**Connect to GitHub, Utility using Python**

Implemented the following

PyGithub is a python wrapper for the GitHub REST API which aims to provide a more conventionally pythonic way of controlling your GitHub. It provides a higher-level API containing a number of convenience functions.

We like to use PyGithub to automate our GithHub. Here are some of the things we used it for:

========GitHub API list===========

* search\_github
* userInfo
* getmyRepos
* getRepoTopic
* getStar
* openIssue
* getLabels
* getContents
* getContentFile
* createFile
* updateFile
* deleteFile
* getClones
* markNotifs
* branchlist
* getbranch
* getheadcomm
* statuscheck
* createcommstatus
* getcommdate
* getissue
* createissue

1. **search\_github**

Lets you search the github based on various quiries for example language say python flask javascript

1. **userInfo**

Gives the Information about Logged in User. This function gives username and Full Name.

1. **getmyRepos**

Lists all the repositories by the logged in user.

1. **getRepoTopic**

Gets the topic of the repository.

1. **getStar**

Gets the number of stars on the Repo.

1. **openIssue**

Opens Issue.

1. **getLabels**

Get all the Labels.

1. **getContents**

Lists the Contents.

1. **getContentFile**

Gets a particular contenct File.

1. **createFile**

Creates a particular file in the repository.

1. **updateFile**

Updates the existing file in the repo.

1. **deleteFile**

Deletes the specified file in the repo.

1. **getClones**

Get the number of clones and some other info.

1. **markNotifs**

Marks all notifications as read.

1. **Branchlist**

Lists the branches of the repo.

1. **Getbranch**

Gets a particular branch.

1. **Getheadcomm**

Gets info about the head commit.

1. **Statuscheck**

See required status checks of a branch.

1. **Createcommstatus**

Create commit status check, you can provide sha for different commit deagault has already been provided.

1. **Getcommdate**

Gets the info like date and author of a commit.

1. **Getissue**

Gets an Issue.

1. **Createissue**

Creates an Issue.

**To Implement the Github using REST API in Python follow the below steps**

1.Create an Account on Github.

2.Create a repository for some action.

Eg: <https://github.com/dev-manojm/python-utility>

3. Create an Api-Token

Create an API token from your GitHub account:

Go to security settings>API tokens

1.Click Create API token.

2.From the dialog that appears, enter a memorable and concise Label for your token and click Create.

3.Click Copy to clipboard, then paste the token to your script, or elsewhere to save.

4. You are good to go!

**On your Laptop / Computer perform following steps**

1. Install Python 3.4 and above version on System

2.Install a Python code editor (I’m using Pycharm Community Edition )

3.Install required packages and extensions on editor and system

4.Open py-github.py (filename.py) file in your editor

5.Check API token and enter it accordingly as per you

6.If required packages and extensions are not installed

Run command (windows 10)

py -m pip install PyGithub

7.Then run command py filename.py (py-github.py) (windows 10)

Note : if you are working on different operating systems there are different commands other than windows 10 try “python “ instead of “py”.