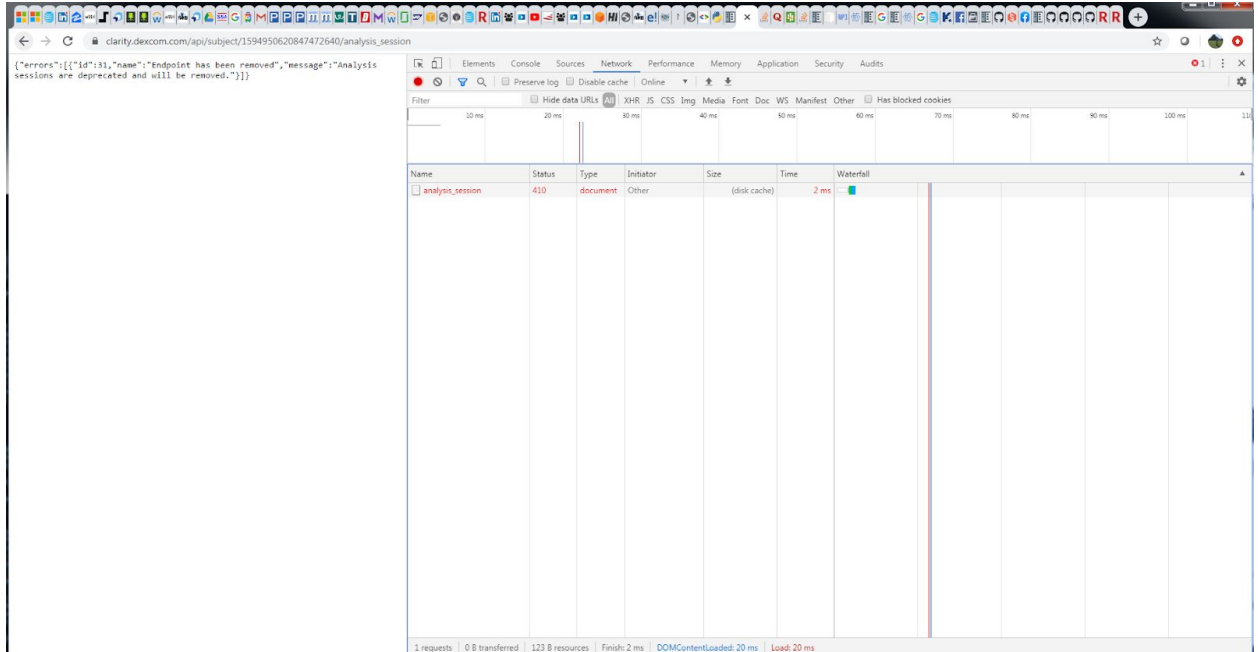


# 1. Reaching Endpoint in Browser

Executing manual steps to reach the endpoint in Chrome Browser, we see that the error message indicates that the document cannot be reached, because the 'Endpoint has been removed':

