Project Report

on

Web Scraping

(Python Based Project)

Bachelor Of Technology

Computer Science and Engineering

By

Swetanshu Pandey (20262)

COMPUTER SCIENCE AND ENGINEERING DEPARTMENT

KAMLA NEHRU INSTITUTE OF TECHNOLOGY, Sultanpur, INDIA

March 2022

ACKNOWLEDGEMENT

This is to acknowledge all those without whom this project would not have been a reality. Firstly, We would like to thank the mentors and colleagues who gave me immense support and made us understand how to make this project. Without their guidance, the project would not have been complete. We are glad that we were able to complete this project and understand many things. The preparation of this project was an immense learning experience and we inculcated many personal qualities during this process like responsibility, punctuality, confidence, and others.

INDEX

INTRODUCTION

Objective:

Web scraping aims to transforms specific contents

from a web site into a structure form:

A database ,a spreadsheet,an XML representation,etc.

E-commerce
Data Science
Job Boards
Marketing & Sales
Data Journalism
Web Scraping Applications
Finance

Problem Statement

To build a python project for the web scraping model.

MOTIVATION

A large amount of data is available on the web in a loosely structured formin HTML pages. While this data is largely meant for human consumption, some websites make it available in a structured format via “Web Services” using mechanisms such as REST and SOAP. These web services allow for programmatic interaction with the data. However, a significant number of websites do not make such web services available, but the data in them are interesting nevertheless. In such cases, Web Scrapers are written to extract information in them and to load the data into more structured stores(ex. Databases) so something useful can be learned from it. The variations in the structure of HTML pages on the web disallow a generic one-size-fits-all algorithm from being used to extract the data; data must be extracted on a case-by-case basis. WSL allows the retrieval of the contents of an HTML page, processing of the retrieved HTML for data extraction and persisting of extracted data to a flat-file (which can then be used to load the data into a more structured store such as an RDBMS).

FEATURES

1. Our Easy Web Extract software contains a lot of advanced features to enable users to scrape content from simple to complicate websites but doesn't require any effort to setup a web scraping project. In this page, we will show you only must-known features which makes our **web scraping tool** so easy-to-use as its names.
2. Scraping data of Multiple Threads
3. Scrap data from all sorts of data loading
4. Auto Execute project any time
5. Export data to any format
6. Run project every hours,run project every days,run project at a specific time.

LIBRARY USED

1. Requests (HTTP for Humans) Library for Web Scraping.

2. lxml Library for Web Scraping
3. Beautiful Soup Library for Web Scraping
4. Selenium Library for Web Scraping

PROJECT DESCRIPTION

Our project contains the following files:

URl - A **Uniform Resource Locator** (**URL**), colloquially termed a **web address**,[1] is a reference to a web resource that specifies its location on a computer network and a mechanism for retrieving it. A URL is a specific type of Uniform Resource Identifier (URI), although many people use the two terms interchangeably. URLs occur most commonly to reference web pages but are also used for file transfer , email, database access, and many other applications.

There are four key parts to every web scraping project:

1. Data discovery

The first step of the technical requirement gathering process is defining what data needs extracting and where can it be found.

2. Data extraction

During this step, our goal is to clearly capture what data do we want to extract from the target web pages. Oftentimes, there are vast amounts of data available on a page so the goal here is to focus on the exact data the customer wants.

3. Extraction scale

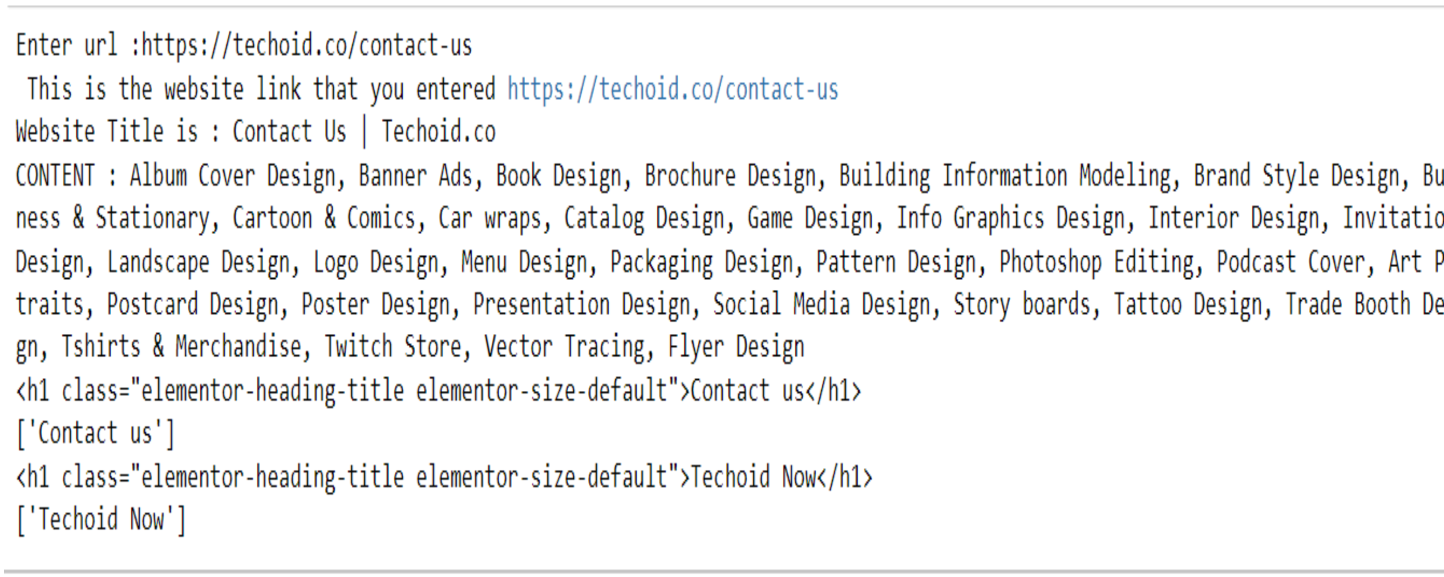
Okay, by this stage you should have a very good idea of the type of data you want to extract and how your crawlers will find and extract it. Next, our goal is to determine the scale of the web scraping project.This is an important step as it allows you to estimate the amount of infrastructural resources (servers, proxies, data storage, etc.) you'll need to execute the project, the amount of data you're likely to receive, and the complexity of the project.

