**Web Scraping**

The company data is extracted from “https://www.crunchbase.com/organization/ “website by inputting the required company name after the ‘/’ of the url of the given website;

( i.e, google https://www.crunchbase.com/organization/***google***)

Details’:

1. Many lists are used to store data used further combined to form a dataframe.
2. Multiple Functions are created to perform appropriate data extract operation.
3. The code can be have more than one company name as input, the names are inputting by keeping a space after one and another.

**‘No. of companies: apple google microsoft’**

1. **Selenium** and **BeautifulSoup** libraries are used for web page traversal and data extraction.
2. **Regex** library is used to extract data on the basis of required feature.

For example:-

def NA(text):

p1="Acquisitions"

if re.search(p1, text):

a1='YES'

else:

a1='NO'

return(a1)

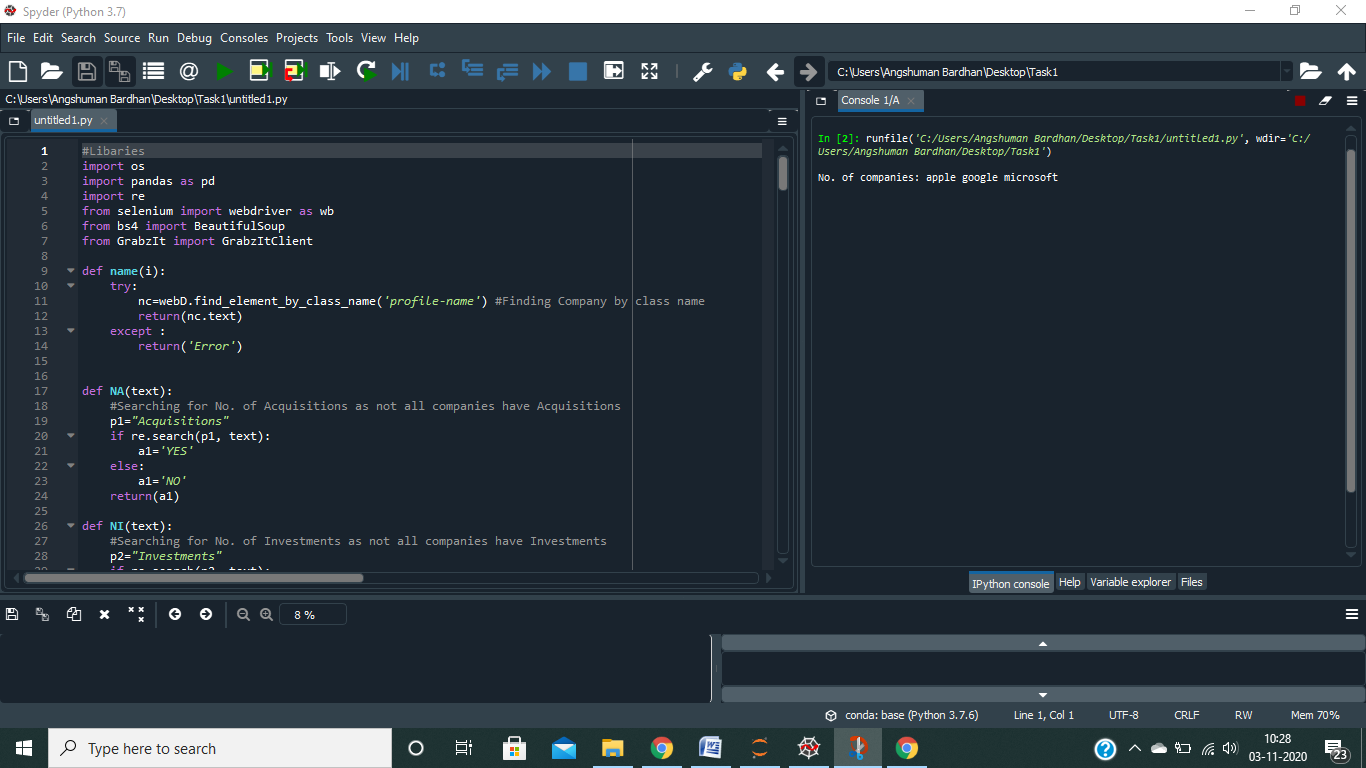
1. Functions :-

* ***name ()***-To extract title name of the web page which is the company name.
* ***NA() and NI()***-It is used to search for the word ‘Acquisitions’ and ‘Investments’ respectively present in the company web page of the website as not all company data of the website has it.
* ***Founder ()***-To obtain the names of founder of the company.
