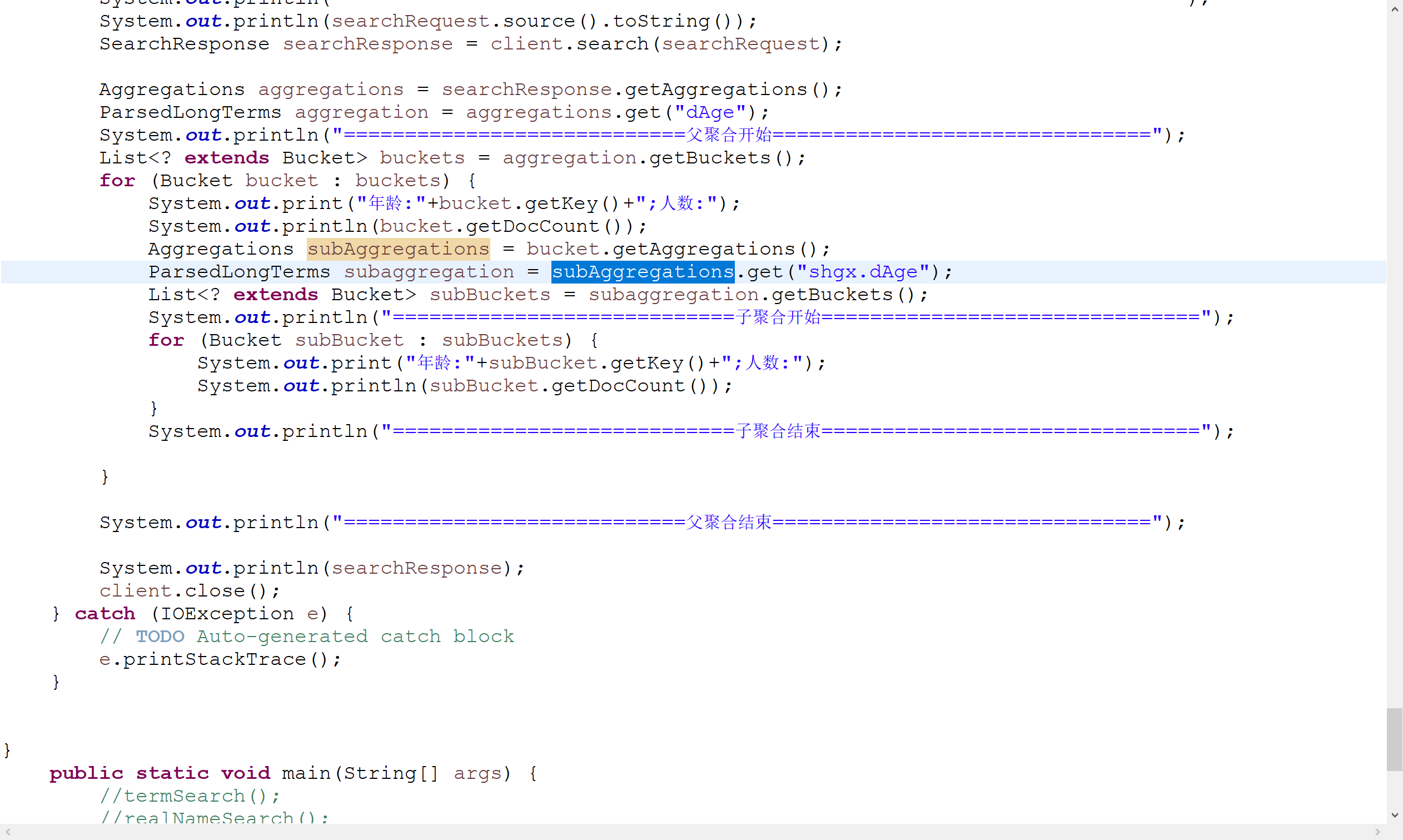
<https://blog.csdn.net/alex_xfboy/article/details/86100037>