4. Data output

Finally, the last step of the project scoping process is defining how do you want to interact with the web scraping solution along with how do you want the data delivered.If you are building the web scraping infrastructure yourself, you really only have one option: you're managing the data, the web scraping infrastructure, and the underlying source code.However, when customers work with Zyte (or other web scraping providers) we can offer them a number of working relationships to best meet their customer's needs.

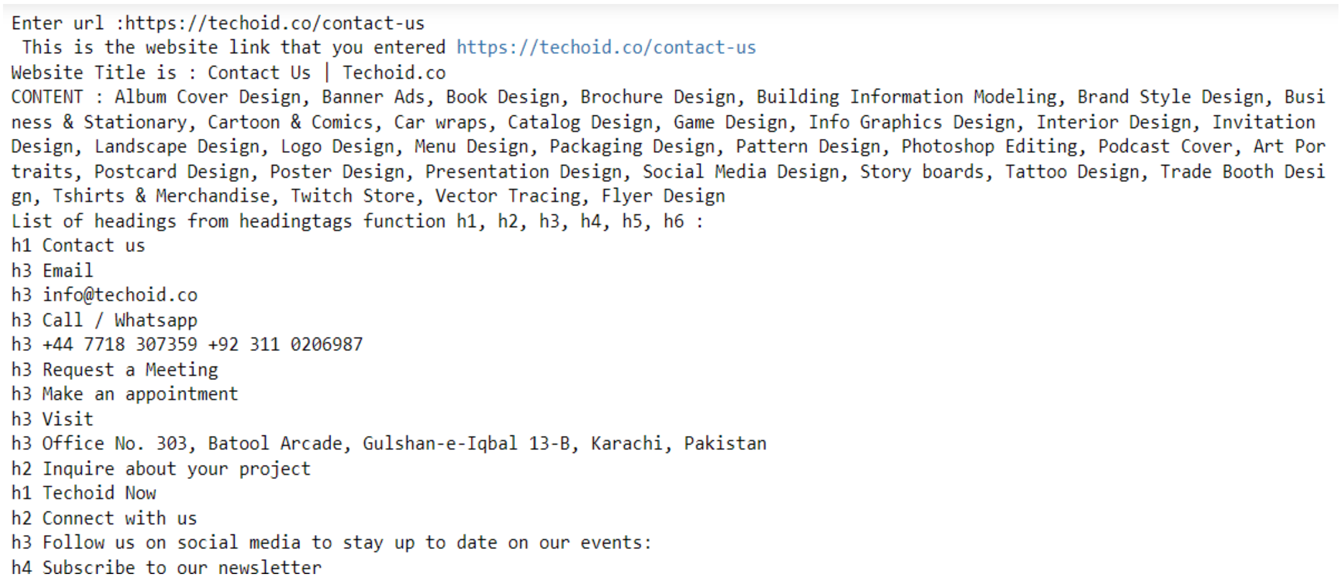
OUTPUTS

To display the contents of a website



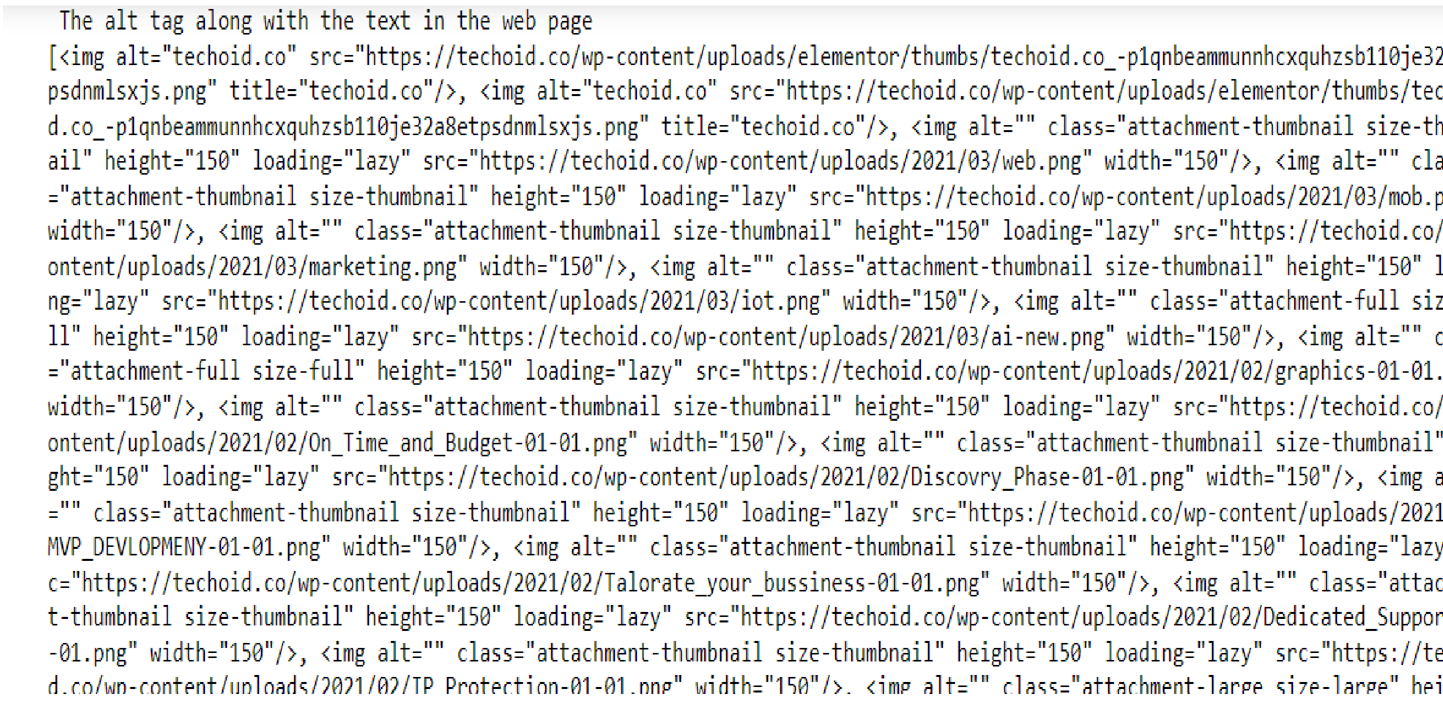
```
Enter url :https://techoid.co/contact-us
 This is the website link that you entered https://techoid.co/contact-us
Website Title is : Contact Us | Techoid.co
CONTENT : Album Cover Design, Banner Ads, Book Design, Brochure Design, Building Information Modeling, Brand Style Design, Bu
ness & Stationary, Cartoon & Comics, Car wraps, Catalog Design, Game Design, Info Graphics Design, Interior Design, Invitatio
Design, Landscape Design, Logo Design, Menu Design, Packaging Design, Pattern Design, Photoshop Editing, Podcast Cover, Art P
traits, Postcard Design, Poster Design, Presentation Design, Social Media Design, Story boards, Tattoo Design, Trade Booth De
gn, Tshirts & Merchandise, Twitch Store, Vector Tracing, Flyer Design
<h1 class="elementor-heading-title elementor-size-default">Contact us</h1>
['Contact us']
<h1 class="elementor-heading-title elementor-size-default">Techoid Now</h1>
['Techoid Now']
```