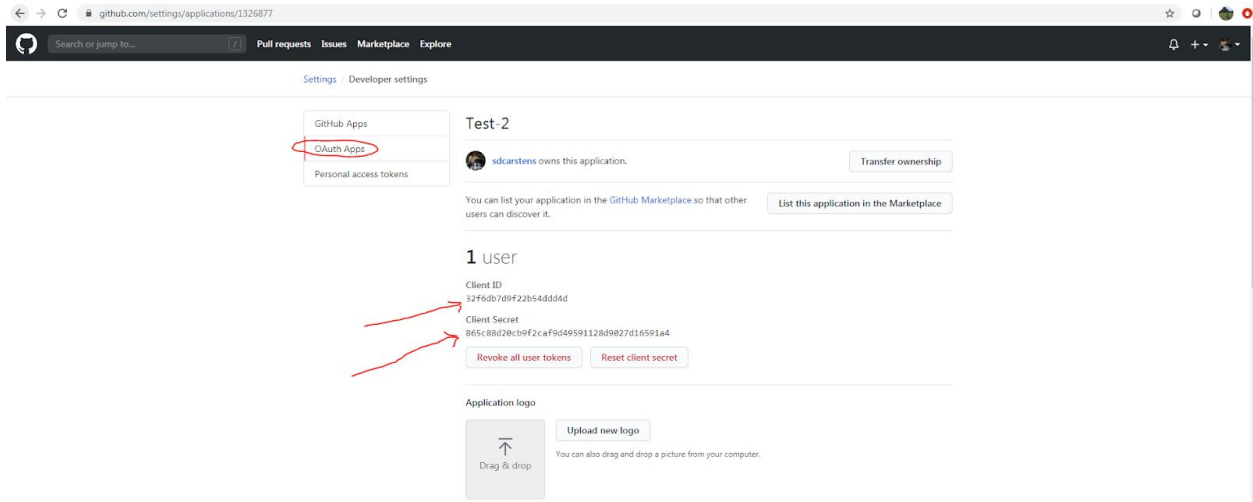


# 2. Managing OAuth2 session.

In order to establish the OAuth2 session, we need to have the 'client\_id' and 'client\_secret' of the application. Since these are not available for the Dexcom application, the example below establishes an OAuth2 session with my GitHub application:

```
#####  
# OAuth2 login example with GitHub application  
#  
# Based on workflow shown here:  
https://requests-oauthlib.readthedocs.io/en/latest/examples/github.html  
#####  
import time  
from requests_oauthlib import OAuth2Session
```

```
# Credentials you get from registering a new GitHub application
client_id = '32f6db7d9f22b54ddd4d'
client_secret = '865c88d20cb9f2caf9d49591128d9027d16591a4'
print('client_id:..... ',client_id)
print('client_secret:..... ',client_secret)
print(' ')
```

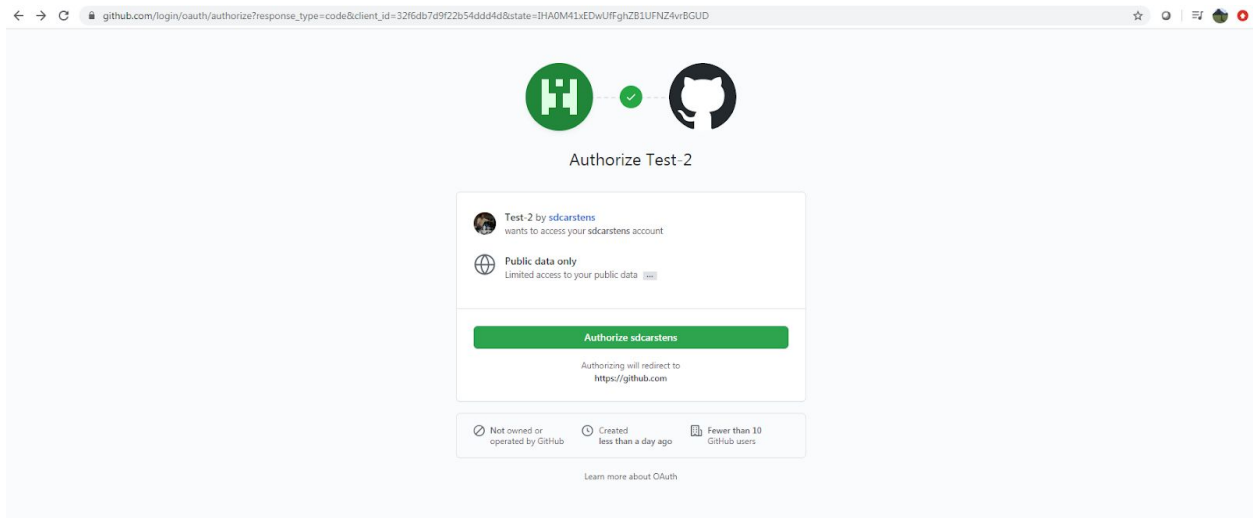


```
# OAuth endpoints given in the GitHub API documentation
authorization_base_url = 'https://github.com/login/oauth/authorize'
token_url = 'https://github.com/login/oauth/access_token'
```

