

Date  
31/01/22



Neeraj Appan T073  
**SHETH L.U.J. COLLEGE OF ARTS &  
SIR M.V. COLLEGE OF SCIENCE & COMMERCE**  
Department of Computer Science

## Information Retrieval Practical -07

Aim - Implement a basic IR system using Lucene

Description -

Lucene -

Apache Lucene is a free and open-source search engine software library originally written in Java by Doug Cutting. It is supported by the Apache Software Foundation and is released under the Apache software license. Lucene is widely used as a standard foundation for non-research search applications.

Lucene core is a Java library providing indexing and search features as well as spell checking, hit highlighting and advanced analysis/tokenization capabilities.

Projects Services Files

Source History

```
1 package Lucene;
2
3 import java.io.IOException;
4
5 import org.apache.lucene.document.Document;
6 import org.apache.lucene.queryParser.ParseException;
7 import org.apache.lucene.search.ScoreDoc;
8 import org.apache.lucene.search.TopDocs;
9
10 public class LuceneTester {
11
12     String indexDir = "E:\\fffiles\\college pracs and projects\\IR\\Lucene\\Index";
13     String dataDir = "E:\\fffiles\\college pracs and projects\\IR\\Lucene\\Data";
14     Indexer indexer;
15     Searcher searcher;
16
17     public static void main(String[] args) {
18         System.out.println("Hanza Appari T073\\n");
19         // TODO Auto-generated method stub
20         LuceneTester tester;
21         try {
22             tester = new
23             tester.create
24             tester.search
25         }
26         catch (IOException e)
27         {
28             e.printStackTrace();
29         }
30         catch (ParseException e)
31         {
32             e.printStackTrace();
33         }
34     }
35     private void createIndex() throws IOException
```

Can be replaced with multicatch  
(Alt-Enter shows hints)

Debugger Console LuceneIR7 (run)

Run: Hanza Appari T073

Indexing E:\\fffiles\\college pracs and projects\\IR\\Lucene\\Data\\Record1.txt  
Indexing E:\\fffiles\\college pracs and projects\\IR\\Lucene\\Data\\Record2.txt  
Indexing E:\\fffiles\\college pracs and projects\\IR\\Lucene\\Data\\Record3.txt  
Indexing E:\\fffiles\\college pracs and projects\\IR\\Lucene\\Data\\Record4.txt  
Indexing E:\\fffiles\\college pracs and projects\\IR\\Lucene\\Data\\Record5.txt  
5 File indexed, time taken: 167 ms  
0 documents found, Time :16  
BUILD SUCCESSFUL (total time: 1 second)

Output Notifications

Projects | Services | Files

Start Page | index.html | LuceneIR7.java | LuceneTester.java | Indexer.java | LuceneConstants.java | Searcher.java | TextFileFilter.java

Source | History

```
1 package Lucene;
2
3 import java.io.File;
4 import java.io.FileFilter;
5 import java.io.FileReader;
6 import java.io.IOException;
7
8 import org.apache.lucene.analysis.standard.StandardAnalyzer;
9 import org.apache.lucene.document.Document;
10 import org.apache.lucene.document.Field;
11 import org.apache.lucene.index.CorruptIndexException;
12 import org.apache.lucene.index.IndexWriter;
13 import org.apache.lucene.store.Directory;
14 import org.apache.lucene.store.FSDirectory;
15 import org.apache.lucene.util.Version;
16
17
18
19 public class Indexer {
20     private IndexWriter writer;
21
22     public Indexer(String indexDirectoryPath) throws IOException {
23         //this directory will contain the indexes
24         Directory indexDirectory =
25             FSDirectory.open(new File(indexDirectoryPath));
26
27         //create the indexer
28         writer = new IndexWriter(indexDirectory,
29             new StandardAnalyzer(Version.LUCENE_36), true,
30             IndexWriter.MaxFieldLength.UNLIMITED);
31     }
32
33     public void close() throws CorruptIndexException, IOException {
34         writer.close();
35     }
36
37     private Document getDocument(File file) throws IOException {
38         Document document = new Document();
39
40         //index file contents
41         Field contentField = new Field(LuceneConstants.CONTENT, new FileReader(file));
42         //index file name
43         Field fileNameField = new Field(LuceneConstants.FILE_NAME,
44             file.getName(), Field.Store.YES, Field.Index.NOT_ANALYZED);
45         //index file path
46     }
47
48 }
```

