# Lucene全文搜索框架

## Lucene简介

### 概念

Lucene是一个Apache提供的开源基于Java开发的全文搜索引擎框架.Lucene可以用来实现任何应用程序的搜索,单它不是一个真正的应用产品.

Lucene是可扩展的,高性能的搜索框架.

### 全文检索的概念

根据输入的关键字在整个项目内进行检索匹配的检索方法.

应用领域:搜索引擎,站内搜索,电商网站....

### 全文检索和like模糊查询的区别

#### .数据结构:

like模糊查询应该是结构化的数据.

全文搜索中数据可以是非结构化的;

#### .搜索原理:

like模糊查询利用的select \* from 表名 where like “%关键字%”查询语句;

全文检索需要先建立一个索引库,在索引库中存放大量的索引文件,每个索引文件都被称为文档.文档中存放的是关键词与对应的内容的映射关系.

#### .搜索效果:

##### 匹配的准确性:

like模糊查询的范围不够广阔,匹配的准确性不够;

全文检索的匹配准确度会更好;

##### 搜索速度:

全文检索速度更快.

### Lucene相关的API概念

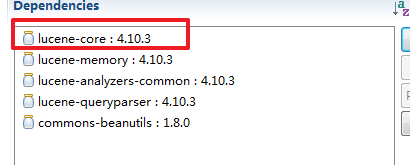
1. .Analyzer:分词器(分析器).它的作用就是根据一定的规则把一个字符串拆分成一个个的词语,并且要去掉其中一些无效的词语(停用词).去掉无效词语的目的在于缩小搜索词库的大小,提高搜索精度和命中率.

在Lucene中默认的分词器是英文的标准的分词器.中文分词器IKAnalyzer.

1. .Document:文档.索引库中存储的索引文件.在Document中包含用户要搜索的各种记录信息.
2. Field:域.一个Document中可以包含N个Field信息.
3. .Term:检索项.搜索中的最小单位,表示文档Document中的某一个词语.每一个Term由两部分来组成:它本身所表示的词语和该词语所在的field位置.
4. .Token:表示Term出现的位置和次数,利用该Token可以找到Term出现的位置.
5. .Segment:片段.在创建Document文档之前,会先创建出一个个的小的索引文件,这个索引文件就被称为Segment,最终这些小的索引文件会合并成一个大的Document.