To display the contact information



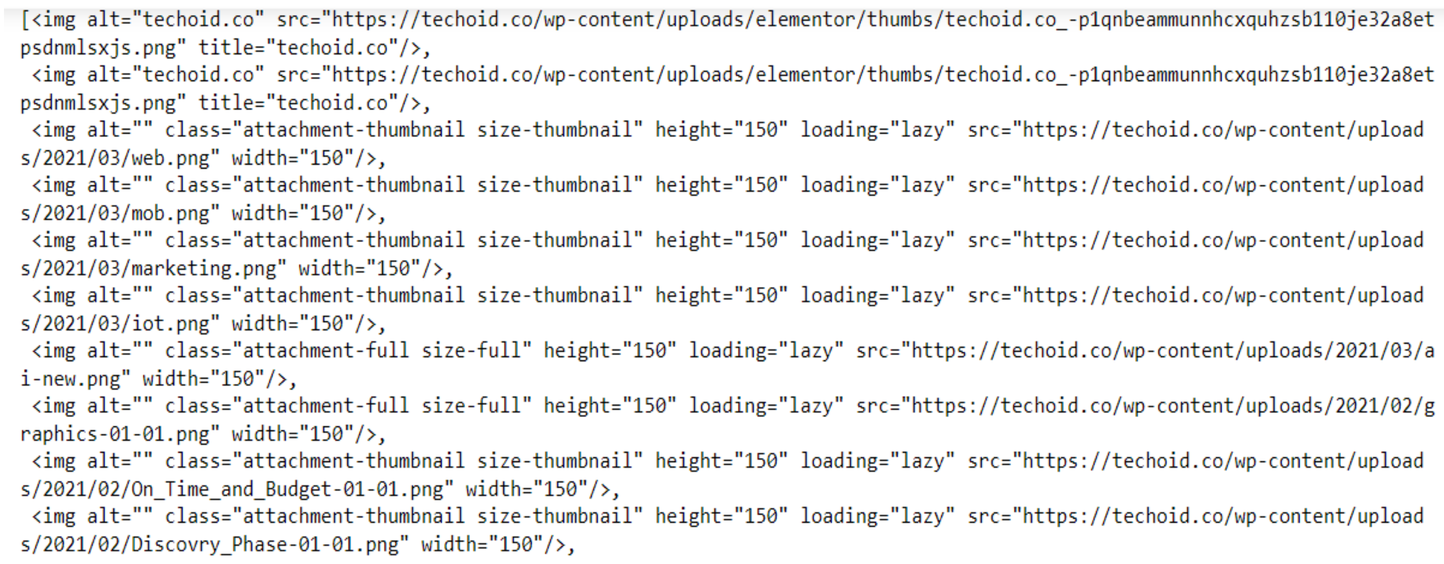
```
Enter url :https://techoid.co/contact-us
 This is the website link that you entered https://techoid.co/contact-us
Website Title is : Contact Us | Techoid.co
CONTENT : Album Cover Design, Banner Ads, Book Design, Brochure Design, Building Information Modeling, Brand Style Design, Busi
ness & Stationary, Cartoon & Comics, Car wraps, Catalog Design, Game Design, Info Graphics Design, Interior Design, Invitation
Design, Landscape Design, Logo Design, Menu Design, Packaging Design, Pattern Design, Photoshop Editing, Podcast Cover, Art Por
traits, Postcard Design, Poster Design, Presentation Design, Social Media Design, Story boards, Tattoo Design, Trade Booth Desi
gn, Tshirts & Merchandise, Twitch Store, Vector Tracing, Flyer Design
List of headings from headingtags function h1, h2, h3, h4, h5, h6 :
h1 Contact us
h3 Email
h3 info@techoid.co
h3 Call / Whatsapp
h3 +44 7718 307359 +92 311 0206987
h3 Request a Meeting
h3 Make an appointment
h3 Visit
h3 Office No. 303, Batool Arcade, Gulshan-e-Iqbal 13-B, Karachi, Pakistan
h2 Inquire about your project
h1 Techoid Now
h2 Connect with us
h3 Follow us on social media to stay up to date on our events:
h4 Subscribe to our newsletter
```

