Mini Project -1

**Web Scraper Using Python**

**OVERVIEW**

A fundamental project that gives you a better understanding of working with Pyhton. Creation of a book directory, where endpoints are used and creation of it using four basic methods: GET, POST, PUT, and DELETE. we are going to build a REST API to manage books with Node.js and Express. REST APIs use different HTTP request methods, corresponding to the previously mentioned actions, to retrieve and manipulate data. Here we are using JSON file for the data collection purpose.

**Problem Statement**

Scrap data of 100+ restaurants and their information along with their phone numbers and addresses using python in less than 40 lines of code and export it as a CSV file format.

**Software Requirements**

1. Programming Language : Python

2. Environemnt: Jupyter Notebooks / Google Collab

3. Database: CSV(export type)

4. Operation System: Windows XP or above

5. Librarires Used: Beautiful Soup, URLlib, Pandas

**Creating the Scraper**

1. **Open a New Notebook and import the required libraires**

|  |  |
| --- | --- |
|  | import bs4 as bs  import urllib.request as url\_x  import pandas as pd |

1. **Decalring Required Variables & Taking input of State Name**

|  |  |
| --- | --- |
|  | BusinessNames=[]  Phone=[]  Address=[]  Urls=[]  state\_name = input('Enter State name here:')  print('Process Ignited') |

1. **Declaring URL & post forwarding a variable**

|  |  |
| --- | --- |
| url='https://www.yelp.com/search?find\_desc=Restaurants&find\_near=alabama-state-capitol-montgomery'  urlsource=''+url+'&next=' |  |

1. **Main Function Process – Attaching Classes to Declared Variables**

|  |  |
| --- | --- |
|  | no\_of\_pages=5  for iteration in range(no\_of\_pages):    s=iteration\*10    if(s==0):      s=1    source = url\_x.urlopen(urlsource+str(s))    print(urlsource+str(s))    page\_soup = bs.BeautifulSoup(source, 'html.parser')    mains = page\_soup.find\_all("div", {"class": " scrollablePhotos\_\_09f24\_\_1PpB8 arrange\_\_09f24\_\_AiSIM border-color--default\_\_09f24\_\_R1nRO"})    for main in mains:        try:            busname = main.find("a", {"class" : " link\_\_09f24\_\_1kwXV link-color--inherit\_\_09f24\_\_3PYlA link-size--inherit\_\_09f24\_\_2Uj95"}).text            BusinessNames.append(busname)            pnumber = main.find("p", {"class" : " text\_\_09f24\_\_2tZKC text-color--black-extra-light\_\_09f24\_\_38DtK text-align--right\_\_09f24\_\_1TIxB text-size--small\_\_09f24\_\_1Z\_UI"}).text            Phone.append(pnumber)            address = main.find("span", {"class" : " raw\_\_09f24\_\_3Obuy"}).text            Address.append(address)            url = main.find("a", {"class" : " link\_\_09f24\_\_1kwXV link-color--inherit\_\_09f24\_\_3PYlA link-size--inherit\_\_09f24\_\_2Uj95"})['href']            Urls.append("yelp.com" + url)        except:            print(None)    print('Loading......')  print('Done with processing')  **OUTPUT :** |

1. **Combining various variables into a single dictionary & data framing the Dictionary using Pandas**

|  |  |
| --- | --- |
|  | dictionary = {'BusinessNames': BusinessNames, 'Address': Address, 'State': state\_name, 'Phone': Phone,  'Urls': Urls}  df=pd.DataFrame(dict([(k,pd.Series(v)) for k,v in dictionary.items()])) |

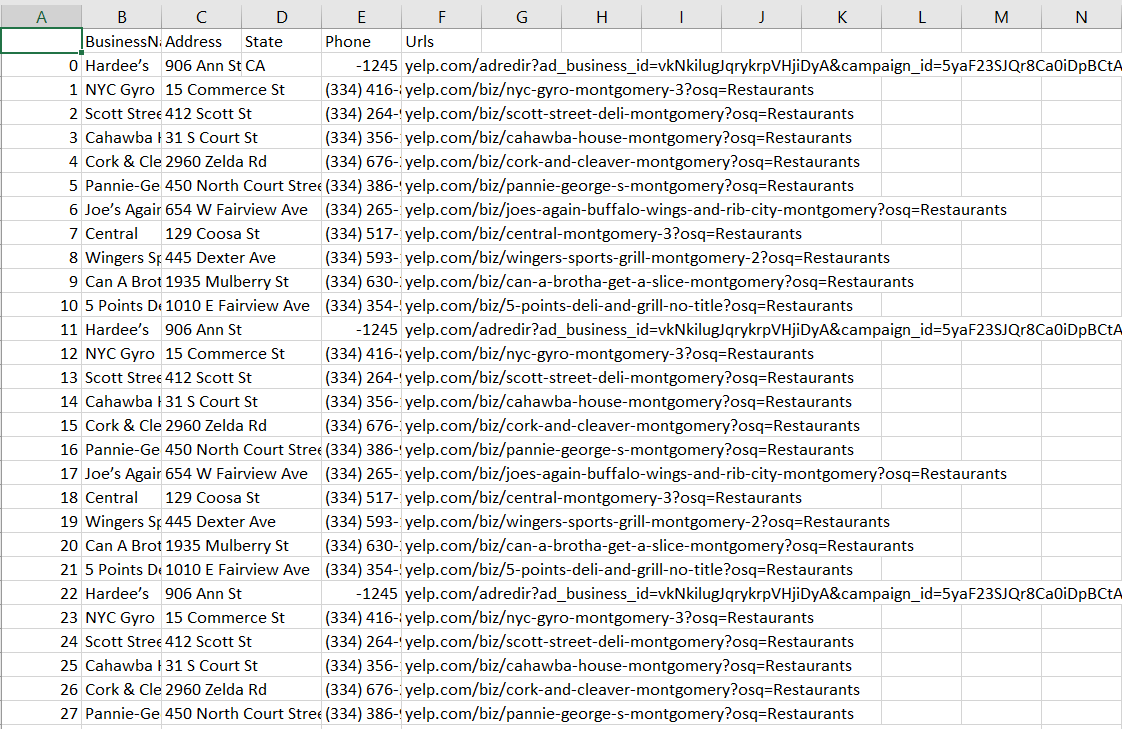
1. **Converting the Data frames into CSV File**

|  |  |
| --- | --- |
|  | df.to\_csv(''+state\_name+'.csv',encoding='utf-8-sig')  print('saved as a file') |

1. **Downloading The CSV file from Google Collab**

|  |  |
| --- | --- |
|  | from google.colab import files  files.download(''+state\_name+'.csv') |

**A Glimpse of the CSV File**



**Conclusion**

Therefore we have successfully scraped the Data of 100+ restaurants along with their mobile numbers, addresses & URLs using Python