```
import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;
import java.util.HashSet;
import java.util.Objects;

/**
 * This is my code! Its goal is to run the search engine
 * CS 312 - Assignment 9
 * @author Victoria Shelton
 */

public class CLI {

    protected String[] args;

    public CLI(String[] a) {
            args = a;
        }
        protected boolean documentFlag;

        private void usage() {
            System.out.println("Usage: <stop list filename> <documents filenames> <-d> (optional)")
;
        }

        public void parse() {
            // If there is no argument, do it again
            if (args.length == 0) {
                usage();
                return;
            }

            // Search
            else {
                SearchEngine engine = new SearchEngine();
                long startTime = 0;

                // Take in stoplist file
                StopList stopList = new StopList(args[0]);

                // Take in document files and check for flag
                for (int i = 1; i < args.length; i++) {
                    if(Objects.equals(args[i], "-d"))
                        documentFlag = true;
                    else
                        engine.addDoc(args[i]);
                }

                // Take in queries
                BufferedReader br = new BufferedReader(new InputStreamReader(Sys
tem.in));
                while(true) {
                    System.out.println("Enter query: ");

                    try {
                        String query = br.readLine();

                        //Start time
                        startTime = System.currentTimeMillis();

                        // debug
                        if(Objects.equals(query, "@@debug")){
                            System.out.println("Index: ");
                            System.out.println(engine.debug());
                        }
                        // terminate
                        else if (Objects.equals(query, "TERMINATE")){
                            break;
```

```
                        }
                        // search
                        else {
                            // Strip stopwords
                            query = stopList.removeStopWords(query);
                            HashSet<Document> result = engine.search(query);
                            System.out.println(query + " found in " + (result == nu
ll ? 0 : result.size() + " documents"));
                            System.out.println(result);
                            // if documentflag is true, then print the contents
of each document
                            if (documentFlag) {
                                for (Document d : result) {
                                    System.out.println(d.printContents());
                                }
                            }
                        }

                    }
                    // If invalid
                    catch (IOException e) {
                        e.printStackTrace();
                        System.out.println("Invalid query, try again");
                    }

                    // Stop time
                    long stopTime = System.currentTimeMillis();
                    long elapsedTime = stopTime - startTime;
                    System.out.println("@@ search took " + elapsedTime +
                            "ms");
                }
            }
        }

    public static void main(String [] args)
    {
        CLI  c = new CLI(args);
        c.parse();
    }

}
```

Dec 11, 22 1:20 **Document.java** Page 1/1

```
import java.io.File;
import java.io.FileNotFoundException;
import java.util.Scanner;

/**
 * This is my code! Its goal is to keep track of a document and index it
 * CS 312 - Assignment 9
 * @author Victoria Shelton
 */


public class Document{

    protected String name;
    protected String contents;

    public Document(String filename)
    {
        try {
            File file = new File(filename);
            name = file.getName();
            Scanner scanner = new Scanner(file);

            while (scanner.hasNextLine()) {
                String temp = scanner.nextLine();
                if(contents==null)
                    contents = temp;
                else
                    contents = contents + "\n" + temp;
            }
        }
        catch (FileNotFoundException f){
            System.out.println("File not found.");
            f.printStackTrace();
        }
    }

    public String toString() {
        return name;
    }

    public String printContents() {
        return "[" + name + "]" + "\n" + contents;
    }
}
```

Dec 11, 22 1:20 **SearchEngine.java** Page 1/1

```
import java.io.*;
import java.util.HashMap;
import java.util.HashSet;
import java.util.Scanner;

/**
 * This is my code! Its goal is to search documents by their contents
 * CS 312 - Assignment 9
 * @author Victoria Shelton
 */

public class SearchEngine {

    HashMap<String,HashSet<Document>> index = new HashMap<>();
    HashSet<Document> documents = new HashSet<>();


    // Add a document to the set of documents, and indexes its words
    public void addDoc(String filename) {
        // Format and add the doc
        Document doc = new Document(filename);
        documents.add(doc);

        // Add its words to the index
        try {
            File file = new File(filename);
            Scanner scanner = new Scanner(file);
            scanner.useDelimiter("[EM-^Fa-zA-Z]+");
            // Hashset to put in the Hashmap
            HashSet<Document> tempHash = new HashSet<>();
            // read each line
            while (scanner.hasNextLine()) {
                String temp = scanner.nextLine();
                temp = temp.trim().toLowerCase();
                // If the word already exists, add on to the doc list
                if(index.containsKey(temp))
                    tempHash = index.get(temp);
                tempHash.add(doc);

                index.put(temp, tempHash);
            }
        } catch (FileNotFoundException e) {
            e.printStackTrace();
            System.out.println("File not found");
        }
    }

        // search for the documents containing query
    public HashSet<Document> search(String query)
    {
        HashSet<Document> docList = new HashSet<>();
        if(index.containsKey(query))
            docList.addAll(index.get(query));
        return docList;
    }

    public HashMap<String, HashSet<Document>> debug() {
        return index;
    }
}
```

```java
import java.io.File;
import java.io.FileNotFoundException;
import java.util.HashSet;
import java.util.Scanner;

/**
 * This is my code! Its goal is to keep track of the stop list and clean queries
 * CS 312 - Assignment 9
 * @author Victoria Shelton
 */

public class StopList {

    protected HashSet<String> stopList;

    public StopList(String filename)
    {
        stopList = new HashSet<>();
        readStopList(filename);
    }

    // Reads the stoplist file
    private void readStopList(String filename)
    {
        try {
            File file = new File(filename);
            Scanner scanner = new Scanner(file);
            while (scanner.hasNextLine()) {
                stopList.add(scanner.nextLine());
            }
        }
        catch (FileNotFoundException f){
            System.out.println("File not found.");
            f.printStackTrace();
        }
    }

    // Remove stop words from a query
    public String removeStopWords(String query)
    {
        String [] dirty = query.split(" ");
        String clean = "";
        for (int i = 0; i < dirty.length; i++) {
            if (!stopList.contains(dirty[i]))
                clean = clean + " " + dirty[i];
        }
        clean = clean.strip();
        return clean;
    }
}
```