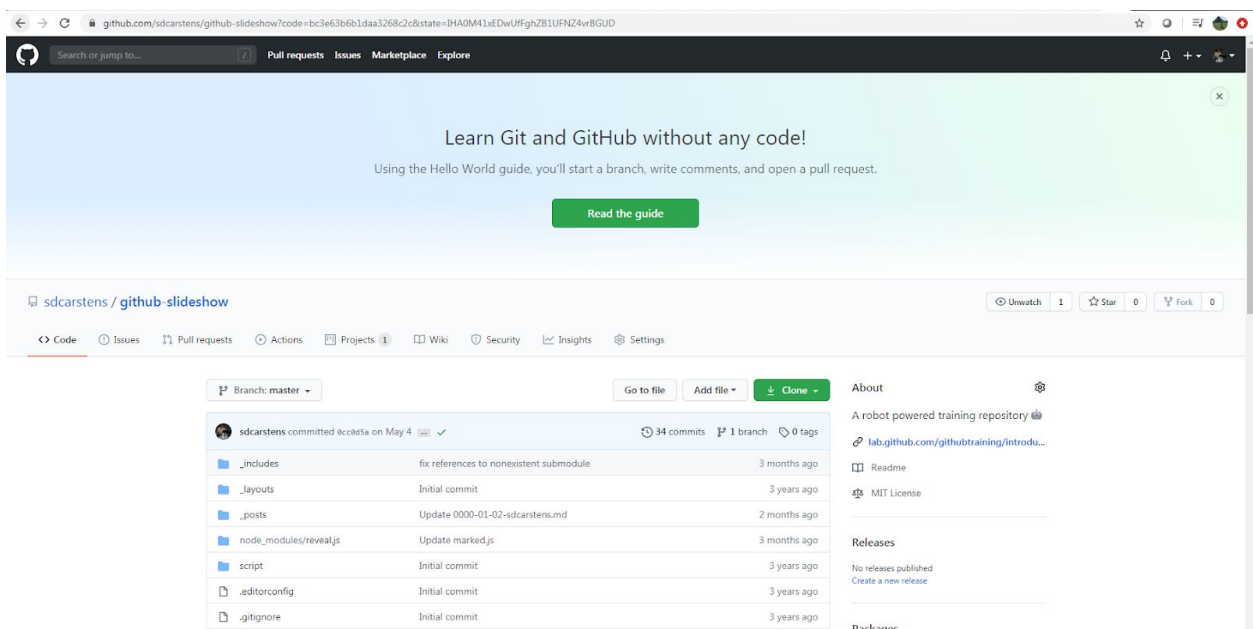
```
# Establish OAuth session to GitHub Application
github = OAuth2Session(client_id)
print('github:..... ',github)
print(' ')
```

# Redirect user to GitHub for Authorization

```
authorization_url, state = github.authorization_url(authorization_base_url)
print( 'Please go here and authorize,', authorization_url)
print(' ')
```



# Get the authorization verifier code from the callback url  
redirect\_response = input('Paste the full redirect URL here:')



```
# Fetch the access token
github.fetch_token(token_url, client_secret=client_secret,
authorization_response=redirect_response)
```

```
# Fetch a protected resource, i.e. user profile
r = github.get('https://api.github.com/user')
print('Protected Resource Content:..... ', r.content)
```

```
C:\b-python\Scripts>python myTest3.py
client_id:..... 32f6db7d9f22b54ddd4d
client_secret:..... 865c88d20cb9f2caf9d49591128d9027d16591a4

github:..... <requests_oauthlib.oauth2_session.OAuth2Session object at 0x0044E6A0>

Please go here and authorize, https://github.com/login/oauth/authorize?response_type=code&client_id=32f6db7d9f22b54ddd4d&state=hcF10gd1IqU5VbCMfJQsKAA8cpwTVL

Paste the full redirect URL here:https://github.com/sdcarstens/github-slideshow?code=aec705440fc5f055fd8a&state=hcF10gd1IqU5VbCMfJQsKAA8cpwTVL

Protected Resource Content:..... b'{"login":"sdcarstens","id":64766243,"node_id":"MDQ6VXNlcjY0NzY2MjQz","avatar_url":"https://avatars1.githubusercontent.com/u/64766243?v=4","github.com/sdcarstens","followers_url":"https://api.github.com/users/sdcarstens/followers","following_url":"https://api.github.com/users/sdcarstens/following{/other_user}","gists_url":"https://api.github.com/users/sdcarstens/gists","starred_url":"https://api.github.com/users/sdcarstens/starred{/owner}{/repo}","subscriptions_url":"https://api.github.com/users/sdcarstens/subscriptions","organizations_url":"https://api.github.com/users/sdcarstens/orgs","events_url":"https://api.github.com/users/sdcarstens/events{/privacy}","received_events_url":"https://api.github.com/users/sdcarstens/received_events","type":"User","site_admin":false,"email":null,"hireable":null,"bio":"Software Systems Maestro - wrangling ornery computer science projects into beautiful music with a little Python, some Splunk, and a pinch of...","following":0,"created_at":"2020-05-04T04:51:29Z","updated_at":"2020-06-30T00:14:15Z"}'

C:\b-python\Scripts>
```