## Lucene开发实现

### 1.添加依赖包



### 2.创建索引库,添加Document文档

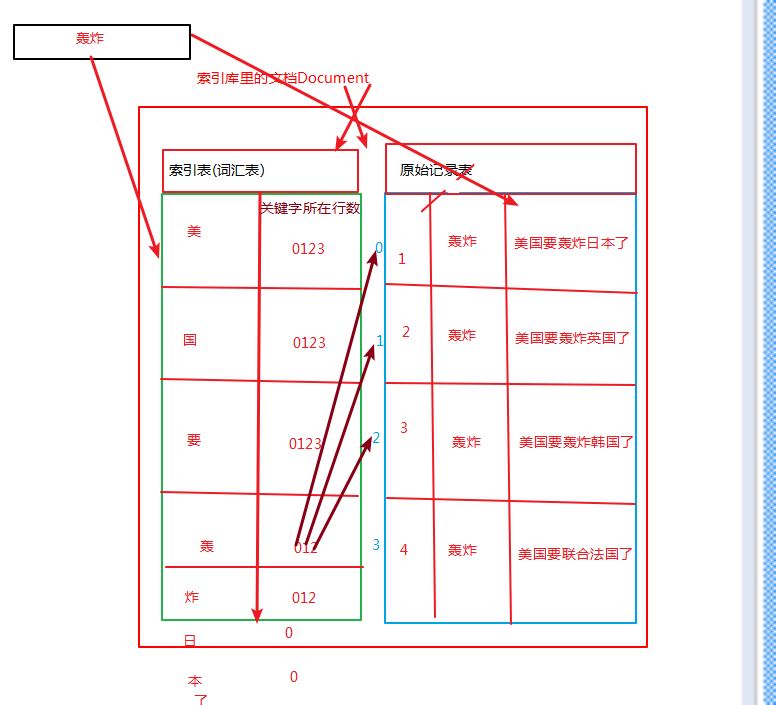


### 3.在索引库中根据关键字进行搜索





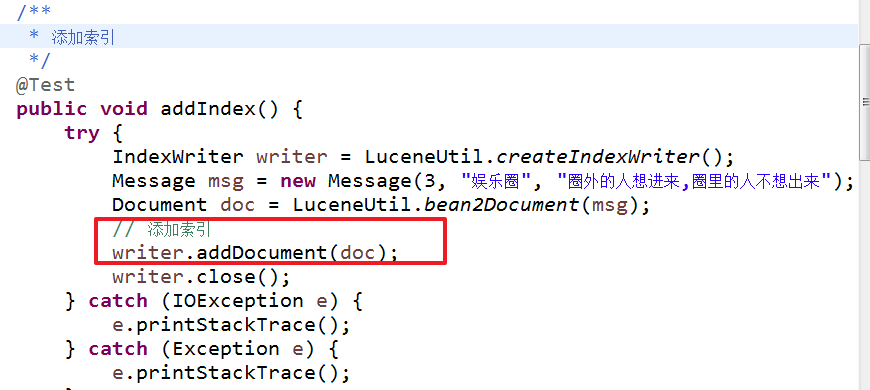
### 4.Lucene根据关键字检索原理(重点)



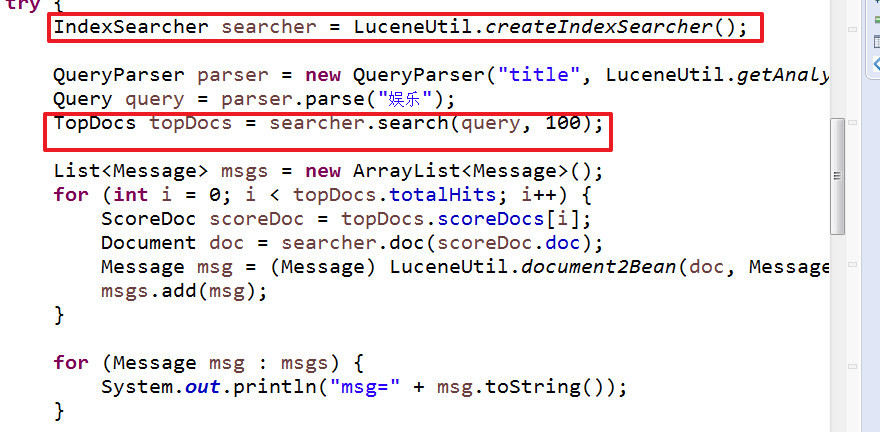
### 封装LuceneUtil工具类

### 对Lucene索引库进行CRUD

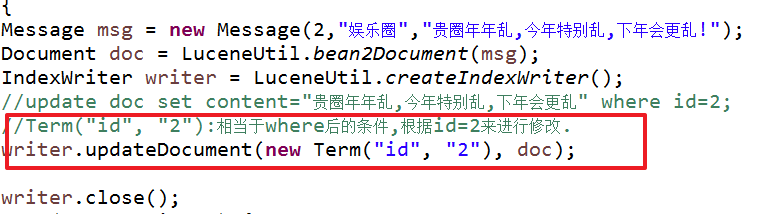
#### 添加索引:



