# 黑马旅游网

## 前言

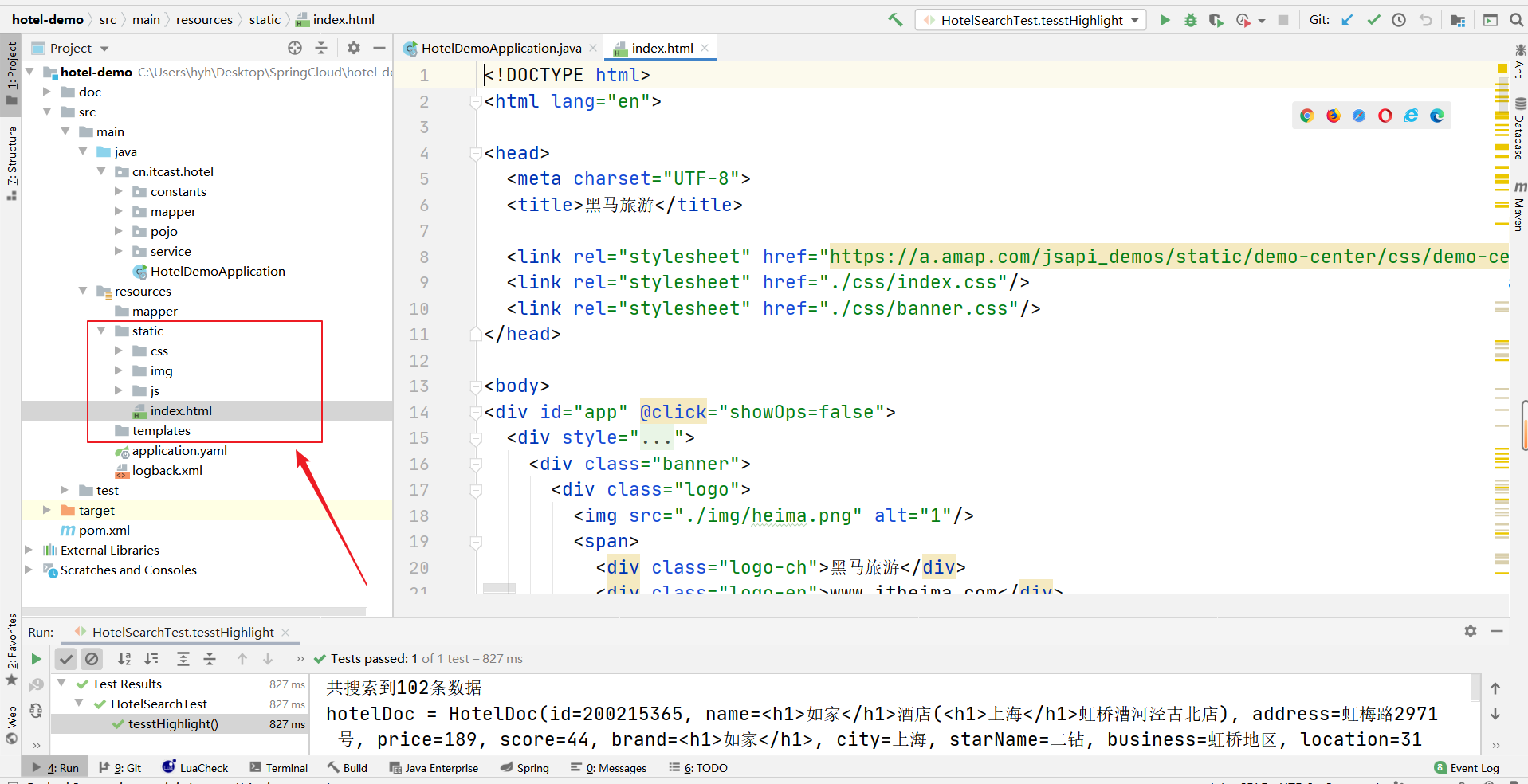
朋友们，经过前面的学习，我们已经学会了用JavaRestClient去实现数据搜索、搜索结果处理，现在是时候来验证学习成果了，我们会利用所学的知识来实现一个真实的案例：黑马旅游！