* ***web ()***-To extract website link of company.
* ***FLT ()***-To extract social media link of company.
* ***reduce () and nan()*** -The data obtain from web() and FLT() are in a list which contain ‘None’ value and duplicate value so to remove ‘None’ elements nan() is operated and reduce() removes the duplicate values.
* ***arr() and arr2()*** –These two functions are used to rearrange the list format in a proper way which further used in the operation of dataframe creation.

1. Chrome driver is used for the accessing the web page data.
2. With the help of **BeautifulSoup** the html page is parsed.
3. Path logo and no. of employees are collected with necessary lists.
4. After the lists are formed, they are converted into a dataframe.
5. The dataframe is further used to create a HTML table and csv file.
6. Screenshots of the company links are taken and stored in a folder created by the code with respect to company name.
7. **Grabzit** library is used for screenshots. When this library is executed the whole code run time gets effected and some time may show error due to excessive time period.

**Screenshots:**

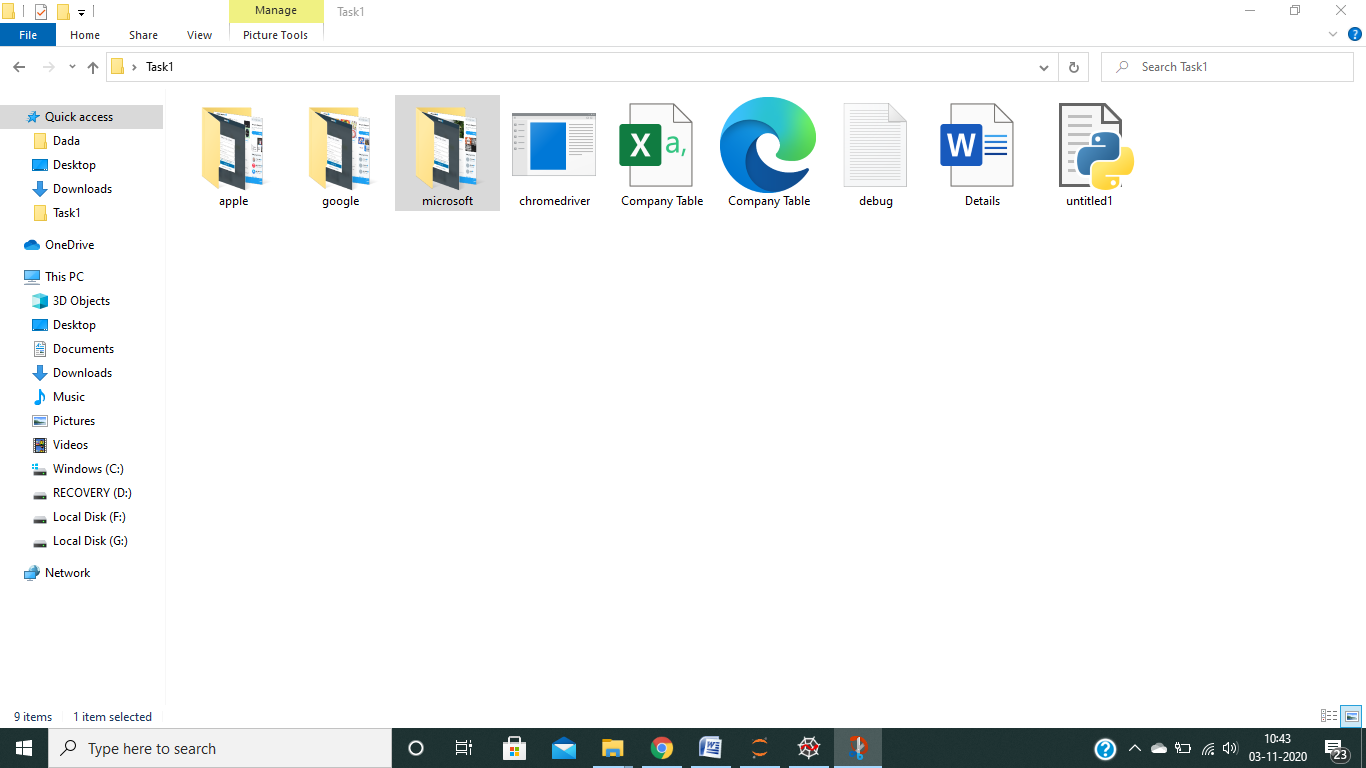
1. **Giving apple, google and microsoft as company name inputs**