#### 查询索引:

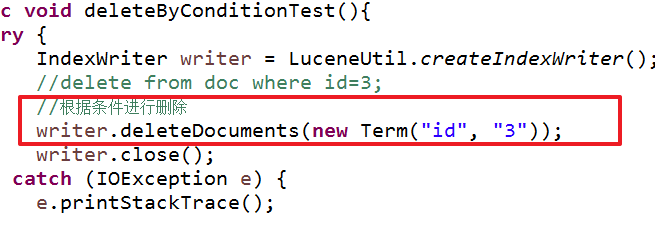


#### 修改索引:

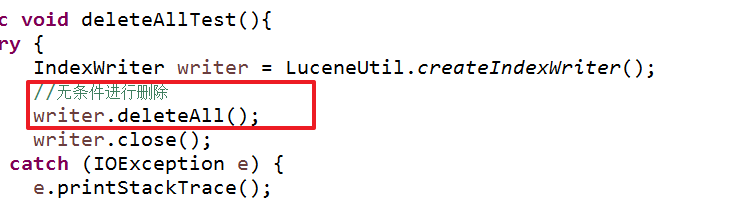


#### 删除索引:

##### 按条件删除:

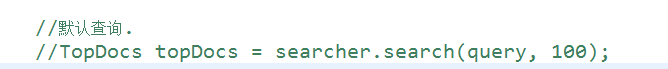


##### 全部删除:

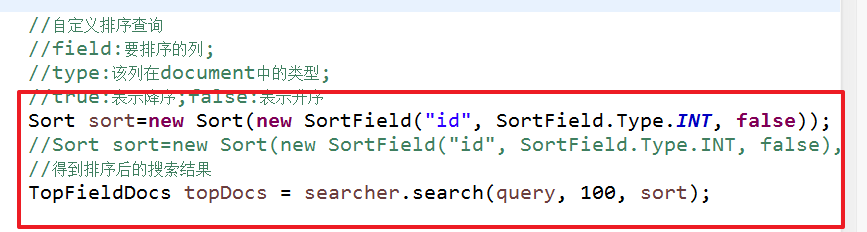


### 对检索结果进行排序

默认的排序是按照匹配相关的得分score来排序:

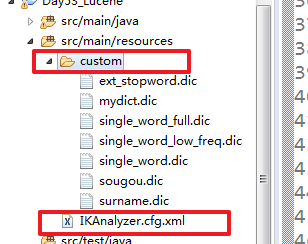


自定义排序:

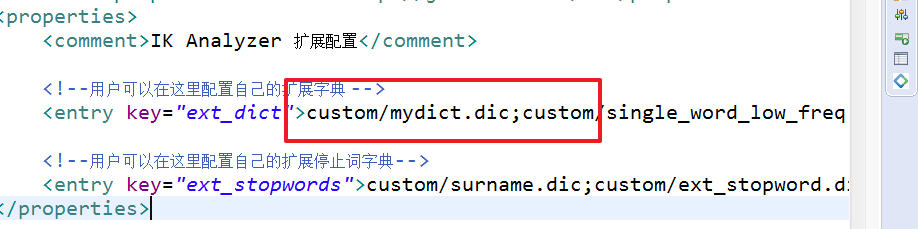


### 配置中文分词器

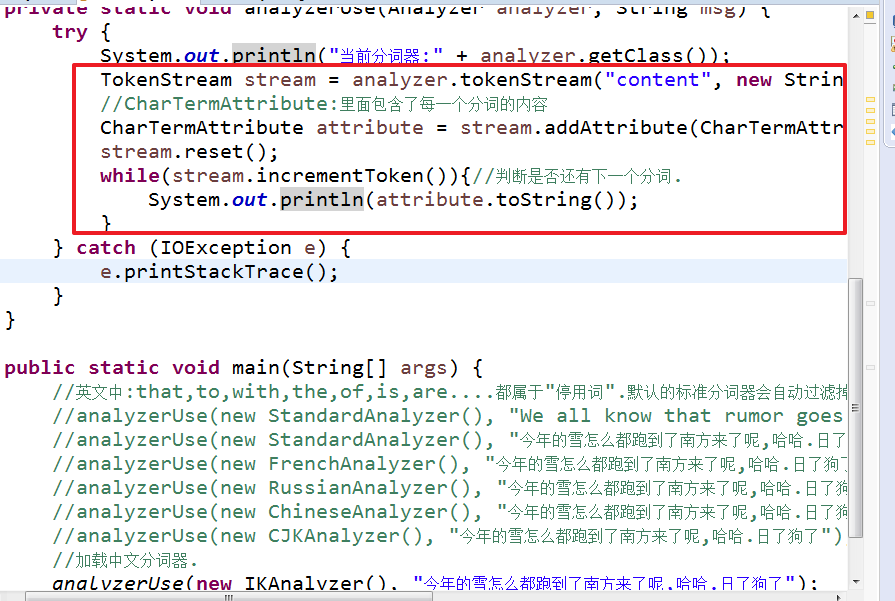
创建IKAnalyzer.cfg.xml配置文件,关联相关词汇表.