Image Alt Tags



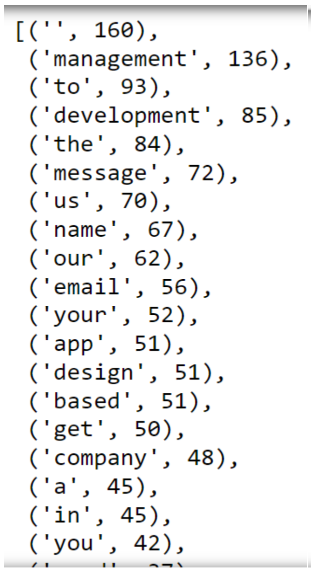
```
    The alt tag along with the text in the web page
   [<img alt="techoid.co" src="https://techoid.co/wp-content/uploads/elementor/thumbs/techoid.co_-p1qnbeammunnhcxquhzsb110je32
   psdnmlsxjs.png" title="techoid.co"/>, <img alt="techoid.co" src="https://techoid.co/wp-content/uploads/elementor/thumbs/tec
   d.co_-p1qnbeammunnhcxquhzsb110je32a8etpsdnmlsxjs.png" title="techoid.co"/>, <img alt="" class="attachment-thumbnail size-th
   ail" height="150" loading="lazy" src="https://techoid.co/wp-content/uploads/2021/03/web.png" width="150"/>, <img alt="" cla
   ="attachment-thumbnail size-thumbnail" height="150" loading="lazy" src="https://techoid.co/wp-content/uploads/2021/03/mob.p
   width="150"/>, <img alt="" class="attachment-thumbnail size-thumbnail" height="150" loading="lazy" src="https://techoid.co/
   ontent/uploads/2021/03/marketing.png" width="150"/>, <img alt="" class="attachment-thumbnail size-thumbnail" height="150" l
   ng="lazy" src="https://techoid.co/wp-content/uploads/2021/03/iot.png" width="150"/>, <img alt="" class="attachment-full siz
   ll" height="150" loading="lazy" src="https://techoid.co/wp-content/uploads/2021/03/ai-new.png" width="150"/>, <img alt="" c
   ="attachment-full size-full" height="150" loading="lazy" src="https://techoid.co/wp-content/uploads/2021/02/graphics-01-01.
   width="150"/>, <img alt="" class="attachment-thumbnail size-thumbnail" height="150" loading="lazy" src="https://techoid.co/
   ontent/uploads/2021/02/On_Time_and_Budget-01-01.png" width="150"/>, <img alt="" class="attachment-thumbnail size-thumbnail"
   ght="150" loading="lazy" src="https://techoid.co/wp-content/uploads/2021/02/Discovry_Phase-01-01.png" width="150"/>, <img a
   ="" class="attachment-thumbnail size-thumbnail" height="150" loading="lazy" src="https://techoid.co/wp-content/uploads/2021
   MVP_DEVLOPMENY-01-01.png" width="150"/>, <img alt="" class="attachment-thumbnail size-thumbnail" height="150" loading="lazy
   c="https://techoid.co/wp-content/uploads/2021/02/Talorate_your_bussiness-01-01.png" width="150"/>, <img alt="" class="attac
   t-thumbnail size-thumbnail" height="150" loading="lazy" src="https://techoid.co/wp-content/uploads/2021/02/Dedicated_Suppor
   -01.png" width="150"/>, <img alt="" class="attachment-thumbnail size-thumbnail" height="150" loading="lazy" src="https://te
   d.co/wp-content/uploads/2021/02/IP_Protection-01-01.png" width="150"/>, <img alt="" class="attachment-large size-large" hei
```

Review image tags in separate line

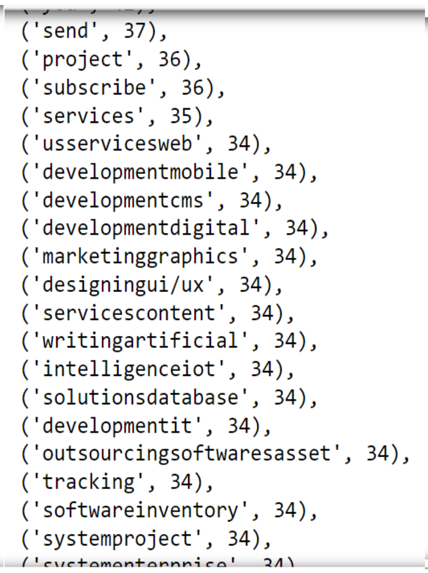


```
[<img alt="techoid.co" src="https://techoid.co/wp-content/uploads/elementor/thumbs/techoid.co_-p1qnbeammunnhcxquhzsb110je32a8et
psdnmlsxjs.png" title="techoid.co"/>,
 <img alt="techoid.co" src="https://techoid.co/wp-content/uploads/elementor/thumbs/techoid.co_-p1qnbeammunnhcxquhzsb110je32a8et
psdnmlsxjs.png" title="techoid.co"/>,
 <img alt="" class="attachment-thumbnail size-thumbnail" height="150" loading="lazy" src="https://techoid.co/wp-content/upload
s/2021/03/web.png" width="150"/>,
 <img alt="" class="attachment-thumbnail size-thumbnail" height="150" loading="lazy" src="https://techoid.co/wp-content/upload
s/2021/03/mob.png" width="150"/>,
 <img alt="" class="attachment-thumbnail size-thumbnail" height="150" loading="lazy" src="https://techoid.co/wp-content/upload
s/2021/03/marketing.png" width="150"/>,
 <img alt="" class="attachment-thumbnail size-thumbnail" height="150" loading="lazy" src="https://techoid.co/wp-content/upload
s/2021/03/iot.png" width="150"/>,
 <img alt="" class="attachment-full size-full" height="150" loading="lazy" src="https://techoid.co/wp-content/uploads/2021/03/a
i-new.png" width="150"/>,
 <img alt="" class="attachment-full size-full" height="150" loading="lazy" src="https://techoid.co/wp-content/uploads/2021/02/g
raphics-01-01.png" width="150"/>,
 <img alt="" class="attachment-thumbnail size-thumbnail" height="150" loading="lazy" src="https://techoid.co/wp-content/upload
s/2021/02/On_Time_and_Budget-01-01.png" width="150"/>,
 <img alt="" class="attachment-thumbnail size-thumbnail" height="150" loading="lazy" src="https://techoid.co/wp-content/upload
s/2021/02/Discovry_Phase-01-01.png" width="150"/>,
```

