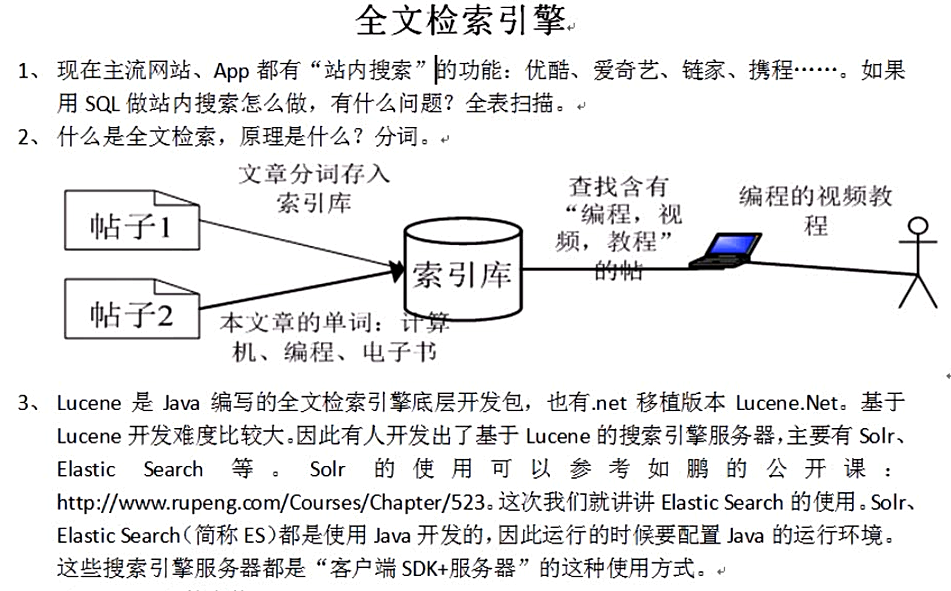
全文检索与es

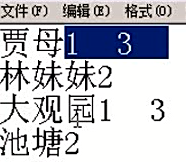
是

全文检索引擎：

5-全文检索概述：



每个词在哪几页出现过，都记录一下。

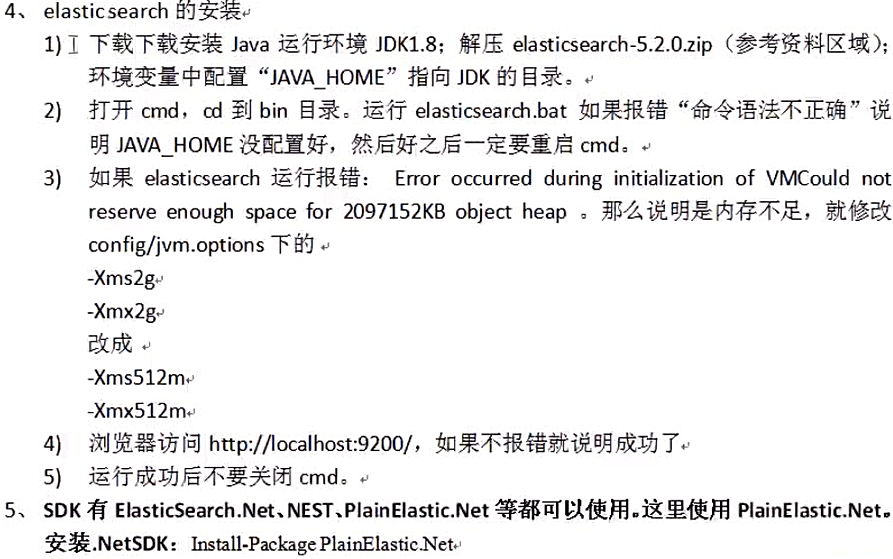


把es当做一个高级的数据库拿来用就可以了，别人都做好了，我们现成拿来用，把数据给他，他把结果给我们，我们展示就可以了。

难道这就是SDK+服务器（也就是es服务器）？

我们先把es的服务器给跑起来，咱们再写程序连上服务器，再把数据扔给他，让他帮咱们进行搜索。

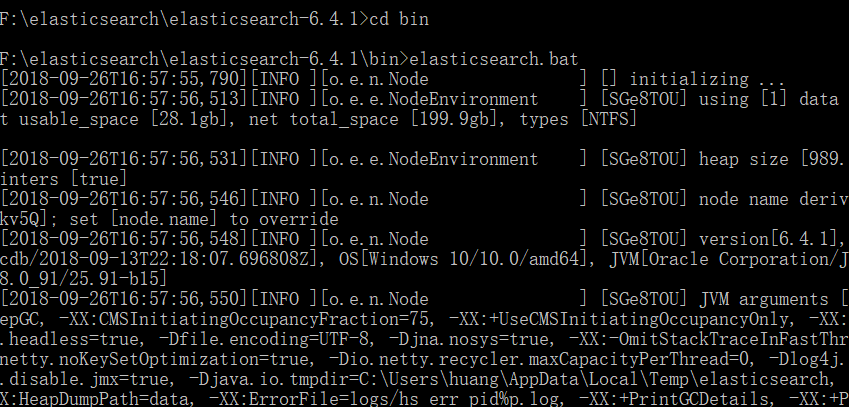
6-ElasticSearch：



我已经安装好jdk和ElasticSearch



执行以后没有报错：

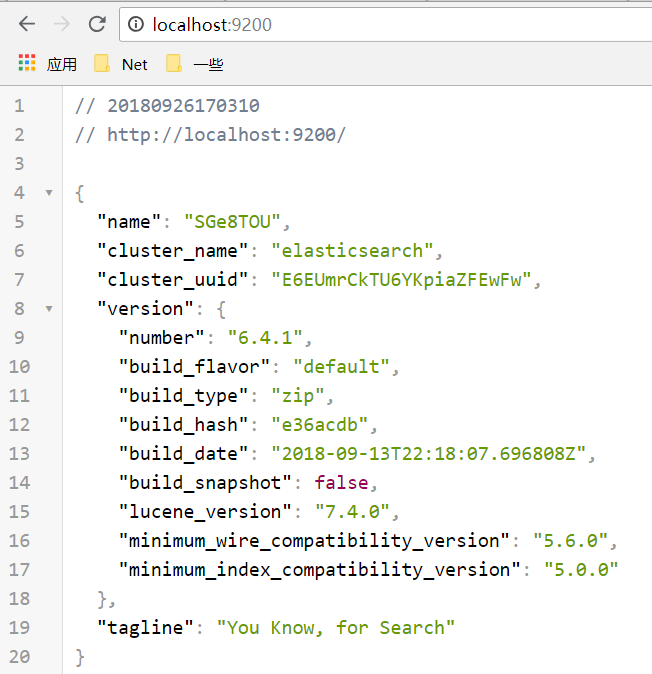


老师的报错了：

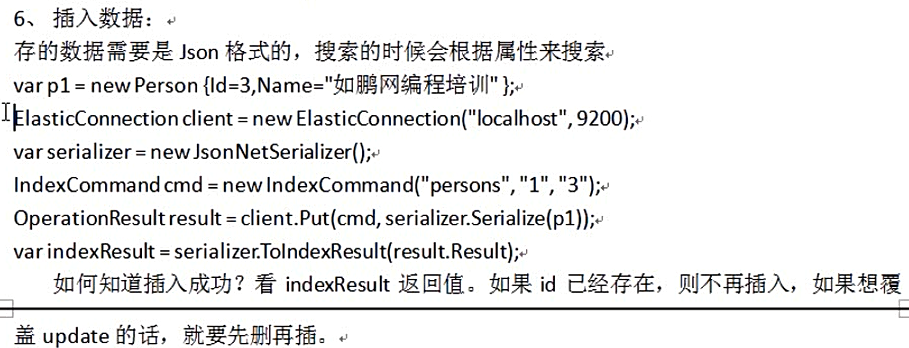


这个报错就是可用内存太小了。

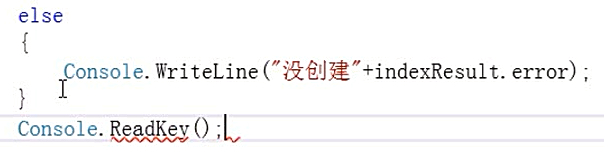
启动起来了：



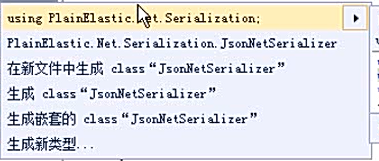
安装PlainElastic







这里就相当于放到persons数据库当中。

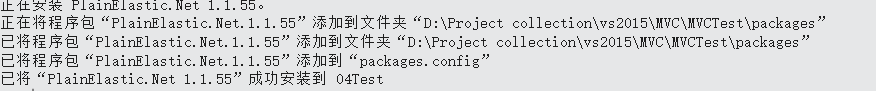


再创建一个数据。

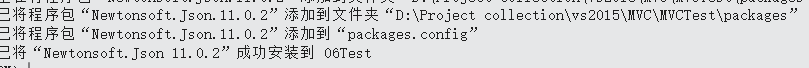


我自己练习：

下载成功：



有异常，安装了最新的newtonsoft.json



还是创建失败了，不知道什么原因。



Total应该指的就是所有的结果，类似百度查询的总结果数。



我自己的代码，反正是没有创建，也没有查询成功，还不知道为什么。

namespace \_06Test

{

class Program

{

//static void Main(string[] args)

//{

// Person p = new Person();

// p.Id = 16;

// p.Name = "花2";

// p.Age = 17;

// p.Desc = "333";

// //ElasticConnection client = new ElasticConnection("");

// //var serializer = new JsonNetSerializer();

// //IndexCommand cmd = new IndexCommand("zsz", "persons", p.Id.ToString());

