Scraping and Chunking: Mini Project

Your goal for this project is to scrape an entire public-hosted knowledge base and chunk it effectively into clean smaller pieces so that related content is preserved in a single chunk.

Considerations:

- Please implement this solution in Python. We recommend using `BeautifulSoup`.
- The priorities in descending order are:
    - Cleanliness/organization of the code - **SIMPLER IS BETTER**
    - Correctness
    - Creativity
- This project is not timed and you may use the internet and LLM helpers as you wish.
- After Finish give the github repo link

Step 1

First, scrape all the Help Articles from the **Notion help center**. Make sure you get every page and all the relevant content from that page. Feel free to ignore any guides in Notion Academy.

Step 2

Extract the core text content from each article. Feel free to ignore images, other media, and any components that are not directly related to the core article. Make sure to include all titles, notes, and paragraphs.

Step 3

Now it's time to split the articles into smaller chunks. This is important for any RAG-type system. Make sure to keep headers and paragraphs together and don't break up bulleted lists mid-list. Your chunks should be roughly 750 characters or fewer but could be more if it's necessary to keep related context together.

<aside> 💡 Tip: LLMs are very good at processing unstructured text (or HTML) and extracting what you want

</aside>

Your final output should be an array of these chunks!

Bonus

The text output of web scraping is often very messy. In particular, if there are tables, lists, or other unusual formatting, just converting HTML to text can look very odd. You could consider using an LLM to help with formatting and prettifying the information so that all the information is captured!