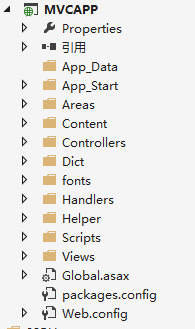
1. 首先在网站根目录的libs文件中添加我发过来的几个程序集，然后在类库Classlibrary中

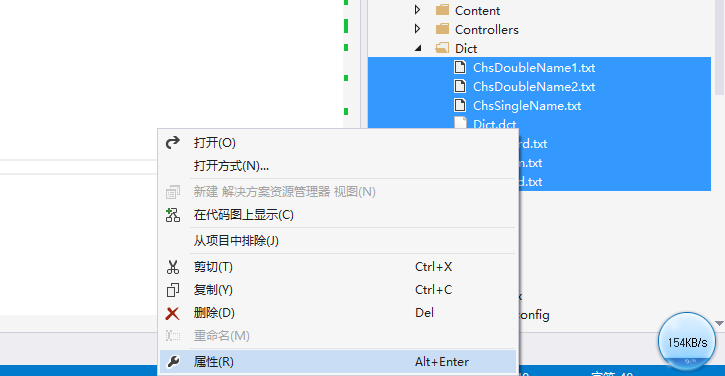
将新增的程序集全部引用。

2，把我发过来的Dict文件添加进MVCAPP项目中



如图中的位置

并且将Dict 文件夹里面的文件按如图操作



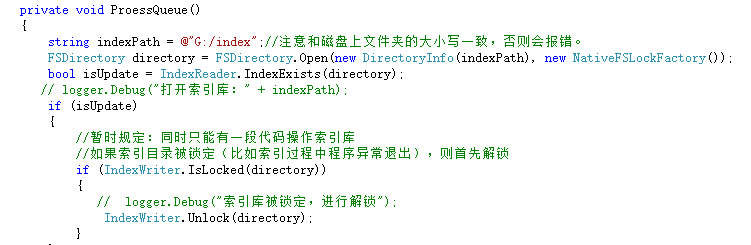
点击属性后将“复制到输出目录”改成“如果较新则复制”如图



在classlibrary中添加我发过来的两个类

IndexManager.cs 和splitWordsHelper.cs

IndexManager.cs里面需要改动一个地方：



Indexpath后面的地址是你在磁盘上的文件夹，这是用来存放索引库的，你在磁盘上某个位置新建一个index 然后把路径改一下就行。

4，①在classlibrary中有一个MarketDownstairs类，这里面是对楼下小铺进行操作的。

在里面找到以下函数

public bool UpLoadGoodInfo(ViewModel.UpLoadGoodModel model ,string imageAdress)

将里面的代码替换成以下

try

{

T007店铺货物表 newGood = new T007店铺货物表();

newGood.GoodIntroduction = model.Introduction;

newGood.GoodName = model.GoodName;

newGood.GoodPhoto = imageAdress;

newGood.GoodPrice = model.GoodPrice;

newGood.ShopID = model.shopId;

newGood.GoodNumber = model.GoodNumber;

//将记录加入数据库

operateContext.BLLSession.IT007店铺货物表BLL.Add(newGood);

///下面是创建索引的代码。

JobInfo jobinfo = new JobInfo { GoodId = newGood.GoodID, JobType = JobType.Add };

IndexManager.Instance.PutJob(jobinfo);//把索引任务放入队列。

return true;

}

catch (Exception)

{

throw;

}

②找到

public GoodsListModel getRightGoods(string keyWords, int pageSize,int page)

替换成如下（注意红色字体部分的地址改成与上面的indexmanager一致）：

public GoodsListModel getRightGoods(string keyWords, int pageSize,int page)

{

///从词库中获得分词

List<string> kw= splitWordsHelper.SplitWords(keyWords)as List<string>;

FSDirectory directory = FSDirectory.Open(new DirectoryInfo(@"G:\index"),

new NoLockFactory());

IndexReader reader = IndexReader.Open(directory, true);

IndexSearcher searcher = new IndexSearcher(reader);

PhraseQuery queryMsg = new PhraseQuery();//查询条件

foreach (string word in kw)

{

queryMsg.Add(new Term("GoodName", word));//Contains("body",word)

}

queryMsg.SetSlop(100);//词的距离超过100就不匹配

PhraseQuery queryTitle = new PhraseQuery();//查询条件

foreach (string word in kw)

{

queryTitle.Add(new Term("GoodIntroduction", word));//Contains("body",word)