// //OperationResult result = client.Put(cmd, serializer.Serialize(p));

// //var indexResult = serializer.ToIndexResult(result.Result);

// //if (indexResult.created)

// //{

// // Console.WriteLine("创建了");

// //}

// //else

// //{

// // Console.WriteLine("创建失败了，原因：" + indexResult.error);

// //}

// ////第二个版本

// ElasticConnection Client = new ElasticConnection("localhost", 9200);

// var serializer = new JsonNetSerializer();

// var jsonDocument = serializer.Serialize(p);

// string cmd = new IndexCommand("zsz", "persons", p.Id.ToString());

// OperationResult result = Client.Put(cmd, jsonDocument);

// var indexResult = serializer.ToIndexResult(result.Result);

// if (indexResult.created)

// {

// Console.WriteLine("创建了");

// Console.WriteLine(indexResult.\_id);

// }

// else

// {

// Console.WriteLine("创建失败了，原因：" + indexResult.error);

// Console.WriteLine(indexResult.\_id);

// }

// //Console.WriteLine(indexResult.\_id);

// Console.ReadKey();

//}

//查询

static void Main(string[] args)

{

////第二个版本

ElasticConnection Client = new ElasticConnection("localhost", 9200);

string cmd = new IndexCommand("zsz", "persons");

string query = new QueryBuilder<Person>()

//1 查询

.Query(b =>

