**REPORT** **Index Building**

**Part A)**

Language Used : JAVA

Functions :

|  |  |  |  |
| --- | --- | --- | --- |
| FUNCTION NAME | INPUT | OUTPUT | INFORMATION |
| createHasMapEntry() | List<String> wordlist  Integer DocumentNumber  List<String> stopWords | Void | Iterates through all the wordlist and adds the word to HashMap along with the document Number if the word is not a stop-word. |
| createPositionHasMapEntry() | List<String> wordlist  Integer DocumentNumber  List<String> stopWords | Void | Iterates through all the wordlist and adds the word to HashMap along with the document Number and its position if the word is not a stop-word. |
| getFileNames() | NULL | LIST<String> fileNames | Returns the list with files names available in the Input folder |
| getFiledata() | String fileName | StringBuilder filedata | Returns the data in the given parameter file |
| tokenizer() | String text | List<String> tokenizedList | Tokenizes the file text to return the list of words from the text |
| writeHashDatatoFile() | HashMap<String,String> Hmap  String name | Void | Creates a file with name as given input Name+current timestamp and write the HasMap data to the file |
| getStopWords() | NULL | List<String> stopwords | Reads the stop words from the file and creates a list of all the stop words |

Implementation :

Note : Setting the document count equal to 1 initially and have created two HashMaps one for Frequency index and another for Position index

Step 1 : get all the **stopWords** using **getStopWords()**

Step 2 : get all the **fileNames** using **getFileNames()**

Step 3 : Iterate through the list of File Names

Step 4 : get file data using **getFiledata(***fileName***)**

Step 5 : get file words by tokenizing the file data **tokenizer(***fileData***)**

Step 6 : create entry for the list of words

**createHashMapEntry(** *fileWords, DocumentCount, stopWords***)**

Step 7 : create entry for the Position HashMap for list of words

**createPositionHashMapEntry(** *fileWords, DocumentCount, stopWords***)**

Step 8 : Increase the DocumentCount by 1

Step 9 : if More files : go to Step 4 else go to step 10

Step10: Write the created HashMap to the file **writeHashDatatoFile(***HashMap,”Frequency”***)**

Step11: Write the created Position HashMap to the file **writeHashDatatoFile(***HashMap,”Position”***)**

**Part B)**

1. **Number of Document indexed**

Total 500 documents were indexed present in the input folder.

1. **Vocabulary**

I have run the program twice

1. Including the Stop-Words in the list

The total number of unique terms discovered where **45,777**

1. After adding the filter to remove the stop-words

The total number of unique terms discovered where **35,463**

**Please refer the file with the unique words**

1. **Including the Stop-words**



1. **After adding the filter to remove the Stop-word**

**Frequency Based index file :** 

**Position based index file :** 