在这个案例里面实现：酒店搜索分页、酒店结果过滤、我周边的酒店、酒店竞价排名等功能！

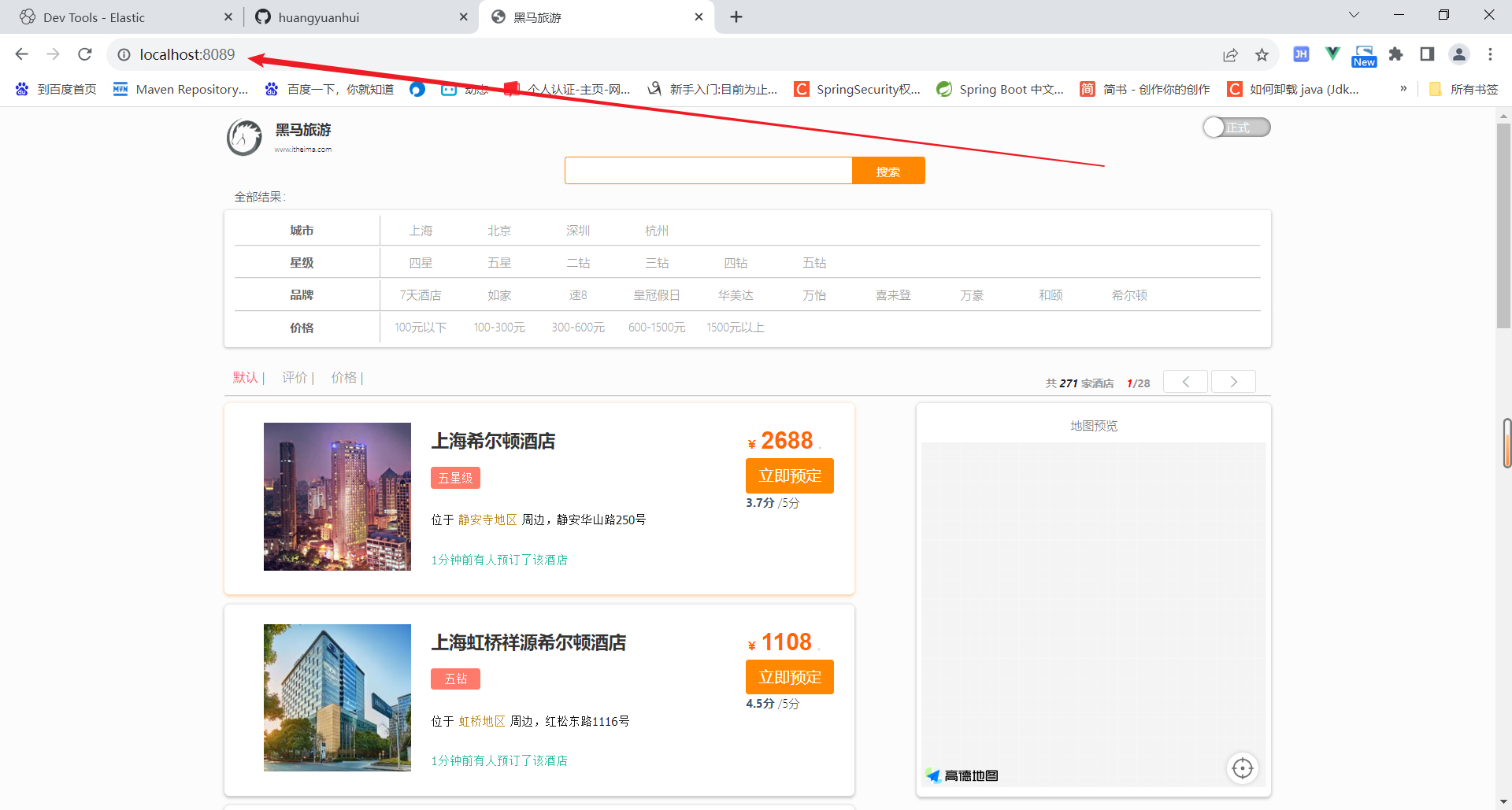
## 准备工作

### 静态文件

黑马旅游网的页面已经在hotel项目中