关联词汇表:



引用中文分词器进行分词:

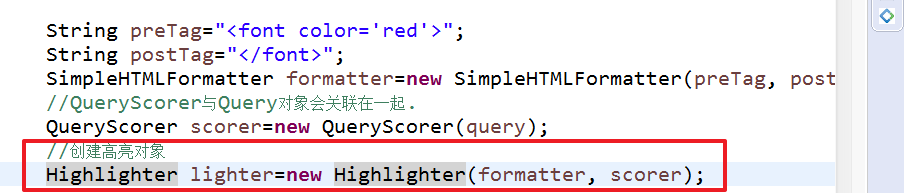


### 对检索项进行高亮显示

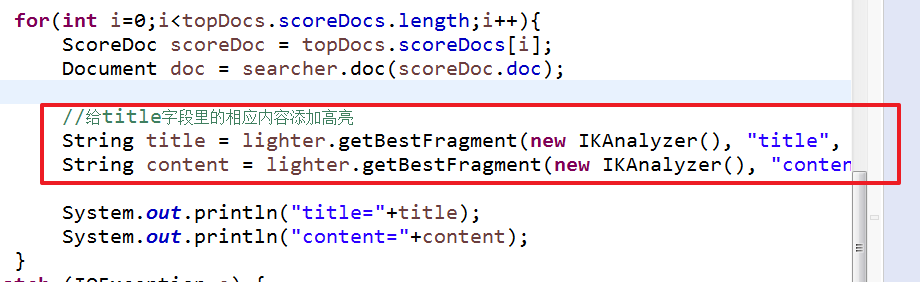
添加依赖包:



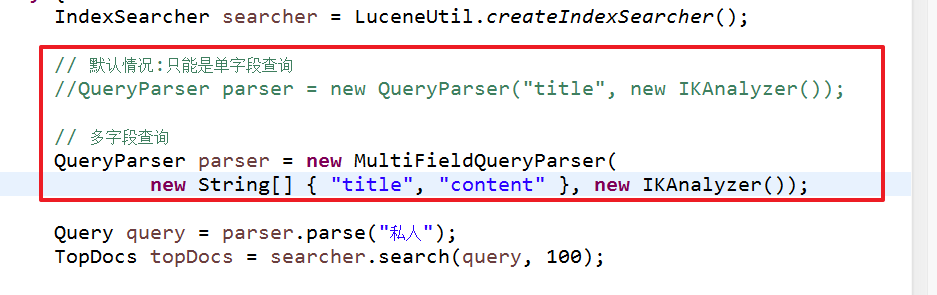
基本设置:



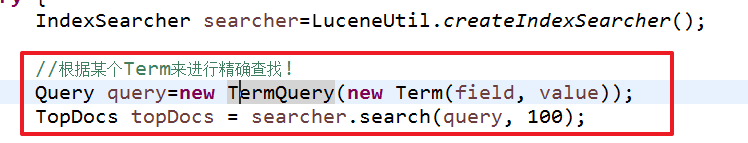
给指定的字段内容进行高亮:

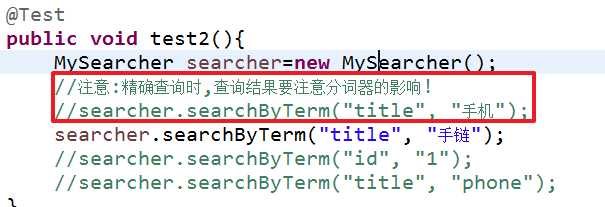


### 10.多字段查询的实现

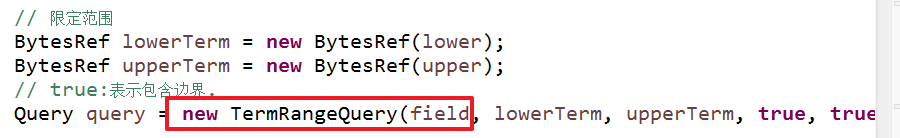


### 11.精确查询

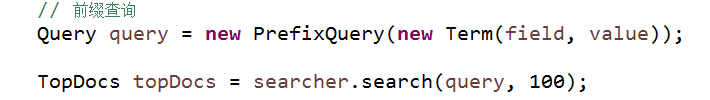




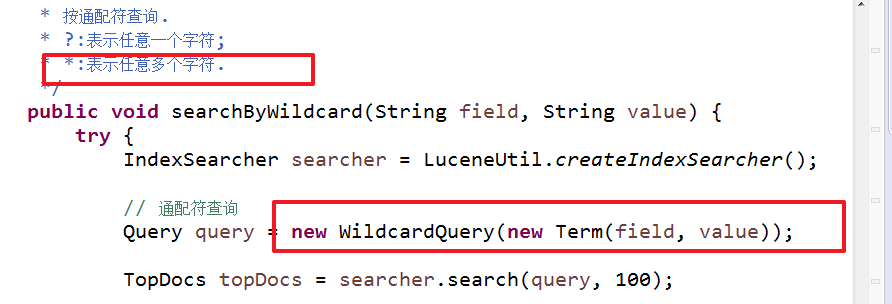
### 12.按范围查询



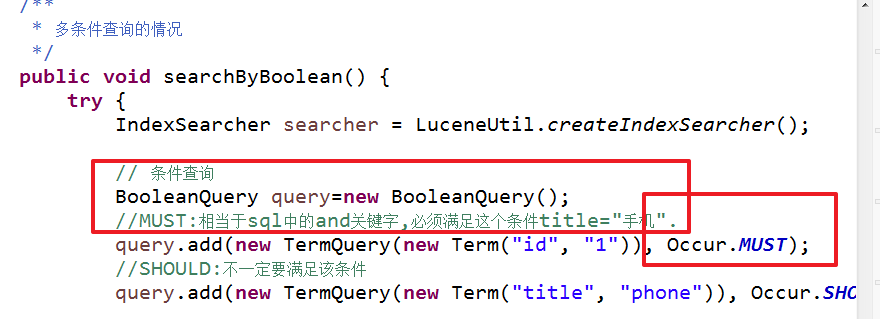
### 13.前缀查询



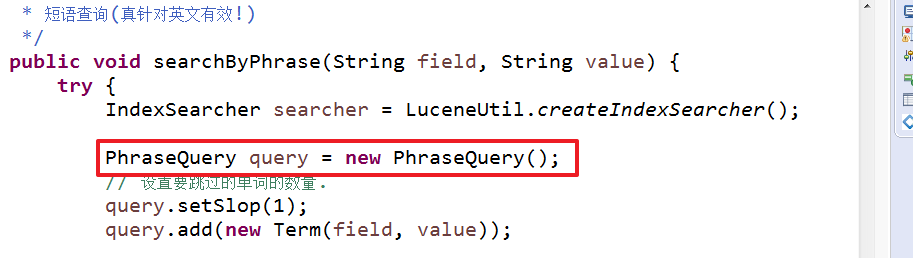
