COMP 4601A
Fall 2023 - Lab #4

Objectives

The goal for this lab is to create a basic server that supports search queries on your existing crawled data using the elasticlunr.js package.

Demonstrating/Submitting

There will be two ways to receive credit for completed labs, outlined below:
1. Attend an in-person lab or office hours and demonstrate your completed lab before the deadline. You will have to show that the goals of the lab have been completed and answer some questions about the lab and your code (see the lab reflection questions for some examples). Your grade will depend on the level of completion, as well as the quality of your design and answers. Only one partner is required for demonstration, though all partners are encouraged to take part. **If you demonstrate your lab this way, you don't need to submit anything on Brightspace.**
2. Record a video demonstration that is <10 minutes long. Ensure that your discussion in the video makes it clear that you have understood the content that the lab covers and that you demonstrate all the required functionality. Submit a ZIP containing a copy of your code (don't include database files, etc.), your answers to the lab reflection questions, and a copy of your demonstration video (either link to a public URL in your README or include the video file directly) to Brightspace. **If you are working with a partner, only one of you should make a submission. Include the names of all group members in the README file.**

Lab Description

Your server for this lab should use the fruit website data generated by your crawler from the previous lab. Your server is required to index the data and provide the required search functionality. You must include at least the page title and the page body contents within the index. You can include additional data if you desire.

The home page for your server (i.e., URL /), must serve a page in which the user can enter a simple text search query and submit the search request. In response to a search request, your server must perform a search using the indexed data and return an HTML page containing a list of the top 10 search results. Each search result must include at least the URL to the original page, the title of the original page, and the search score generated by your search engine for

this search query. As part of your demonstration, you should include some brief analysis of why you think the returned search results are ranked in a relevant manner (e.g., by looking at several pages' data and comparing that data to the search terms).

Lab Reflection Questions

1. How do you load your data from the database to be indexed?
2. What fields do you perform indexing on?
3. How is the search score computed?
4. How scalable is your implementation? How could you improve it?