



Btech

21st March 2018

Scrap the GSOC Website

A Project Report

By

Harshit Bhatia (16BIT0109)

Under the guidance of

GDG 2 Credit Course Mentors

Acknowledgment

I sincerely thank Dr.G.Viswanathan - Chancellor, VIT University, for creating an opportunity to use the facilities available at VIT. I also thank GDG Team for giving us the opportunity to do this project. I also thank the Dean and the entire department of Information Technology, School of Information Technology and Engineering, for giving us this opportunity.

Introduction

Task

Create an API using Flask or any framework you are comfortable with (Flask is easy to learn) and scrape the GSOC website to fetch details of all the organizations in Google summer of codes and in the api send their name, link to their website, description (which comes on clicking learn more), The technologies they use and their contact email.

Example-

The api link is say `http://localhost:8080/orgs` . The user should get a list of organizations and each member of the list should look like this or should have same kind of schema.

```
{  
  organization: 3DTK,  
  link: http://threedtk.de/  
  description: The 3D Toolkit ....,  
  technologies: [c/c++, cmake, opencv, ros, boost], c contact:  
  johannes.schauer@uni-wuerzburg.de  
}
```

Website

<https://summerofcode.withgoogle.com/archive/2017/organizations/>

FrameWorks used- Flask(Python webframework)

Library used- BeautifulSoup(Python library)

Text Editor- Sublime Text and to run the codes I used cmd

Code:-

Scraping.py(code for fetching the details from the website)

```
from bs4 import BeautifulSoup as bsoup
```

```
import requests
```

```
my_url='https://summerofcode.withgoogle.com/archive/2017/organizations/'
```

```
original = "https://summerofcode.withgoogle.com"
```

```
response = requests.get(my_url)
```

```
html = response.content
```

```
soup = bsoup(html,"html.parser")
```

```
organizations = soup.findAll("li",{ 'class': 'organization-card__container'})
```

```
for organization in organizations:
```

```
    page_url=organization.find('a',{ 'class': 'organization-card__link'})
```

```
    organization_name=organization['aria-label']
```

```
about=organization.find('div',{'class':'organization-card__tagline font-black-54'})
```

```
about=about.text
```

```
page_link=original+page_url['href']
```

```
page = requests.get(page_link)
```

```
if page.status_code != 200:
```

```
    break
```

```
page_link=original+page_url['href']
```

```
response1 = requests.get(page_link)
```

```
html1=response1.content
```

```
soup1=bsoup(html1,"html.parser")
```

```
organization_link=soup1.find("a',{'class':"org__link"})
```

```
organization_link=organization_link.text
```

```
technologies=soup1.findAll("li',{'class':"organization__tag  
organization__tag--technology"})
```

```
tech = []
```

```
for t in technologies:
```

