Web Page Evaluator Documentation

Your Name

March 10, 2024

1 Student Information

1.1 Student: Akarshit

Name: Akarshit
Institute: National Institute of Technology (NIT) Srinagar
Batch: 2020-2024
Roll Number: 2020BITE009
Program: B.Tech in Information Technology
Description:

Akarshit, a student of the National Institute of Technology (NIT) Srinagar, is currently pursuing B.Tech in Information Technology and belongs to the batch of 2020-2024. With the roll number 2020BITE009, Akarshit has been an integral part of the academic community at NIT Srinagar.

Known for active participation in academic and extracurricular activities, Akarshit exemplifies dedication, enthusiasm, and a commitment to excellence. He has contributed to [mention any specific contributions or activities, if applicable].

As a member of the batch graduating in 2024, Akarshit brings a dynamic and positive energy to the campus. He actively participates in [mention any clubs, events, or extracurricular activities].

Conclusion:

Akarshit is not just a student but a valuable member of the NIT Srinagar community, contributing to the institute's legacy of excellence and innovation.

2 Overview

The Web Page Evaluator is a Python class designed for web scraping and extracting information from a specified web page. The class, named `Evaluator`, utilizes the `requests` library for making HTTP requests and the `BeautifulSoup` library for parsing HTML content.

3 Class: Evaluator

3.1 Constructor (`__init__` method)

```
def __init__(self)
```

Description:

- Initializes the `Evaluator` class.
- Sets the `headers` attribute with a user-agent string to mimic a web browser.

3.2 Method: `make_request`

```
def make_request(self)
```

Description:

- Sends an HTTP GET request to a specified URL.
- Uses the headers defined in the constructor.
- Handles potential HTTP errors and request exceptions.

3.3 Method: `get_h1_tags`

```
def get_h1_tags(self)
```

Description:

- Retrieves the text content of the first 'h1' tag from the HTML content.
- Prints the text content without HTML tags.
- Indicates if the 'h1' tag is not found.

3.4 Method: `get_title`

```
def get_title(self)
```

Description:

- Extracts and prints the text content of the 'title' tag from the HTML content.
- Indicates if the 'title' tag is not found.

3.5 Method: `get_div`

```
def get_div(self)
```

Description:

- Focuses on the first 'div' tag in the HTML content.
- Prints the text content of the 'div' tag.
- Indicates if the 'div' tag is not found.

4 Usage Example

```
# Create an instance of the Evaluator class
evaluator = Evaluator()

# Call the get_div() method
evaluator.get_div()
```

Description:

- Instantiates an object of the `Evaluator` class.
- Demonstrates the use of the `get_div` method to extract information from the first 'div' tag of a specified web page.

5 Dependencies

- `requests`: Used for making HTTP requests.
- `BeautifulSoup`: Used for parsing HTML content.

6 Notes

- The user-agent string in the headers is configured to mimic a web browser, preventing potential blocking from websites.
- The class is designed to handle HTTP errors and request exceptions gracefully.

7 Conclusion

The Web Page Evaluator class provides a simple and effective way to extract information from web pages, demonstrating its utility in web scraping tasks. Users can leverage its methods to retrieve specific HTML tag content and incorporate the class into their projects for web data extraction.