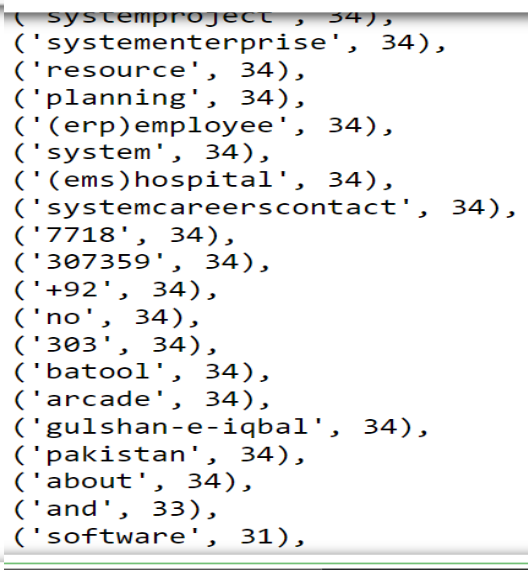
List of common words



```
[('', 160),
 ('management', 136),
 ('to', 93),
 ('development', 85),
 ('the', 84),
 ('message', 72),
 ('us', 70),
 ('name', 67),
 ('our', 62),
 ('email', 56),
 ('your', 52),
 ('app', 51),
 ('design', 51),
 ('based', 51),
 ('get', 50),
 ('company', 48),
 ('a', 45),
 ('in', 45),
 ('you', 42),
```

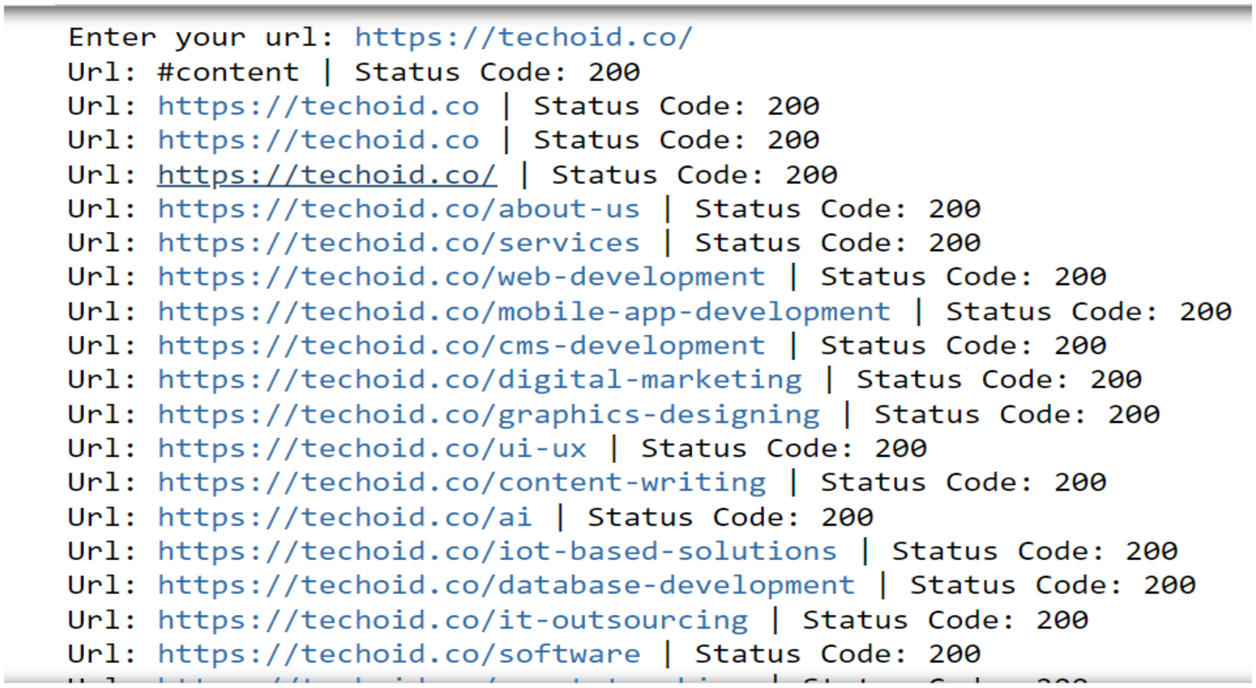


```
('send', 37),
('project', 36),
('subscribe', 36),
('services', 35),
('usserviceswеb', 34),
('developmentmobile', 34),
('developmentcms', 34),
('developmentdigital', 34),
('marketinggraphics', 34),
('designingui/ux', 34),
('servicescontent', 34),
('writingartificial', 34),
('intelligenceiot', 34),
('solutionsdatabase', 34),
('developmentit', 34),
('outsourcingsoftwaresasset', 34),
('tracking', 34),
('softwareinventory', 34),
('systemproject', 34),
```



