Backend

Crawling: Used BeautifulSoup, a python library, to crawl the Java wiki pages. Each individual document consisted of a wiki subtopic. Recorded the header, links , text content for indexing purposes and also the respective html for display purposes.

Stemming: Stemming reduces the words into their respective root forms. Incorporated stemming to increase the chances of matching different forms of a word.

Removing Stop words: Stop words are common words which could be removed without compromising the quality of the text content.

Used the ‘nltk’ library (natural language toolkit) for stemming and removing stop word.

After the wiki pages were thoroughly crawled, the documents were passed to Solr for indexing. Solr is built on Apache Lucene. I chose to go with Solr because it uses the BM25 (Best matching 25) ranking function. It is different from others because it is derived from probabilistic information retrieval. BM25 is a bag-of-words retrieval function that ranks a set of documents based on the query terms appearing in each document, regardless of the inter-relationship between the query terms within a document (e.g., their relative proximity). It is not a single function, but actually a whole family of scoring functions, with slightly different components and parameters.[wikipedia]

User-Interface

The user interface is designed in such a way that the users can clearly navigate around. The recommended documents for each post could be accessed by clicking on a button right below the post. A collapsible and scrollable drop down interface appears which contains the top ten relevant documents for the posts. After a user has gone through the documents, he/she can hide away the documents and move on to the next question.