Global Academy Of Technology

Department of Computer Science and Engineering

Report

On

PYTHON PROJECT

VI Semester

Academic Year: 2018-2019

Title: WEB SCRAPING USING BEAUTIFUL SOUP

USN	Name	Signature
1GA14CS010	Akash Kumar S	
1GA15CS053	G Janany	
1GA16CS191	Vishal Kumar	

Guide
[Mr.Shyam Sundar]

Objective of the Project

To build a system that is capable of extracting large amounts of data from websites whereby the data is extracted and saved to a local file or displayed. It is either custom built for a specific website or is one which can be configured to work with any website. With the click of a button we can easily save the data available in the website to a file in our computer.

System Requirement Specification

Software Requirements Specification

- Language used : Python Programming Language
- IDE/Compiler used : PyCharm
- OS used : Windows 10

Hardware Requirements Specification

- Processor : i7 8th generation
- Hard Disk : 1 TB
- Monitor : HD LED Antiglare
- Keyboard : Island Style

Source Code

```
# make sure to have python ver 3.5 or higher
# 1> install requests using - pip install requests
# 2> install beautifulsoup using - pip install beautifulsoup4
# 3> install lxml using - pip install lxml
     (enter the commands on cmd promt , not on python shell)


import requests   #imports requests module
import bs4        #imports beautifulsoup module


res = requests.get('https://en.wikipedia.org/wiki/Python_(programming_language)')
res.text                                #obtains the entire HTML and/or CSS code of the
                                         website

soup = bs4.BeautifulSoup(res.text, 'lxml')                #lxml is a data structure
result = soup.select('.mw-body-content h2')               #here you can give any HTML
                                                           tag which you want to scrape


for i in soup.select('https://en.wikipedia.org/wiki/Python_(programming_language)'):
print(i.text)


result                          #displays the required data in html code
result[0]                       #displays first element in the array(in this case there is only
                                 one element)
result[0].getText()             #displays the required data in string format
```

Snapshots

1.Snapshot of Source Code

```
Python 3.7.3 Shell
File  Edit  Shell  Debug  Options  Window  Help
Python 3.7.3 (v3.7.3:ef4ec6ed12, Mar 25 2019, 22:22:05) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> import requests
>>> import bs4
>>> res=requests.get('https://en.wikipedia.org/wiki/Python_(programming_language)')
>>> res
<Response [200]>
>>> res.text
Squeezed text (2323 lines).
>>>
============================== RESTART: Shell ==============================
>>> import requests
>>> import bs4
>>> res = requests.get('https://en.wikipedia.org/wiki/Python_(programming_language)')
>>> res.text
Squeezed text (2323 lines).
>>> soup=bs4.BeautifulSoup(res.text , 'lxml')
>>> type(soup)
<class 'bs4.BeautifulSoup'>
>>> result=soup.select('.toctext')
```

2. Snapshot of Result

```
Python 3.7.3 Shell
File  Edit  Shell  Debug  Options  Window  Help
>>> result
[<span class="toctext">History</span>, <span class="toctext">Features and philosophy</span>, <span class="toctext">Syntax and semantics</span>, <span class="toctext">I
ndentation</span>, <span class="toctext">Statements and control flow</span>, <span class="toctext">Expressions</span>, <span class="toctext">Methods</span>, <span clas
s="toctext">Typing</span>, <span class="toctext">Mathematics</span>, <span class="toctext">Libraries</span>, <span class="toctext">Development environments</span>, <sp
an class="toctext">Implementations</span>, <span class="toctext">Reference implementation</span>, <span class="toctext">Other implementations</span>, <span class="toct
ext">Unsupported implementations</span>, <span class="toctext">Cross-compilers to other languages</span>, <span class="toctext">Performance</span>, <span class="toctex
t">Development</span>, <span class="toctext">Naming</span>, <span class="toctext">API documentation generators</span>, <span class="toctext">Uses</span>, <span class="
toctext">Languages influenced by Python</span>, <span class="toctext">See also</span>, <span class="toctext">References</span>, <span class="toctext">Sources</span>, <
span class="toctext">Further reading</span>, <span class="toctext">External links</span>]
>>> for i in soup.select('.toctext'):
        print(i.text)


History
Features and philosophy
Syntax and semantics
Indentation
Statements and control flow
Expressions
Methods
Typing
Mathematics
Libraries
Development environments
Implementations
Reference implementation
Other implementations
Unsupported implementations
Cross-compilers to other languages
Performance
Development
Naming
API documentation generators
Uses
Languages influenced by Python
See also
References
Sources
Further reading
External links
>>>
```

3. Snapshot of Webpage