# Web-Crawler Documentation

Omkar Shirpure (22B0910)

# Contents

# 1 Introduction

The Web-Crawler is a Python program designed as the culmination of a comprehensive course on web development and data extraction. This end-of-semester project aimed to assess the students' mastery of the skills learned throughout the course. The Web-Crawler tackles the problem of efficiently extracting information from websites, addressing the challenges of navigating through webpages, retrieving links, and collecting data on various file types. By automating these tasks, the Web-Crawler serves as a practical demonstration of the students' abilities to apply their acquired knowledge and skills to real-world scenarios.

# 2 How to Use the Web-Crawler

To use the Web-Crawler program, follow these steps:

1. Open a command-line interface.
2. Run the Python script with the following command: `python web_crawler.py`

   `without find function:`

   ```
   python .\web-crawler.py -t 2 -u https://itc.gymkhana.iitb.ac.in/wncc/soc -o output.html_
   ```

   `with find function:`

   ```
   python .\web-crawler.py -t 1 -u https://quotes.toscrape.com -o q.html -f friend Albert_
   ```

3. Provide the required command-line arguments:
   - `-u` or `--url`: The URL of the website to crawl.
   - `-t` or `--threshold`: The recursion threshold (maximum depth) for crawling.
   - `-o` or `--output`: (Optional) The output file name to save the HTML flow chart. If not provided, the flow chart will be displayed in the console.
   - `-f` or `--find`: (Optional) One or more keywords to search for in the links. Only links containing these keywords will be displayed in the flow chart.
4. Press Enter to execute the program.
5. The program will crawl the website and generate an HTML flow chart displaying the links and files found. If an output file name is provided, the flow chart will be saved to that file; otherwise, it will be displayed in the console.

# 3 Output

The output in the terminal

```
Processing: https://quotes.toscrape.com

Flow chart saved to q.html
```

```
Processing: https://itc.gymkhana.iitb.ac.in/wncc/soc

Processing: https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project285.html

Processing: https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project270.html

Processing: https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project284.html

Processing: https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project265.html

Processing: https://itc.gymkhana.iitb.ac.in/wncc/svg/light-siber-one.svg

Processing: https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project228.html

Processing: https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project226.html

Processing: https://itc.gymkhana.iitb.ac.in/wncc/soc/

Processing: https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project227.html

Processing: https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project301.html

Processing: https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project279.html

Processing: https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project290.html

Processing: https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project224.html

Processing: https://itc.gymkhana.iitb.ac.in/wncc/

Processing: https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project242.html

Processing: https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project247.html

Processing: https://itc.gymkhana.iitb.ac.in/wncc/team

Processing: https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project246.html

Processing: https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project276.html

Processing: https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project283.html
```

Output in the HTML to display the links and browse easily :

# __Web_Crawler__

- Main URL: https://quotes.toscrape.com

You searched for word/s ['friend', 'Albert']

(only the links with the searched keyword will be displayed (total being overall total number))

On Recursion Level 1: https://quotes.toscrape.com
Total HTML found: 49

https://quotes.toscrape.com/tag/friendship/
https://quotes.toscrape.com/author/Albert-Einstein
https://quotes.toscrape.com/tag/friends/
Total CSS found: 2

# __Web_Crawler__

- Main URL: https://itc.gymkhana.iitb.ac.in/wncc/soc

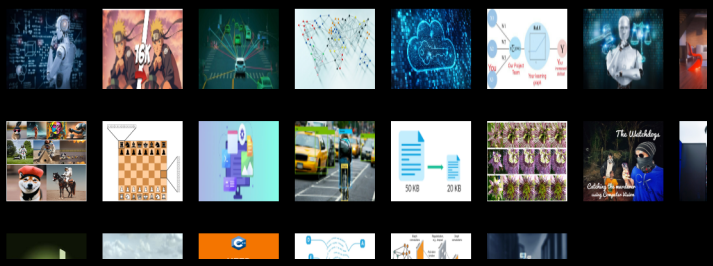
On Recursion Level 1: https://itc.gymkhana.iitb.ac.in/wncc/soc
Total HTML found: 85

https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/item223.html
https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project248.html
https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project221.html
https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project286.html
https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project252.html
https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project260.html
https://www.instagram.com/wncc.iitb/
https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project241.html
https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project245.html
https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project250.html
https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project257.html
https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project299.html
https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project289.html
https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project229.html
https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project243.html
https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project285.html
https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project261.html
https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project230.html
https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project267.html
https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project268.html
https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project302.html
https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project255.html
https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project259.html
https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project288.html
https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project284.html
https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project264.html
https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project296.html
https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project290.html
https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project272.html
https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project283.html

https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project249.html
https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project281.html
https://itc.gymkhana.iitb.ac.in/wncc/soc/projects/2023/project293.html
Total CSS found: 4

https://itc.gymkhana.iitb.ac.in/wncc/assets/css/style.css
https://itc.gymkhana.iitb.ac.in/wncc/assets/plugins/bootstrap/bootstrap.min.css
https://itc.gymkhana.iitb.ac.in/wncc/assets/plugins/themify-icons/themify-icons.css
https://itc.gymkhana.iitb.ac.in/wncc/assets/plugins/slick/slick.css
Total JS found: 5

https://itc.gymkhana.iitb.ac.in/wncc/assets/plugins/jQuery/jquery.min.js
https://itc.gymkhana.iitb.ac.in/wncc/assets/js/script2.js
https://itc.gymkhana.iitb.ac.in/wncc/assets/plugins/shuffle/shuffle.min.js
https://itc.gymkhana.iitb.ac.in/wncc/assets/plugins/bootstrap/bootstrap.min.js
https://itc.gymkhana.iitb.ac.in/wncc/assets/plugins/slick/slick.min.js
Total JPG found: 26



# 4 Explanation of Code

The code can be divided into the following parts:

1. Importing necessary libraries and setting up global variables and data structures.

2. The `parse_arguments()` function parses the command-line arguments provided to the program, such as the website URL, recursion threshold, output file name, and keyword(s) to search for.

3. The `get_links(url)` function retrieves all the links present on a webpage specified by the given URL. It sends an HTTP GET request to the URL using the `requests` library and parses the HTML content using BeautifulSoup. It extracts links from anchor tags, link tags, script tags, and image tags.

4. The `filter_internal_links(links, domain)` function filters out the internal links from the list of all links. It checks if the domain of a link matches the specified domain.

5. The `count_files_by_type(links)` function counts the files by their types. It creates a dictionary with file extensions as keys and lists of corresponding file links as values.

6. The `crawl(url, threshold, depth)` function is the main crawler function. It recursively crawls the website starting from the given URL up to a specified recursion depth. It keeps track of visited links, counts files by their types, and generates a flow map of the crawling process.

7. The `main()` function is the entry point of the program. It parses command-line arguments, sets up global variables, and initiates the crawling process. It also generates an HTML flow chart of the crawling process and optionally saves it to an output file.

## 5 Resources

1. argparse library : https://docs.python.org/3/library/argparse.html
2. beautifulsoup: https://pypi.org/project/beautifulsoup4/
3. urllib.parse : https://docs.python.org/3/library/urllib.parse.html