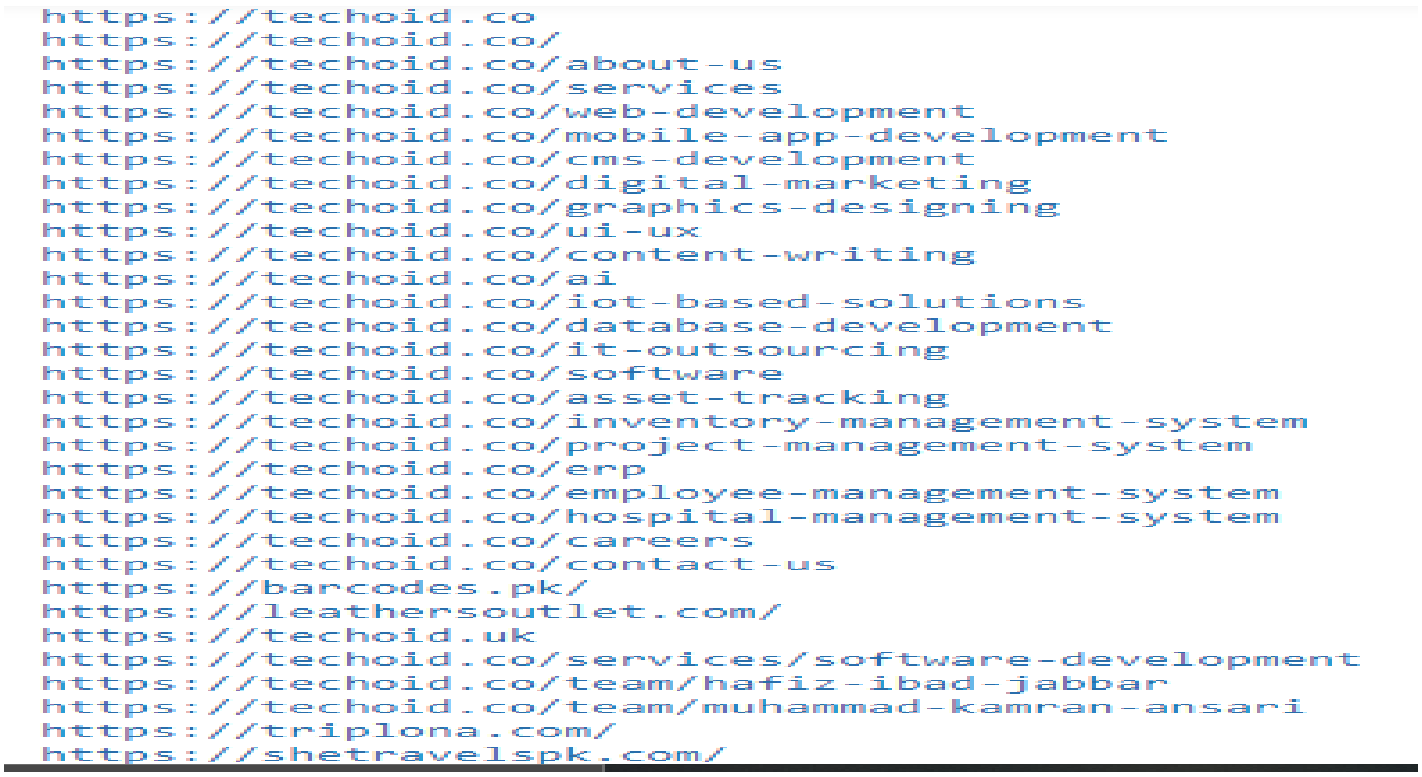
```
('systementerprise', 34),
('resource', 34),
('planning', 34),
('(erp)employee', 34),
('system', 34),
('(ems)hospital', 34),
('systemcareerscontact', 34),
('7718', 34),
('307359', 34),
('+92', 34),
('no', 34),
('303', 34),
('batool', 34),
('arcade', 34),
('gulshan-e-iqbal', 34),
('pakistan', 34),
('about', 34),
('and', 33),
('software', 31),
```

Broken links inside a webpage



```
Enter your url: https://techoid.co/
Url: #content | Status Code: 200
Url: https://techoid.co | Status Code: 200
Url: https://techoid.co | Status Code: 200
Url: https://techoid.co/ | Status Code: 200
Url: https://techoid.co/about-us | Status Code: 200
Url: https://techoid.co/services | Status Code: 200
Url: https://techoid.co/web-development | Status Code: 200
Url: https://techoid.co/mobile-app-development | Status Code: 200
Url: https://techoid.co/cms-development | Status Code: 200
Url: https://techoid.co/digital-marketing | Status Code: 200
Url: https://techoid.co/graphics-designing | Status Code: 200
Url: https://techoid.co/ui-ux | Status Code: 200
Url: https://techoid.co/content-writing | Status Code: 200
Url: https://techoid.co/ai | Status Code: 200
Url: https://techoid.co/iot-based-solutions | Status Code: 200
Url: https://techoid.co/database-development | Status Code: 200
Url: https://techoid.co/it-outsourcing | Status Code: 200
Url: https://techoid.co/software | Status Code: 200
```