1. **Web- page opening of each company**

|  |  |  |
| --- | --- | --- |
| C:\Users\Angshuman Bardhan\Desktop\DATA PIC\apple.png | C:\Users\Angshuman Bardhan\Desktop\DATA PIC\google.png | C:\Users\Angshuman Bardhan\Desktop\DATA PIC\microsoft.png |

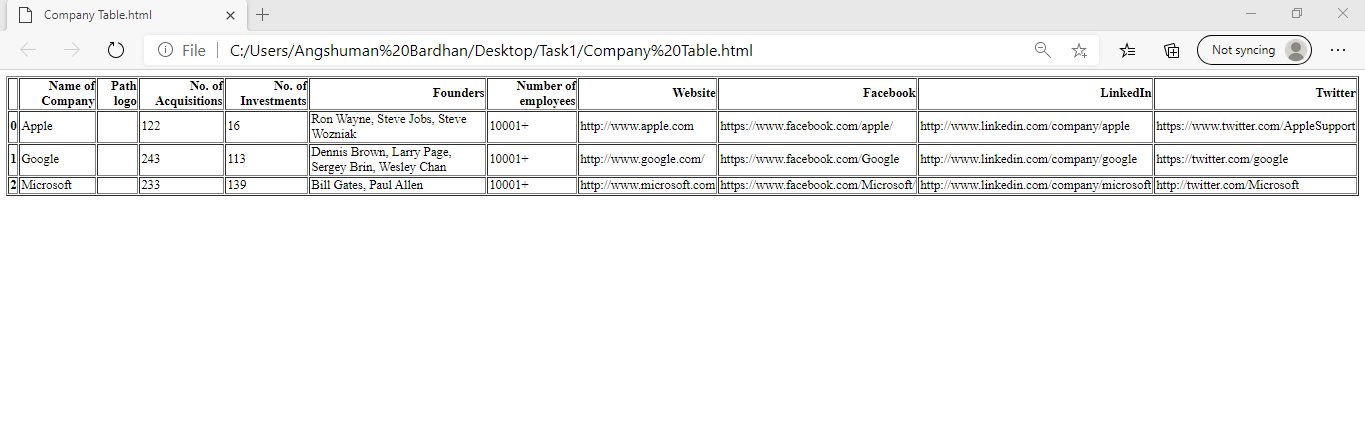
1. **Folders of company screenshots are created and screenshots are stored. HTML table and csv file is also created**



1. **Link based screenshots**

|  |  |  |
| --- | --- | --- |
| **Apple**  C:\Users\Angshuman Bardhan\Desktop\DATA PIC\apple links.png | **Google**  C:\Users\Angshuman Bardhan\Desktop\DATA PIC\google links.png | **Microsoft**  C:\Users\Angshuman Bardhan\Desktop\DATA PIC\microsoft links.png |

1. **HTML TABLE**



1. **CSV FILE**