# 包含子聚合



|  |
| --- |
| Aggregations aggregations = searchResponse.getAggregations();  ParsedLongTerms aggregation = aggregations.get("dAge");  System.out.println("===============父聚合开始=======");  List<? extends Bucket> buckets = aggregation.getBuckets();  for (Bucket bucket : buckets) {  System.out.print("年龄:"+bucket.getKey()+";人数:");  System.out.println(bucket.getDocCount());  Aggregations subAggregations = bucket.getAggregations();  ParsedLongTerms subaggregation = subAggregations.get("shgx.dAge");  List<? extends Bucket> subBuckets = subaggregation.getBuckets();  System.out.println("===========子聚合开始============");  for (Bucket subBucket : subBuckets) {  System.out.print("年龄:"+subBucket.getKey()+";人数:");  System.out.println(subBucket.getDocCount());  }  System.out.println("=======子聚合结束=============");  }    System.out.println("============父聚合结束================");  System.out.println(searchResponse);  client.close(); |

这种聚合社会关系的聚合，查询的结果是多少条，并不是指的社会关系记录有多少条，而是社会关系符合条件的用户条数。例如10个社会关系符合聚合条件，但是这10个社会关系是2个用户下的，所以返回的数据是2，而不是10.

# 统计最大值、最小值、平均值、总和

## 设置聚合条件

|  |
| --- |
| TermsAggregationBuilder aggregation = AggregationBuilders.*terms*("userName").field("userName.keyword");  aggregation.subAggregation(AggregationBuilders.*stats*("stat\_dSrje")  .field("srjl.dSrje")); |

首先根据userName进行分组，分组之后添加子聚合进行计算统计

因为在es中userName的类型是text的，所以这里要将转化成为keyword类型的进行统计。

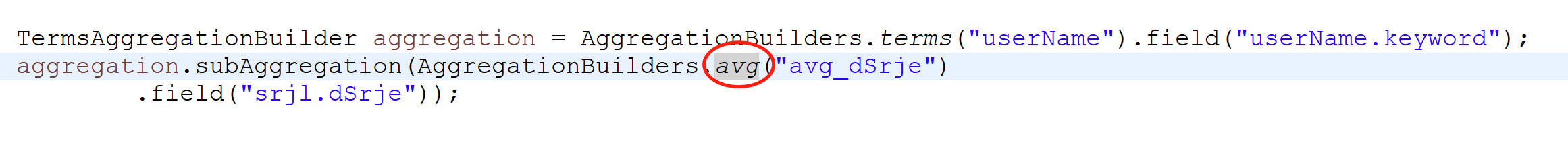
## 获取聚合结果

|  |
| --- |
| Aggregations aggregations = searchResponse.getAggregations();  Terms byCompanyAggregation = aggregations.get("userName");  List<? extends Terms.Bucket> buckets = byCompanyAggregation.getBuckets();  for (Terms.Bucket bucket : buckets) {  Stats stat = bucket.getAggregations().get("stat\_dSrje");  System.out.println("用户名："+bucket.getKey()); //获取分组名称  System.out.println("平均值："+stat.getAvg());  System.out.println("总数："+stat.getSum());  System.out.println("最大值："+stat.getMaxAsString());  System.out.println("最小值："+stat.getMin());  } |

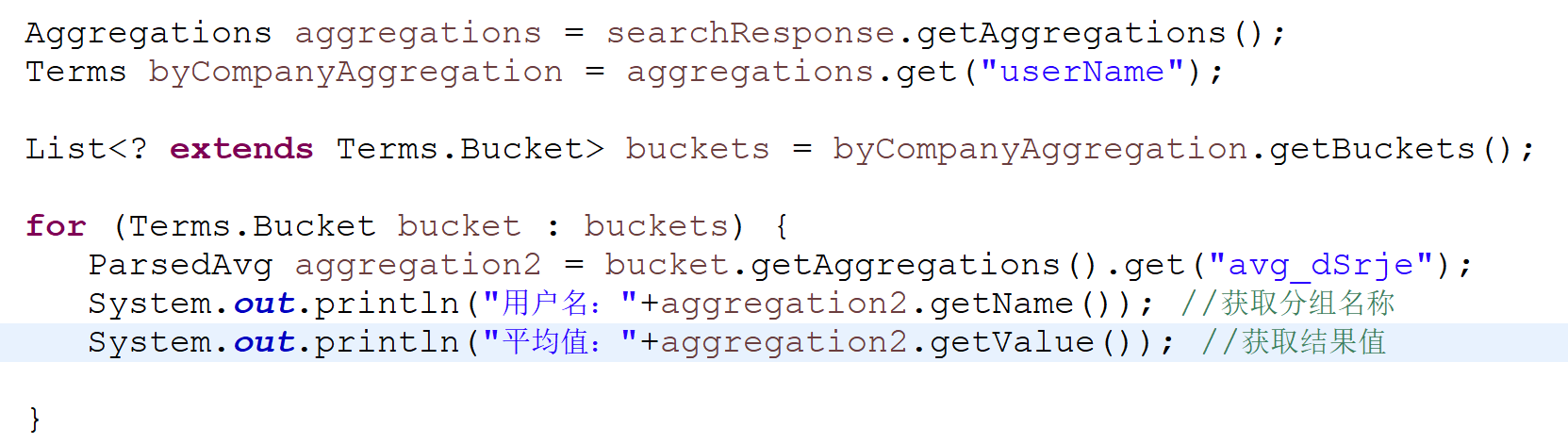
这里直接将多个结果直接都查询出来，为了以后的扩展做准备。用到的对象是Terms和Stats。

