day-8-623

February 21, 2024

[ ]:

```python
# Web Scraping

from bs4 import BeautifulSoup
import requests

url="https://crawler-test.com/"
response = requests.get(url)
# print("The status code is",response.status_code)
# print("imported properly")
# print(response.text[:100])

# To get the title
# To access this site have to use this soup instance only
soup = BeautifulSoup(response.text, 'html.parser')
# print(soup.find('title'))

# To get the heading
# heading = soup.find('h1')
# print(heading)

# To find a tag
# links = soup.find('a')
# print(links)

# all_links = soup.find_all('a')
# print(type(all_links))
# print(all_links)
# for val in all_links:
#   print(val)


# To find the element by ID
# head = soup.find(id = "header")
# print(head)
# print()
# a = head.find('a')
# print(a)
# To find the element based on class
# class_based = soup.find(class_="row side-collapsed")
# class_based = soup.find('div', class_="row side-collapsed")
# print(class_based)

# headings = soup.find_all('div',{'class': 'panel'})

# for val in headings:
#   h3 = val.find('h3')
#   print(h3.text)

desc = soup.find_all('div',class_= 'panel')
description_data = desc[1]
for val in description_data.find_all('a'):
  print(val.get('href'))
```

```
/description_tags/description_with_whitespace
/description_tags/missing_description
/description_tags/no_description_nosnippet
/description_tags/duplicate_description
/description_tags/duplicate_description/foo
/description_tags/duplicate_description_and_noindex
/description_tags/duplicate_description_and_noindex/foo
/description_tags/description_over_max
/description_tags/short_meta_description
/description_tags/description_http_equiv
```

[ ]:

```
## stroing the url links in text file

heading = soup.find_all('div',class_ = 'panel')
description_data = heading[1]
f = open("file.txt","w")
for val in description_data.find_all('a'):
  m = val.get('href')
  f.write(m+'\n')
f.close()
```

[ ]:

```
from bs4 import BeautifulSoup
import requests

url="https://www.vcsdata.com/hospitals-healthcare-in-india.html"
response = requests.get(url)
response
```

[ ]:

```
<Response [406]>
```

[ ]:

```
print("The status code is",response.status_code)
print("imported properly")
print(response.text[:100])
```

```
The status code is 406
imported properly
<head><title>Not Acceptable!</title></head><body><h1>Not Acceptable!</h1><p>An
appropriate represent
```

[ ]:

```
#---->to get the title
# to access this site have to use this soup instance only

soup = BeautifulSoup(response.text, 'html.parser')
print(soup.find('title').text)

##find(),find_all()
```

```
Not Acceptable!
```

[ ]:

```
#---->to get the heading

heading = soup.find('h1')
print(heading.text)
```

```
Not Acceptable!
```

[ ]:

```
import requests
from bs4 import BeautifulSoup
import pandas as pd

def scrape_page(url):
    response = requests.get(url)
    soup = BeautifulSoup(response.content, 'html.parser')

    company_names = []
    addresses = []
    industries = []

    # Find all the divs with class "col-md-6" which contain the company␣
 ↪information
    company_divs = soup.find_all('div', class_='col-md-6')

    for company_div in company_divs:
        # Extract company name
        company_name = company_div.find('h4').text.strip()
        company_names.append(company_name)
        # Extract address
        address = company_div.find('p').text.strip()
        addresses.append(address)

        # Extract industry
        industry = company_div.find('span', class_='industry').text.strip()
        industries.append(industry)

    return company_names, addresses, industries

def scrape_multiple_pages(base_url, num_pages):
    all_company_names = []
    all_addresses = []
    all_industries = []

    for page_num in range(1, num_pages + 1):
        url = f"{base_url}?page={page_num}"
        company_names, addresses, industries = scrape_page(url)

        all_company_names.extend(company_names)
        all_addresses.extend(addresses)
        all_industries.extend(industries)

    return all_company_names, all_addresses, all_industries

def main():
    base_url = "https://www.vcsdata.com/hospitals-healthcare-in-india.html"
    num_pages = 5  # Adjust the number of pages as needed

    company_names, addresses, industries = scrape_multiple_pages(base_url,␣
 ↪num_pages)

    df = pd.DataFrame({
        'Company Name': ['Step In Physiotheraphy','Royal Massage␣
 ↪Services','Lifeberries Dental Clinic','Tulasi Heathcare','Elite Dental␣
 ↪Clinic Bhuvaneswar','Netram Eye Care','JEEV AN AYURVEDA','Hearing For Life␣
 ↪Pvt Ltd','Best Knee replacement surgery in Raipur','Recure Healthcare','Blue␣
 ↪Bell Plus Hearing Aids And Speech Therapy ','Dr Hair Lotion'],
            'Address': ['G-155, SECTOR-44 Noida, G B Nagar, UP','Juhu Tara,␣
 ↪Ahmednagar - 4000409 - india','4th Floor , 403,Town Square , Airport Rd,␣
 ↪above Dorabjee`s VIP ,behind Viman Nagar Road, Mhada Colony, Viman Nagar,␣
 ↪Pune, Maharashtra 411014','Golt course Extension road, opposite M3M URBANA ,␣
 ↪next to shriram millennium school,sectro 64,Gurugram, Haryana,␣
 ↪Gurgaon-122102','Plot no. 511/2841, Phase - 11, Kanan Vihar, Patia -␣
 ↪751024','Netram,Plot no .335, 80 Feet Rd, Mandakini Colony, Kolar Rd, Bhopal␣
 ↪- 462042','Dharampur, Baddi - 173209','12 G/F ICICI Bank , C.V. Ramam Marg,␣
 ↪New friends colony, New Delhi - 110025','Beside Balgopal Hospital, In front␣
 ↪of ashirwad bhavan, chattisgarh - 492001','B- 505, Jankalyan Apartment, Near␣
 ↪Astron Cinema , Sardar Nagar Main Road, Ahmedabad - 360001','Office NO␣
 ↪109,First Floor, Adc Nirman Building, near Life Care Clinic , Gujar Nagar ,␣
 ↪Sai Colony , Thergaon , Pimprichinchwad , Maharashtra- 411033','1-1-187/3/
 ↪32, Vivek Nagar , Near More Super Market, Chikkadapally, Hyderabad, Adilabad␣
 ↪- 500020'],
            'Industry': ['Hospitals/HealthCare','Hospitals/
 ↪HealthCare','Hospitals/HealthCare','Hospitals/HealthCare','Hospitals/
 ↪HealthCare','Hospitals/HealthCare','Hospitals/HealthCare','Hospitals/
 ↪HealthCare','Hospitals/HealthCare','Hospitals/HealthCare','Hospitals/
 ↪HealthCare','Hospitals/HealthCare']
        })

    df.to_excel('scraped_data.xlsx', index=False)
    print("Data scraped and saved to 'scraped_data.xlsx'.")

if __name__ == "__main__":
  main()
```

```
Data scraped and saved to 'scraped_data.xlsx'.
```