### 14.通配符查询



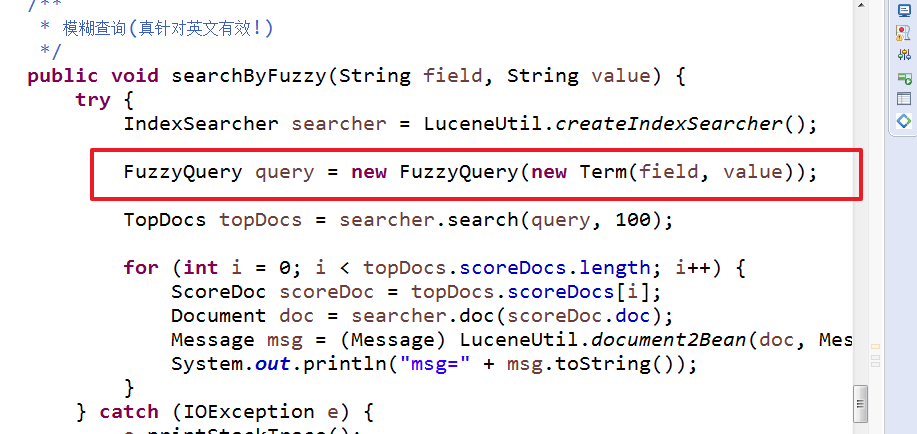
### 15.多条件查询



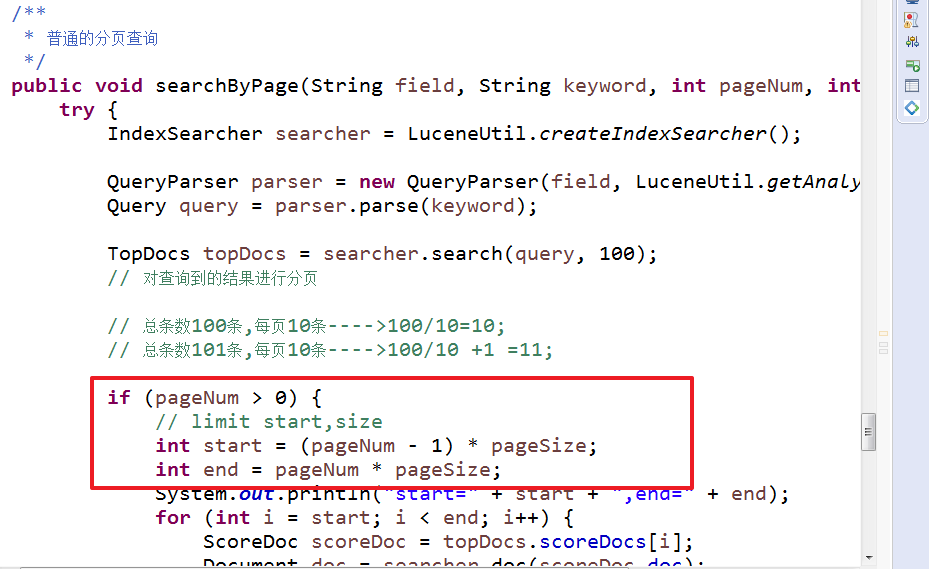
### 16.短语查询



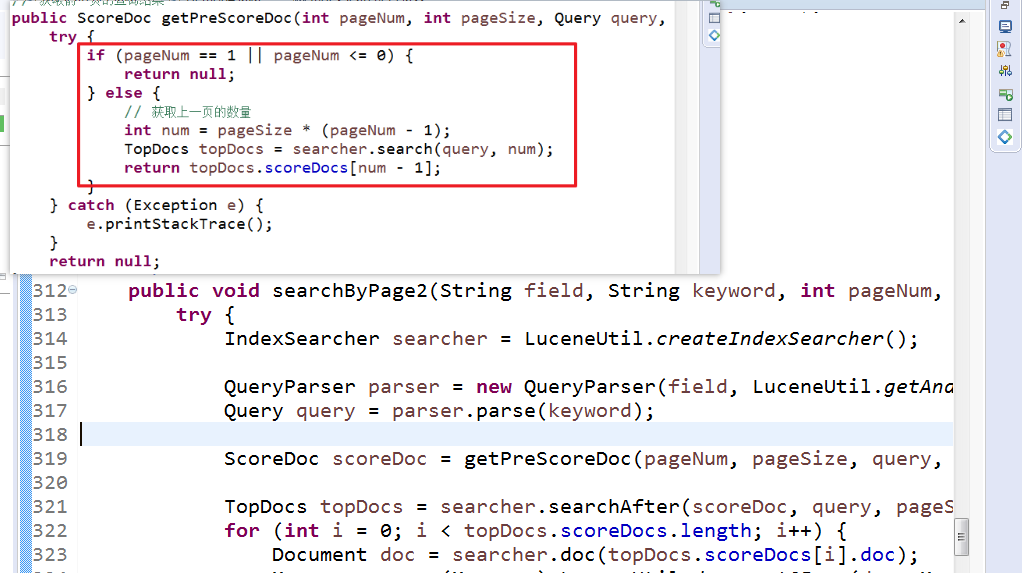
### 17.模糊查询



### 18.普通分页查询



### 19.改进的分页查询



### 作业:

前端写一个搜索页面----->有个输入框,按钮,仿百度;

后端:按照三层架构,web层+service层+dao层.利用Lucene实现实现;

要求:Lucene的索引库里的记录数据要求是从数据库中查询得到了.

预习:Solr!------->SolrJ