b.Bool(m =>

//并且关系

m.Must(t =>

//分词的最小单位或关系查询

t.QueryString(t1 => t1.DefaultField("Name").Query("花"))

//.QueryString(t1 => t1.DefaultField("name").Query(key))

// t .Terms(t2=>t2.Field("intro").Values("研究","方鸿渐"))

//范围查询

// .Range(r => r.Field("age").From("100").To("200") )

)

)

)

//分页

//.From(from)

//.Size(size)

//排序

// .Sort(c => c.Field("age", SortDirection.desc))

//添加高亮

//.Highlight(h => h

// .PreTags("<b>")

// .PostTags("</b>")

// .Fields(

// f => f.FieldName("intro").Order(HighlightOrder.score),

// f => f.FieldName("\_all")

// )

// )

.Build();

string result = Client.Post(cmd, query);

var serializer = new JsonNetSerializer();

var searchResult = serializer.ToSearchResult<Person>(result);

//searchResult.hits.total//一共有多少匹配结果

//searchResult.Documents//当前页的查询结果

//Console.WriteLine(searchResult.hits.total);

foreach (var doc in searchResult.Documents)

{

Console.WriteLine(doc.Id + "," + doc.Name + "," + doc.Age + "," + doc.Desc);

}

Console.ReadKey();

}

}

}

看到：



下载了源码，做添加操作。



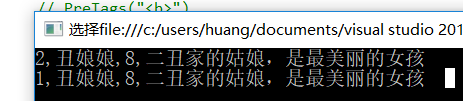
但是找不到路径。



似乎找到问题的所在了- - 没有引用对对象！不对不对还是不对。



终于成功了，原来是我ElasticSearch的版本太高了，用了老师的5.20就ok，创建成功以及查询成功了。



代码：

class Program

{

static void Main(string[] args)