## 单求平均值

设置聚合条件

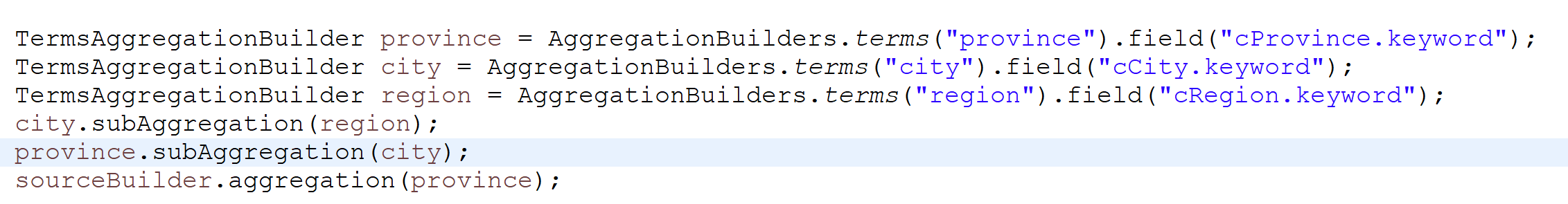


查询结果处理



# 统计所有省份下所有市，以及所有市下面所有县的用户数量

## 设置聚合条件

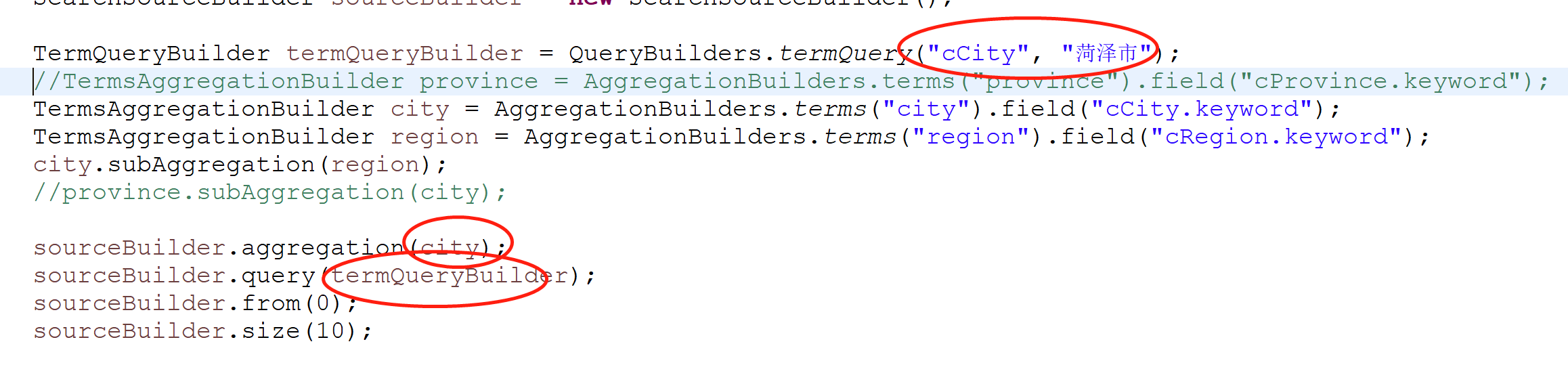


## 获取聚合结果



# 统计某个省或者某个市的用户数据

## 设置查询条件和聚合条件



## 获取结果

