**Implementation Report**

**Task 1:**

**Lucene.java** indexes the corpus files using the org.apache.lucene.index.IndexWriter library and retrieves documents for a query using org.apache.lucene.search.IndexSearch library.

The analyzer used by this class is ‘SimpleAnalyzer’ as it mainly splits the words in the corpus using non-letter characters and lowercase them later on. This probably helps us to consider Lucene.java as a baseline to compare the results generated by another retrieval model ‘BM25’ described in Task 2

For the better understanding, I have added 2 methods in the starter program (shifted code from Main method):

1. indexer
2. retriever

The queries are automatically loaded from queries.txt file. If you wish to add more queries or check the results, please refer readme.txt file for more information.

**Task 2:**

**RetrievalModel** is a Parent class containing all the vital features:

* loads query from input file
* fetches index entries
* processes index entries which can be used retrieval model subclasses
* Saves the result in the desired output format

The ‘computeScore’ function in this class is kept unimplemented to accommodate Retrieval Models subclasses for future implementation. For this assignment, the retrieval model used is ‘BM25 Model’.

**BM25Model** is a Subclass which inherits all the properties from Retrieval Model Parent class. It overrides the ‘computeScore() function which is built on BM25 Model’.

Procedure:

* Firstly, it computes ‘scoring function’ containing the probability of the relevant(pi) and non-relevant documents(si) with some relevant information (ri=0, R=0 for our case).
* It then finds the doc\_term\_weight (fi = ‘number of documents where this term has appeared’)
* Next is finding query\_term weight (qfi = ‘frequency of term in the query’) component.
* The sum of the product of these 3 components for every word in the query is BM25 score for a given document.
* The function repeats the same evaluation for each document-query word in the index postings and outputs top 100 hits of the document IDs.

For this assignment, since no stopping or stemming techniques were used, the system\_name in the output is simply – ‘BM25Model’

Note: Since, the index entries for all the queries were loaded at once, query latency factor for this assignment was not considered.