieve data up to 3 levels deep, repeating the process of retrieving Followers (up to 5 Followers total) for each Follower found. Data should be returned in JSON format.
2. Code should be checked into a public GitHub repository, so that the CenturyLink team members can review your code at any time.
3. A readme file should be included and checked into GitHub with instructions on how to execute and test your API.
4. Bonus credit: API endpoint is publicly accessible and fully functional, so that CenturyLink team members can execute and test your API at any time.

**Bonus Challenge:**

The bonus challenge NOT required!

1. Write an API endpoint that accepts a GitHub ID and retrieves the Repository names (up to 3 Repositories total) associated with the passed in GitHub ID, along with the Stargazer GitHub ID’s (up to 3 Stargazers total) associated with each Repository. Retrieve data up to 3 levels deep, repeating the process of retrieving the associated Repositories (up to 3 Repositories total) for each Stargazer (up to 3 Stargazers total) found. Data should be returned in JSON format.
2. See requirements 2, 3, and 4 above.

**Tips**:

1. You’ll need a GitHub account (which are free) to complete this coding challenge.
2. GitHub provides API’s for retrieving Followers, Repositories, and Stargazers. Please see the following GitHub API documentation:
   1. <https://developer.github.com/v3/users/followers/>
   2. <https://developer.github.com/v3/repos/>
   3. <https://developer.github.com/v3/activity/starring/>