```
/**
* This is my code! Its goal is to create a class for the CLI
* CS 312 - Assignment 9
* @author Karis Woo
*/

import java.util.HashMap;
import java.util.HashSet;
import java.util.Set;
import java.util.Scanner;
import java.util.Iterator;
import java.io.*;

public class CLI
{
  /**
  *  Takes Command Line Input to execute
  */
  //Expected time: O(n)
  public static void main(String[] args)
  {
    if(args.length =< 1)
      usage(args);
    //process args
    else
    {
      //for each input file minus -d and stoplist
      boolean isDebug = false;
      int i=0;

      Scanner scan = new Scanner(System.in);

      if(args[0].equals("-d"))
      {
        isDebug =true;
        i++;
      }

      Stoplist stop = new Stoplist();
      HashSet<String> stoplist = stop.buildStoplist(args[i]);

      Index invertedIndex = new Index();

      for(int in=i; in<args.length; in++)
      {
        invertedIndex.buildIndex(args[in], stoplist);
      }

      while(scan.hasNextLine())
      {
        String searchFor = scan.nextLine();
        Search s = new Search(searchFor, invertedIndex.ind, isDebug);
      }
    }
  }

  /**
  *  Output usage message if insufficient args
  */
  //Expected time: O(1)
  void usage(String[] args)
  {
    System.out.println("java CLI <a stoplist> <the docs>");
  }
}
```



```
/**
* This is my code! Its goal is to create a class for the inverted index
* CS 312 - Assignment 9
* @author Karis Woo
*/

import java.util.HashSet;
import java.util.Set;
import java.io.*;
import java.util.HashMap;
import java.util.Scanner;
import java.util.Iterator;

/**
  *  Class that creates and builds the inverted index from given file and with t
he stoplist
  */
class Index
{
    protected HashMap<String, HashSet<File>> ind;

    /**
    *  creates empty inverted index with HashMap
    */
    //Expected time: O(1)
    public Index()
    {
      ind = new HashMap<>();
    }

    /**
    *  reads document from param docName and fills inverted index with word as k
ey and doc name as set value
    */
    //Expected time: O(n)
    public void buildIndex(String docName, HashSet<String> stoplist)
    {
      try
      {
        File file = new File (docName);
        BufferedReader br = new BufferedReader (new FileReader(file));
        String asRead = new Scanner(br).useDelimiter("\\A").next();

        StringReader sr = new StringReader(asRead);
        Iterator<String> ii = sr.iterator();

        for(String key : sr)
        {

          if(!stoplist.contains(key))
          {
              HashSet<File> fileList = ind.get(key);

              if(fileList==null)
                fileList= new HashSet<File>();

              fileList.add(file);
              ind.put(key, fileList);
          }
        }
        //close scanner
        br.close();
      }
      catch(Exception io)
      {
        io.printStackTrace();
      }
    }
}
```

```
/**
* This is my code! Its goal is to create a class for the search engine
* CS 312 - Assignment 9
* @author Karis Woo
*/

import java.util.HashSet;
import java.util.Set;
import java.io.File;
import java.util.HashMap;
import java.util.Scanner;
import java.util.Iterator;
import java.nio.file.Paths;
import java.nio.file.Path;
import java.io.*;

/**
*  Class that performs search of inverted index for query word(s) and retrieval
of set of files attacthed to query
*/
public class Search
{
  /**
  *  takes the query and retrieves the set associated with the key in inverted i
ndex. If query has multiple words, take matching first set then intersections of
 later sets
  */
  //Expected time: O(n)
  public Search(String query, HashMap<String, HashSet<File>> invIndex, boolean i
sDebug)
  {
    try
    {
      StringReader sr = new StringReader(query);
      Iterator<String> ii = sr.iterator();
      HashSet<File> docList = null;
      HashSet<String> strings = null;

      //Expected time: O(n)
      for(String target : sr)
      {
          if(invIndex.containsKey(target))
          {
            if(docList==null)
            {
              docList = new HashSet<File>();
              docList=invIndex.get(target);
            }
            else
            {
              HashSet<File> d=invIndex.get(target);
              docList.retainAll(d);
            }
          }
      }

      if(docList!=null)
      {
        strings = new HashSet<String>();
        for(File f: docList)
          strings.add(stripFiles(f));
      }

      System.out.println("query ''" + query + "''" + " returned " + strings);
      System.out.println("--- found in " + (strings == null ? 0 : strings.size()) +
 " documents");

      if(isDebug && docList!=null)
        debugFunction(docList);
    }
    catch(Exception io)
    {
      io.printStackTrace();
    }
  }

  /**
   *  strips the file path from the name
   */
  //Expected time: O(1)
  String stripFiles(File f)
  {
      String s = f.toString();
      Path p = Paths.get(s);
      return p.getFileName().toString();
  }

  /**
   *  prints the contents of the files if debug flag is true
   */
  //Expected time: O(n)
  void debugFunction(HashSet<File> docs)
  {
    try
    {
      for(File f: docs)
      {
        System.out.println(f);
        BufferedReader br = new BufferedReader (new FileReader(f));
        String asRead = new Scanner(br).useDelimiter("\\A").next();
        System.out.println(asRead);
        br.close();
      }
    }
    catch(Exception io)
    {
     io.printStackTrace();
    }
  }
}
```

```
/**
* This is my code! Its goal is to create a class for the string reader
* CS 312 - Assignment 9
* @author Dr. Binkley
*/
import java.util.Scanner;
import java.util.Iterator;

/**
  *  Class that filters out all ignored characters of the input string
  */
public class StringReader implements Iterable<String>
{
  protected String st = null;

  //Expected time: O(1)
  public StringReader(String s)
  {
    st = s;
  }

  //Expected time: O(1)
  public Iterator<String> iterator()
  {
    return new Scanner(st).useDelimiter("[^a-zA-Z]+");
  }
}
```



```
/**
* This is my code! Its goal is to create a class for the stoplist
* CS 312 - Assignment 9
* @author Karis Woo
*/
import java.util.HashSet;
import java.util.Set;
import java.util.Scanner;
import java.io.*;
import java.util.Iterator;

/**
  *  Class that holds the Stoplist - list of words to be ignored when building t
he index
  */
class Stoplist
{
  protected HashSet<String> stoplist;

  /**
  *  creates empty Stoplist with HashSet
  */
  public Stoplist()
  {
    stoplist = new HashSet<String>();

  }

  /**
  *  reads stoplist file and fills stoplist HashSet from name of stoplist file
  */
  public HashSet<String> buildStoplist(String stoplistStr)
    {
      try
      {
        File file = new File (stoplistStr);
        BufferedReader br;
        br= new BufferedReader (new FileReader(file));
        String asRead = new Scanner(br).useDelimiter("\\A").next();

        StringReader sr = new StringReader(asRead);
        Iterator<String> ii = sr.iterator();

        for(String s : sr)
          stoplist.add(s);

        br.close();
      }
      catch(Exception io)
      {
        io.printStackTrace();
      }
      return stoplist;
    }
}
```