Extraction of all URLs from a website without duplication



```
https://techoid.co
https://techoid.co/
https://techoid.co/about-us
https://techoid.co/services
https://techoid.co/web-development
https://techoid.co/mobile-app-development
https://techoid.co/cms-development
https://techoid.co/digital-marketing
https://techoid.co/graphics-designing
https://techoid.co/ui-ux
https://techoid.co/content-writing
https://techoid.co/ai
https://techoid.co/iot-based-solutions
https://techoid.co/database-development
https://techoid.co/it-outsourcing
https://techoid.co/software
https://techoid.co/asset-tracking
https://techoid.co/inventory-management-system
https://techoid.co/project-management-system
https://techoid.co/erp
https://techoid.co/employee-management-system
https://techoid.co/hospital-management-system
https://techoid.co/careers
https://techoid.co/contact-us
https://barcodes.pk/
https://leathersoutlet.com/
https://techoid.uk
https://techoid.co/services/software-development
https://techoid.co/team/hafiz-ibad-jabbar
https://techoid.co/team/muhammad-kamran-ansari
https://triplona.com/
https://shetravelspk.com/
```

SOURCE CODE

```
#Importing Libraries

from bs4 import BeautifulSoup

import urllib

from urllib import request

import urllib.request as ur

# Getting input for webiste from user

urlinput = input("Enter url :")

print(" This is the website link that you entered", urlinput)



# For extracting specific tags from webpage

def getTags(tag):

  s = ur.urlopen(urlinput)

  soup = BeautifulSoup(s.read())

  return soup.findAll(tag)

# For extracting specific title & meta description from webpage

def titleandmetaTags():

    s = ur.urlopen(urlinput)
    soup = BeautifulSoup(s.read())

    #----- Extracting Title from website ------#

    title = soup.title.string

    print ('Website Title is :', title)

    #-----  Extracting Meta description from website ------#

    meta_description = soup.find_all('meta')

    for tag in meta_description:

        if 'name' in tag.attrs.keys() and tag.attrs['name'].strip().lower() in
['description', 'keywords']:

            #print ('NAME    :',tag.attrs['name'].lower())

            print ('CONTENT :',tag.attrs['content'])
```

For extracting ALT tags (Image Alter tags)

```
import urllib.request as ur

url_input = input("Enter url :")

print("The website link that you entered is:", url_input)

def alt_tag():

  url =  ur.urlopen(url_input)
```

```
 htmlSource = url.read()

 url.close()

 soup = BeautifulSoup(htmlSource)

 print('\n The alt tag along with the text in the web page')

 print(soup.find_all('img',alt= True))
```

For reviewing alt tags in seperate lines

```
soup.find_all('img',alt= True)
```

For counting words inside a web page

```
import requests

from bs4 import BeautifulSoup

from collections import Counter

from string import punctuation

# Getting content from web page

r = requests.get("https://techoid.co/contact-us")

soup = BeautifulSoup(r.content)

# For getting words within paragrphs
```

```
text_paragraph = (".join(s.findAll(text=True))for s in soup.findAll('p'))

count_paragraph = Counter((x.rstrip(punctuation).lower() for y in
text_paragraph for x in y.split()))

# For getting words inside div tags

text_div = (".join(s.findAll(text=True))for s in soup.findAll('div'))

count_div = Counter((x.rstrip(punctuation).lower() for y in text_div
for x in y.split()))

# Adding two counters for getting a list with words count (from most
to less common)

total = count_div + count_paragraph

list_most_common_words = total.most_common()
```

List of common words

```
list_most_common_words
```

6. For getting the source code of the webpage

Here, we will be using 'page_source' method is used retrieve the page source of the webpage the user is currently accessing.

NOTE: (Page source : The source code/page source is the programming behind any webpage)

Extraction of all URLs from a website without duplication

```
import re
import requests

from bs4 import BeautifulSoup

all_links = set()

for i in range(7):

   r = requests.get(("https://techoid.co/?page={}").format(i))

   soup = BeautifulSoup(r.content , "html.parser")

   for link in soup.find_all("a",href=re.compile('/')):

            link = (link.get('href'))

            if link not in all_links:

              print(link)

            all_links.add(link)
```

REFERENCES

1.https://www.javatpoint.com/web-scraping-using-python –
2.https://likegeeks.com/python-web-scraping/ -
3.https://www.w3resource.com/python-exercises/web-

scraping/index.php –
4.https://www.youtube.com/watch?v=uufDGjTuq34 –
5.https://realpython.com/beautiful-soup-web-scraper-python/ -
6.https://www.tutorialspoint.com/requests/requests_web_scraping_using_requests.htm –
7.https://docs.scrapy.org/en/latest/intro/tutorial.html –