Python Web Scraping Nouns from Wiktionary

Installation

Python Version:

The script is developed with Python 3.7 and above.

Required Libraries:

The script uses requests for HTTP requests and BeautifulSoup from the bs4 package for parsing HTML content. You can install them using pip, the Python package installer. Run the following commands in your terminal:

pip install requests
pip install beautifulsoup4

How to Use:

This script scrapes nouns of a given language from different categories on Wiktionary.

Step 1: Define Language and Categories

First, specify the language you want to scrape from. Assign the name of the language as a string to the language variable. Next, specify the categories from which you want to scrape the nouns. You should assign a list of category names as strings to the categories variable. These category names must follow the format used by Wiktionary. For example, if you're scraping English nouns, your category names might look like "Category:English_nouns".

Step 2: Run the Script

Once you have defined the language and categories, you can run the script. The script will:

1. Iterate through each category.
2. For each category, it will retrieve all page URLs associated with the category (taking pagination into account).
3. For each page URL, it will scrape all the nouns listed on the page.
4. All the nouns from all categories will be compiled into a single list.

Step 3: Output

At the end of the script, the list of all nouns is saved to a text file. The name of the text file is the language name followed by "_nouns.txt". Each noun is written on a new line in the file.

Functions

The script defines several functions to modularize the tasks:

- get_page_data(url): Returns a BeautifulSoup object for the HTML content of the given URL.
- get_pagination_links(url): Returns a list of URLs for all pages in the pagination sequence of the given URL.
- get_nouns(url): Returns a list of all the nouns on the given URL.
- main(): Orchestrates the entire scraping process by calling the other functions in the appropriate order and handling data between them.

Execution

To execute the script, you can run it as a standalone Python file from the terminal or as a script in a Python IDE. If you're using the terminal, navigate to the directory containing the script and run:

‘python filename.py’

Replace "filename.py" with the actual name of the Python file.