### 3. Test For Interacting with API HTTP Request calls

Below is the output from a Test which creates a Session object and executes POST and GET requests to the Dexcom REST API. The error message returned again says that the 'Endpoint has been removed'.

```
C:\Windows\system32\cmd.exe
G:\h-python\Scripts>python myTest2.py
[options]
Config option URL: https://aami.dexcom.com
Config option Username: nilepatst001
Config option Password: Password01
Config option POST_URL: https://clarity.dexcom.com/api/subject/1594950628047472648/analysis_session
Setup here
Headers: <<User-Agent: 'python-requests/2.24.0', 'Accept-Encoding': 'gzip, deflate', 'Accept': '*/*', 'Connection': 'keep-alive'>>
Auth: None
Response: <Response [200]>
Response.text: <<DOCTYPE html><html ng-app="userManagementWebApp" ng-strict-di><head><!--if gte IE 9 ! !IE 1>
<script async src="https://www.googletagmanager.com/gtag/js?id-UA-130780636-5"></script>
window.dataLayer = window.dataLayer || [];
function gtag() < dataLayer.push(arguments); >
gtag('js', new Date());
gtag('config', 'UA-130780636-5');
</script>
<!--ifend--><meta charset="UTF-8"><meta http-equiv="X-UA-Compatible" content="IE=edge"><meta http-equiv="Content-Type" content="text/html; charset=utf-8"><meta name="v1
tcut icon" href="assets/favicon.ico" type="image/vnd.microsoft.icon"><title>Dexcom - Account Management</title><link href="assets/themes/dexcom-new/css/popup.css?v=15889280436
3692804366" rel="stylesheet"><link href="assets/themes/dexcom-new/css/dexcom.css?v=158892804366" rel="stylesheet"><link href="less/app.min.css?v=158892804366" rel="stylesheet
type="text/javascript" src="libs/jquery/jquery-2.1.3.min.js?v=158892804366"></script><script type="text/javascript" src="assets/scripts/once.js?v=158892804366"></script><scri
4366"></script><script type="text/javascript" src="assets/scripts/menu_helper.js?v=158892804366"></script><script type="text/javascript">function autoResize(frameObj) {
var newHeight;
if (document.getElementById (
newHeight = frameObj.contentWindow.document.body.scrollHeight;
}
frameObj.height = newHeight) + "px";
}</script></head><body ng-controller="appController" class="html not-front not-logged-in page-user page-user-register context-user dexcom-page-user-register no-
rfix" id="page"><ng-include src="'app/partials/header.html'"></ng-include><section id="section-content" class="section section-content"><div id="landing-page" ng-show="viewData
zone zone-content clearfix" ng-view><div><div><div id="content-pages" ng-show="viewData.isLandingPage==false"><div id="zone-content" class="zone zone-content wait clear
dex-ajaxloader"></div></div></div><div id="zone-content" class="zone zone-content clearfix container-12" ng-view></div></div>
<script src="libs/platform/platform.js?v=158892804366"></script><script src="libs/dex/BrowserCheck/BrowserCheckValidatop.js?v=158892804366"></script><script src="libs/underscore-1.
.min.js?v=158892804366"></script><script src="libs/jquery/jquery-2.1.3.min.js?v=158892804366"></script><script src="libs/es6-promise/es6-promise.min.js?v=158892804366"></sc
heClient-out2/client-out2.js?v=158892804366"></script><script src="app/dexEnvironment.min.js?v=158892804366"></script><script src="app/components.min.js?v=158892804366"
"></script><script src="libs/dex/UserRegistration/dex.UserRegistration.min.js?v=158892804366"></script><script src="app/index.min.js?v=158892804366"></script></body></html>

Response.Status_Code: 200

Response: <Response [401]>
Response.text: The resource requires authentication, which was not supplied with the request
Response.Status_Code: 401

data: <<errors: [{id: 31, name: 'Endpoint has been removed', message: 'Analysis sessions are deprecated and will be removed.'}]>>
analysisSessionId: 31
C:\h-python\lib\unittest\case.py:633: ResourceWarning: unclosed <ssl.SSLSocket fd=396, family=AddressFamily.AF_INET, type=SocketKind.SOCK_STREAM, proto=0, laddr=('192.168.1.2',
method()
ResourceWarning: Enable tracemalloc to get the object allocation traceback
C:\h-python\lib\unittest\case.py:633: ResourceWarning: unclosed <ssl.SSLSocket fd=432, family=AddressFamily.AF_INET, type=SocketKind.SOCK_STREAM, proto=0, laddr=('192.168.1.2',
method()
ResourceWarning: Enable tracemalloc to get the object allocation traceback
tearDown here.
.
Ran 1 test in 0.913s
OK
```