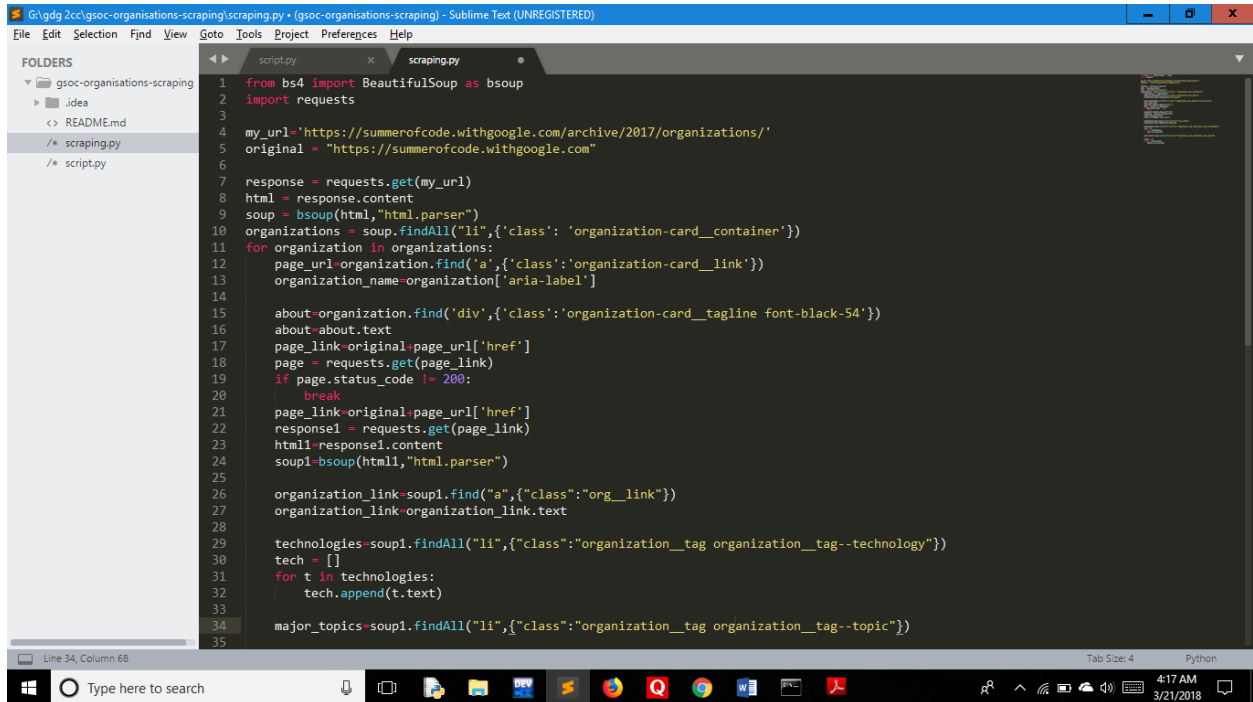
```
    tech.append(t.text)
```

```
major_topics=soup1.findAll("li',{'class':"organization__tag  
organization__tag--topic"})
```

```
topics = []
```

```
for q in major_topics:
```

```
    topics.append(q.text)
```



```
1 from bs4 import BeautifulSoup as bsoup
2 import requests
3
4 my_url='https://summerofcode.withgoogle.com/archive/2017/organizations/'
5 original = "https://summerofcode.withgoogle.com"
6
7 response = requests.get(my_url)
8 html = response.content
9 soup = bsoup(html,"html.parser")
10 organizations = soup.findAll("li",{'class': 'organization-card_container'})
11 for organization in organizations:
12     page_url=organization.find('a',{'class':'organization-card_link'})
13     organization_name=organization['aria-label']
14
15     about=organization.find('div',{'class':'organization-card_tagline font-black-54'})
16     about=about.text
17     page_link=original+page_url['href']
18     page = requests.get(page_link)
19     if page.status_code != 200:
20         break
21     page_link=original+page_url['href']
22     response1 = requests.get(page_link)
23     html1=response1.content
24     soup1=bsoup(html1,"html.parser")
25
26     organization_link=soup1.find("a',{'class':'org__link'})
27     organization_link=organization_link.text
28
29     technologies=soup1.findAll("li',{'class':'organization_tag organization_tag--technology'})
30     tech = []
31     for t in technologies:
32         tech.append(t.text)
33
34     major_topics=soup1.findAll("li',{'class':'organization_tag organization_tag--topic'})
35
```

Script.py(Main File used Flask api to send in json format)

```
from flask import Flask,jsonify
```

```
from flask import render_template
```

```
import requests
```

```
from bs4 import BeautifulSoup as bsoup
```

```
import json
```

```
app=Flask(__name__)
```

```

@app.route('/orgs')
def orgs():
    my_url='https://summerofcode.withgoogle.com/archive/2017/organizations/'
    original = "https://summerofcode.withgoogle.com"

    result = list()

    response = requests.get(my_url)
    html = response.content
    soup = bsoup(html,"html.parser")
    organizations = soup.findAll("li",{ 'class': 'organization-card__container'})

    counter = 0
    for organization in organizations:
        page_url=organization.find('a',{ 'class': 'organization-card__link'})
        organization_name=organization['aria-label']

        about=organization.find('div',{ 'class': 'organization-card__tagline
font-black-54'})
        about=about.text
        page_link=original+page_url['href']
        page = requests.get(page_link)
        if page.status_code != 200:

```

```

        break

    page_link=original+page_url['href']
    response1 = requests.get(page_link)
    html1=response1.content
    soup1=bsoup(html1,"html.parser")
    organization_link=soup1.find("a",{"class":"org__link"})
    organization_link=organization_link.text

    technologies=soup1.findAll("li",{"class":"organization__tag
organization__tag--technology"})
    tech = []
    for t in technologies:
        tech.append(t.text)

    major_topics=soup1.findAll("li",{"class":"organization__tag
organization__tag--topic"})

    topics = []
    for q in major_topics:
        topics.append(q.text)

    counter += 1
    print(counter)
    result.append({
        'organization_name': organization_name,

```



```

        'description': about,
        'link': organization_link,
        'technologies': tech,
        'topics': topics
    })

    # if (counter==5):
    #     break

    print("~")
    print(result)

    return json.dumps(result)

if __name__ == "__main__":
    app.run(debug=True)

```

The screenshot shows a Sublime Text editor window titled "G:\gdg 2cc\groc-organisations-scraping\script.py - (groc-organisations-scraping) - Sublime Text (UNREGISTERED)". The left sidebar displays the "FOLDERS" panel with a tree view showing the project structure: "groc-organisations-scraping" (containing ".idea", "README.md", "scraping.py", and "script.py"). The main editor area shows the "script.py" file with the following code:

```

1 from flask import Flask, jsonify
2 from flask import render_template
3 import requests
4 from bs4 import BeautifulSoup as bsoup
5
6 import json
7
8 app=Flask(__name__)
9
10 @app.route('/orgs')
11 def orgs():
12     my_url='https://summerofcode.withgoogle.com/archive/2017/organizations/'
13     original = "https://summerofcode.withgoogle.com"
14
15     result = list()
16
17     response = requests.get(my_url)
18     html = response.content
19     soup = bsoup(html,"html.parser")
20     organizations = soup.findAll("li",{'class': 'organization-card_container'})
21
22     counter = 0
23     for organization in organizations:
24         page_url=organization.find('a',{'class':'organization-card_link'})
25         organization_name=organization['aria-label']
26
27         about=organization.find('div',{'class':'organization-card_tagline font-black-54'})
28         about.about.text
29         page_link=original.page_url['href']
30         page = requests.get(page_link)
31         if page.status_code != 200:
32             break
33         page_link=original.page_url['href']
34         response1 = requests.get(page_link)
35         html=response1.content

```

The Windows taskbar at the bottom shows the time as 4:19 AM on 3/21/2018, along with various system icons and the search bar.

Output:

I am showing you data upto 15 organisations.

[illegible]

The screenshot shows a web browser with multiple tabs. The active tab is 'localhost:5000/orgs#'. The browser's address bar shows the URL. The page content is a JSON API response, displayed in a dark-themed editor. The JSON data is as follows:

```
{
  "organization_name": "52North Initiative for Geospatial Open Source Software GmbH",
  "description": "52North works on innovative ideas and technologies in geoinformatics.",
  "link": "http://52north.org/",
  "technologies": [
    "web services",
    "ogc standards",
    "java",
    "javascript",
    "web"
  ],
  "topics": [
    "geoinformatics",
    "sensor web",
    "web-based geoprocessing",
    "spatial data infrastructures",
    "spatial information"
  ]
},
{
  "organization_name": "AboutCode",
  "description": "Open Source for Open Source: software license, origin and packages discovery",
  "link": "http://aboutcode.org"
}
```

The browser's taskbar at the bottom shows the Windows Start button, a search bar, and several application icons including File Explorer, Edge, and various development tools. The system clock in the bottom right corner indicates the time is 4:23 AM on 3/21/2018.

References:-

<https://summerofcode.withgoogle.com/archive/2017/organizations/>

<https://www.crummy.com/software/BeautifulSoup/bs4/doc/>

<http://flask.pocoo.org/docs/0.12/>