}

queryTitle.SetSlop(100);//词的距离超过100就不匹配

//复合查询，在标题和正文中搜索

//Should是Or，Must是And

BooleanQuery boolQuery = new BooleanQuery();

boolQuery.Add(queryMsg, BooleanClause.Occur.SHOULD);

boolQuery.Add(queryTitle, BooleanClause.Occur.SHOULD);

//Stopwatch stopwatch = new Stopwatch();

//stopwatch.Start();

//foreach (string word in kw.Split(' '))//先用空格，让用户去分词，空格分隔的就是词“计算机 专业”

//{

// query.Add(new Term("msg", word));//Contains("body",word)

//}

//BooleanQuery

//where Contains("body","计算机") and Contains("body","专业")

TopScoreDocCollector collector = TopScoreDocCollector.create(1000, true);//盛放搜索结果的容器

searcher.Search(boolQuery, null, collector);//用query这个查询条件进行搜索，搜索结果放入collector容器中

List<T007店铺货物表> list = new List<T007店铺货物表>();

// collector.GetTotalHits()查询结果的总条数

ScoreDoc[] docs = collector.TopDocs(0, collector.GetTotalHits()).scoreDocs;

for (int i = 0; i < docs.Length; i++)

{

int docId = docs[i].doc;//文档编号（lucene.net内部分配的，和number无关）

Document doc = searcher.Doc(docId);//根据文档编号拿到文档对象

int id = Convert.ToInt32(doc.Get("GoodID"));//取出文档的number字段的值。必须是Field.Store.YES才能取出来

string goodIntroduction = doc.Get("GoodIntroduction");

string goodName = doc.Get("GoodName");

int goodNumber=Convert.ToInt32( doc.Get("GoodNumber"));

string goodPhotoUrl=doc.Get("GoodPhoto");

decimal goodPrice=Convert.ToDecimal( doc.Get("GoodPrice"));

int shopid=Convert.ToInt32( doc.Get("ShopID"));

T007店铺货物表 good = new T007店铺货物表();

good.GoodID=id;

good.GoodName=goodName;

good.GoodIntroduction=goodIntroduction;

good.GoodNumber=goodNumber;

good.GoodPhoto=goodPhotoUrl;

good.GoodPrice=goodPrice;

good.ShopID=shopid;

//sr.Id = id;

//sr.Msg = HighLight(kw, msg);

//sr.Title = title;

//sr.Url = url;

list.Add(good);

}

GoodsListModel goodlist=new GoodsListModel()

{

goodsList=list,

pageInfo=new PagingInfo{ currentpage=page, itemperpage=pageSize, Totalitems=list.Count()}

};

return goodlist;

// return operateContext.BLLSession.IT007店铺货物表BLL.GetListBy(m => m.GoodName==keyWords).Skip((page-1)\*pageSize).Take(pageSize).ToList();

}

③找到public bool DeleteGoods(int id)

替换成如下：

public bool DeleteGoods(int id)

{

var t007 = operateContext.BLLSession.IT007店铺货物表BLL.GetListBy(m => m.GoodID==id).FirstOrDefault();

if (operateContext.BLLSession.IT007店铺货物表BLL.Del(t007) == 1)

{

///下面是创建索引的代码。

JobInfo jobinfo = new JobInfo { GoodId = t007.GoodID, JobType = JobType.Delete };

IndexManager.Instance.PutJob(jobinfo);//把索引任务放入队列。

return true;

}

else

{

return false;

}

}

4，在MVCAPP的controller中找到MarketDownstairsController

找到：public ActionResult GetRightGoods(int page = 1) 函数

替换成：

/// <summary>

/// 根据关键词获得指定的货物

/// </summary>

/// <returns></returns>

public ActionResult GetRightGoods(int page = 1)

{

string keyWords = Server.HtmlDecode(this.Request.Form["keyWords"]);

if (keyWords != null)

{

Session["key"] = keyWords;

return View(new ClassLibrary.MarketDownstairs().getRightGoods(keyWords,page,pageSize));

}

else

{

if (Session["key"] != null)

{

return View(new ClassLibrary.MarketDownstairs().getRightGoods(keyWords, page, pageSize));

}

return RedirectToAction("MarketIndex");

}

}

5，最后一步。

在MVCAPP/global.asax的 protected void Application\_Start()

下面添加一行代码如下（先using classlibrary）：

IndexManager.Instance.Start();///启动任务

5，测试的时候先上传货物（数据库新增的同时创建了一条索引）再进行搜索。主要是通过盘古分词匹配货物简介和货物名字。