Output | Notifications

70:1 | INS

LuceneIR7 - NetBeans IDE 8.2

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

Search (Ctrl+F)

Projects Services Files

Start Page index.html LuceneIR7.java LuceneTester.java Indexer.java LuceneConstants.java Searcher.java TextFileFilter.java

Source History

E:\oracle\netbeans\LuceneIR7\src\Lucene\LuceneConstants.java

```
1 package Lucene;
2
3 public class LuceneConstants {
4     public static final String CONTENTS = "contents";
5     public static final String FILE_NAME = "filename";
6     public static final String FILE_PATH = "filepath";
7     public static final int MAX_SEARCH = 10;
8 }
9
```

assignment  
CalcClientApp  
CalculatorServerApp  
checkpro  
DotNetWClient1  
Eshop  
fxcalc  
Source Packages  
Test Packages  
Libraries  
Test Libraries  
inheritance  
JavaApplication5  
JavaFXApplication1  
JavaFxFirst  
JSP\_Book\_Mgmtmt  
LuceneIR7  
Source Packages  
<default package>  
Lucene.java  
Lucene  
Indexer.java  
LuceneConstants.java  
LuceneTester.java  
Searcher.java  
TextFileFilter.java  
Lucene.Index  
write.lock  
LuceneIR7  
Test Packages  
Libraries  
Test Libraries  
methods  
miniproj  
onlinebanking  
p4client  
PLZ  
project  
Web Pages  
Source Packages  
<default package>  
MainP.java  
Prac2update.java  
loginprac5.java  
prac2.java  
prac2delete.java  
pracScookies.java  
Test Packages  
Libraries

Output Notifications

1:1 INS

Type here to search

18:21 31-01-2022

Projects | Services | Files

Start Page | index.html | LuceneIR7.java | LuceneTester.java | Indexer.java | LuceneConstants.java | Searcher.java | TextFileFilter.java

Source | History

```
1 package Lucene;
2
3 import java.io.File;
4 import java.io.IOException;
5
6 import org.apache.lucene.analysis.standard.StandardAnalyzer;
7 import org.apache.lucene.document.Document;
8 import org.apache.lucene.index.CorruptIndexException;
9 import org.apache.lucene.queryParser.ParseException;
10 import org.apache.lucene.queryParser.QueryParser;
11 import org.apache.lucene.search.IndexSearcher;
12 import org.apache.lucene.search.Query;
13 import org.apache.lucene.search.ScoreDoc;
14 import org.apache.lucene.search.TopDocs;
15 import org.apache.lucene.store.Directory;
16 import org.apache.lucene.store.FSDirectory;
17 import org.apache.lucene.util.Version;
18
19 public class Searcher {
20     IndexSearcher indexSearcher;
21     QueryParser queryParser;
22     Query query;
23
24     public Searcher(String indexDirectoryPath)
25         throws IOException {
26         Directory indexDirectory =
27             FSDirectory.open(new File(indexDirectoryPath));
28         indexSearcher = new IndexSearcher(indexDirectory);
29         queryParser = new QueryParser(Version.LUCENE_36, LuceneConstants.CONTENTS,
30             new StandardAnalyzer(Version.LUCENE_36));
31     }
32
33     public TopDocs search(String searchQuery)
34         throws IOException, ParseException {
35         query = queryParser.parse(searchQuery);
36         return indexSearcher.search(query, LuceneConstants.MAX_SEARCH);
37     }
38
39     public Document getDocument(ScoreDoc scoreDoc)
40         throws CorruptIndexException, IOException {
41         return indexSearcher.doc(scoreDoc.doc);
42     }
43
44     public void close() throws IOException {
45         indexSearcher.close();
46     }
47 }
48
```

Output | Notifications

1:1 | INS



Projects | Services | Files

Start Page | index.html | LuceneIR7.java | LuceneTester.java | Indexer.java | LuceneConstants.java | Searcher.java | TextFileFilter.java

Source | History

```
1 package Lucene;
2
3 import java.io.File;
4 import java.io.FileFilter;
5
6 public class TextFileFilter implements FileFilter{
7
8     public boolean accept(File pathname)
9     {
10         return pathname.getName().toLowerCase().endsWith(".txt");
11     }
12 }
13
14
```

Output | Notifications

1:1 | INS