The Python code can be found here: <https://github.com/sdcarstens/Dexcom-Test2>

```
#####
#
#   Test 2 - API Automation with 'requests' library and Session
#
#####

import time
import unittest
import configparser
from selenium import webdriver
from getpass import getpass

import locators
import element
import page

import requests
import json
from requests.auth import HTTPBasicAuth
```

```

#
# Read in config options for URL, Username, Password
#
config = configparser.RawConfigParser()
config.read('myTest1.conf')

config_url      = config.get('options', 'url')
config_username = config.get('options', 'username')
config_password = config.get('options', 'password')
config_post_url = config.get('options', 'post_url')

print(config.sections())
print("Config option URL: ", config_url)
print("Config option Username: ", config_username)
print("Config option Password: ", config_password)
print("Config option POST_URL: ", config_post_url)

```

```

class DexcomApiAutomation(unittest.TestCase):
    """Use the Sessions object class and unittest framework to test Dexcom API endpoint."""

    def setUp(self):
        print("setUp here.")
        #self.driver = webdriver.Chrome()
        #self.driver.get(config_url)

    def test_search_in_python_org(self):
        """
        Tests the ability to POST to
        https://clarity.dexcom.com/api/subject/1594950620847472640/analysis_session
        """

        # Create new session object
        newSession = requests.Session()

        #
        # Login to Dexcom URL
        #
        # Expect status code 200, successful login
        #
        data = {"login": config_username, "password": config_password}
        auth = HTTPBasicAuth(config_username, config_password)
        r = newSession.post(url=config_url, data=data, auth=auth)

        print("Headers:::::::::: ", newSession.headers)
        print("Auth:::::::::: ", newSession.auth)
        print("Response:::::::::: ", r)

```

```

print("Response.text::::::::::::: ",r.text)
print("Response.Status_Code::::::::: ",r.status_code)
assert r.status_code == 200, "Expected status code should be 200"
print("-----")
print(" ")

#
#   Now use the POST request to test API endpoint
#
#   Expect response status_code 401 for invalid authentication
#
data = {'sender': 'Alice', 'receiver': 'Bob', 'message': 'We did it!'}
headers = {'Content-type': 'application/json', 'Accept': 'text/plain'}
r = newSession.post(config_post_url, data=json.dumps(data), headers=headers,
auth=auth)
print("Response::::::::::::: ",r)
print("Response.text::::::::::::: ",r.text)
print("Response.Status_Code::::::::: ",r.status_code)
assert r.status_code == 401, "Expected status code should be 401"
print("-----")
print(" ")

#
#   Now use the GET request to test API endpoint
#
#   Expect response: {'errors': [{'id': 31, 'name': 'Endpoint has been removed',
'message': 'Analysis sessions are deprecated and will be removed.'}]}
#
r = newSession.get(url = config_post_url, auth=auth)
data = r.json()
print("data::::::::::::: ",data)
analysisSessionId = data['errors'][0]['id']
print("analysisSessionID::::::::: ",analysisSessionId)
assert analysisSessionId != 'None', "The value of analysisSessionId should not be
None"

def tearDown(self):
    print("tearDown here.")
    #self.driver.close()

if __name__ == "__main__":
    unittest.main()

```