{

//Person p1 = new Person();

//p1.Id = 2;

//p1.Age = 8;

//p1.Name = "丑娘娘";

//p1.Desc = "二丑家的姑娘，是最美丽的女孩";

//ElasticConnection client = new ElasticConnection("localhost", 9200);

//var serializer = new JsonNetSerializer();

////第一个参数相当于“数据库”，第二个参数相当于“表”，第三个参数相当于“主键”

//IndexCommand cmd = new IndexCommand("zsz", "persons", p1.Id.ToString());

////Put()第二个参数是要插入的数据

//OperationResult result = client.Put(cmd, serializer.Serialize(p1));

//var indexResult = serializer.ToIndexResult(result.Result);

//if (indexResult.created)

//{

// Console.WriteLine("创建了");

//}

//else

//{

// Console.WriteLine("没创建" + indexResult.error);

//}

//查询

ElasticConnection client = new ElasticConnection("localhost", 9200);

SearchCommand cmd = new SearchCommand("zsz", "persons");

var query = new QueryBuilder<Person>()

.Query(b =>

b.Bool(m =>

//并且关系

m.Must(t =>

//分词的最小单位或关系查询

t.QueryString(t1 => t1.DefaultField("Name").Query("娘"))

)

)

)

////分页

//.From(0)

//.Size(10)

////排序

//// .Sort(c => c.Field("age", SortDirection.desc))

////添加高亮

//.Highlight(h => h

//.PreTags("<b>")

//如鹏网《掌上租》项目课件 rupeng.com

//.PostTags("</b>")

//.Fields(

//f => f.FieldName("Name").Order(HighlightOrder.score)

//)

//)

.Build();

var result = client.Post(cmd, query);

var serializer = new JsonNetSerializer();

var list = serializer.ToSearchResult<Person>(result);

foreach (var doc in list.Documents)

{

Console.WriteLine(doc.Id+","+doc.Name + "," +doc.Age + "," +doc.Desc);

}

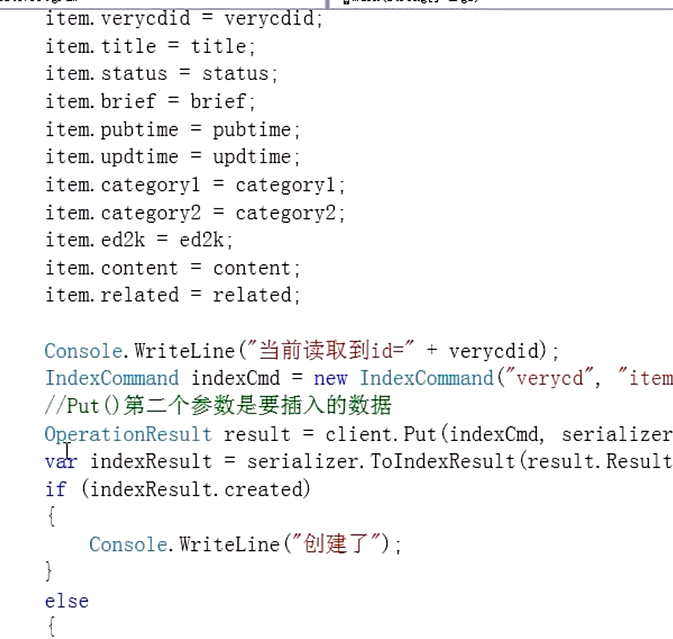
Console.ReadKey();

}

}

将SqlLite中的数据导入到ElasticSearch中，然后去搜索。





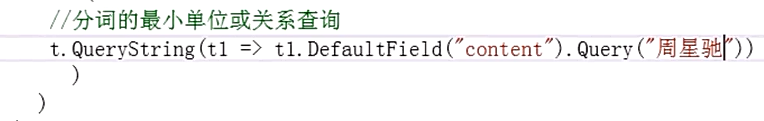
接下来把数据导入到es中。



这样一直导入，老师表示单个线程导入了7个小时。



测试，查询content中包含周星驰的数据。



两百多万条数据，感觉一两秒钟就出来了。

