Ravn Backend Role Recruitment Exercise

This problem should not take more than two hours to complete. Please read the specification carefully to avoid unnecessary work.

Problem

The objective of this exercise is to build a basic search engine. Design and build a solution that can index documents into an in-memory search index and allow queries to be run against it.

Indexing

We have provided a CSV of a small number of documents. These are BBC News articles that were crawled using our RAVN Pipeline product. Your solution will accept a file path to this CSV document as a command line argument.

The format of the CSV is as follows (ie, a CSV line per document):
<id>,<title>,<body>

Querying

We expect to enter one or more search terms and receive an unsorted set of results that contain the search terms in either the title or body. Each result should be on its own line and be of the form:
<id>,<title>
Your solution should accept a search query as a command line argument. Results should be printed to the console.

Your search engine should be case-insensitive.

Interface

You should run your application at the command line as follows:

```
java -jar search.jar <input file CSV> [search term(s) ...]
```

To search the index for documents that contain the terms ‘PlayStation’ **and** ‘Xbox’ in documents.csv, for example, we expect to run your solution like this:

```
java -jar search.jar documents.csv PlayStation Xbox
```

Scope

We do not expect solutions to consider

- Result sorting/scoring
- Scaling
- Syncing to disk
- Stemming, Stop words
- Other standard query features in full text search solutions
  (phrase matching, faceting, hit highlighting, clustering etc)

You may be asked about how you would approach these problems in interview.

Limitations

Your solution may not use any libraries, except:
Apache Commons, Google's Guava, Logging & Unit testing frameworks.

Reviewing your solution

You should provide us with your project source files as a zip and also send the built search.jar artifact.

We will review your submissions with focus on

- Data Structures & Algorithms
- Appropriate use of design patterns
- Unit testing
- Performance
- Approach, style and readability

We would like to see solutions which build an in-memory data structure from the input file. Although we will only be running a single query against it, imagine a scenario where in the future we would like to reuse this data structure to avoid re-reading the data file for new queries.