准备好了，我们没有使用前后端分离的开发模式，前后端都在一个项目中开发：

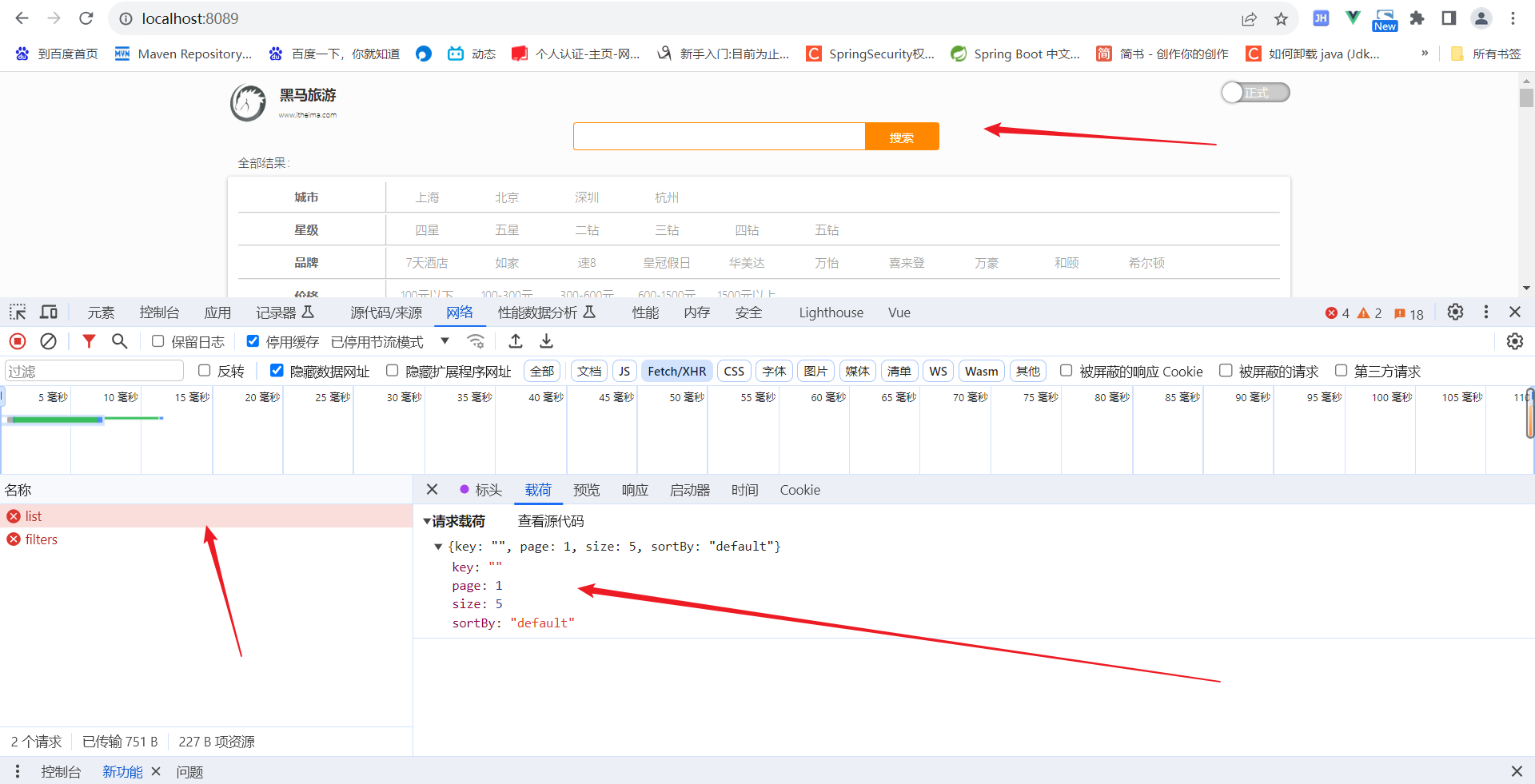


### 启动运行项目，浏览器访问：<http://localhost:8089/>

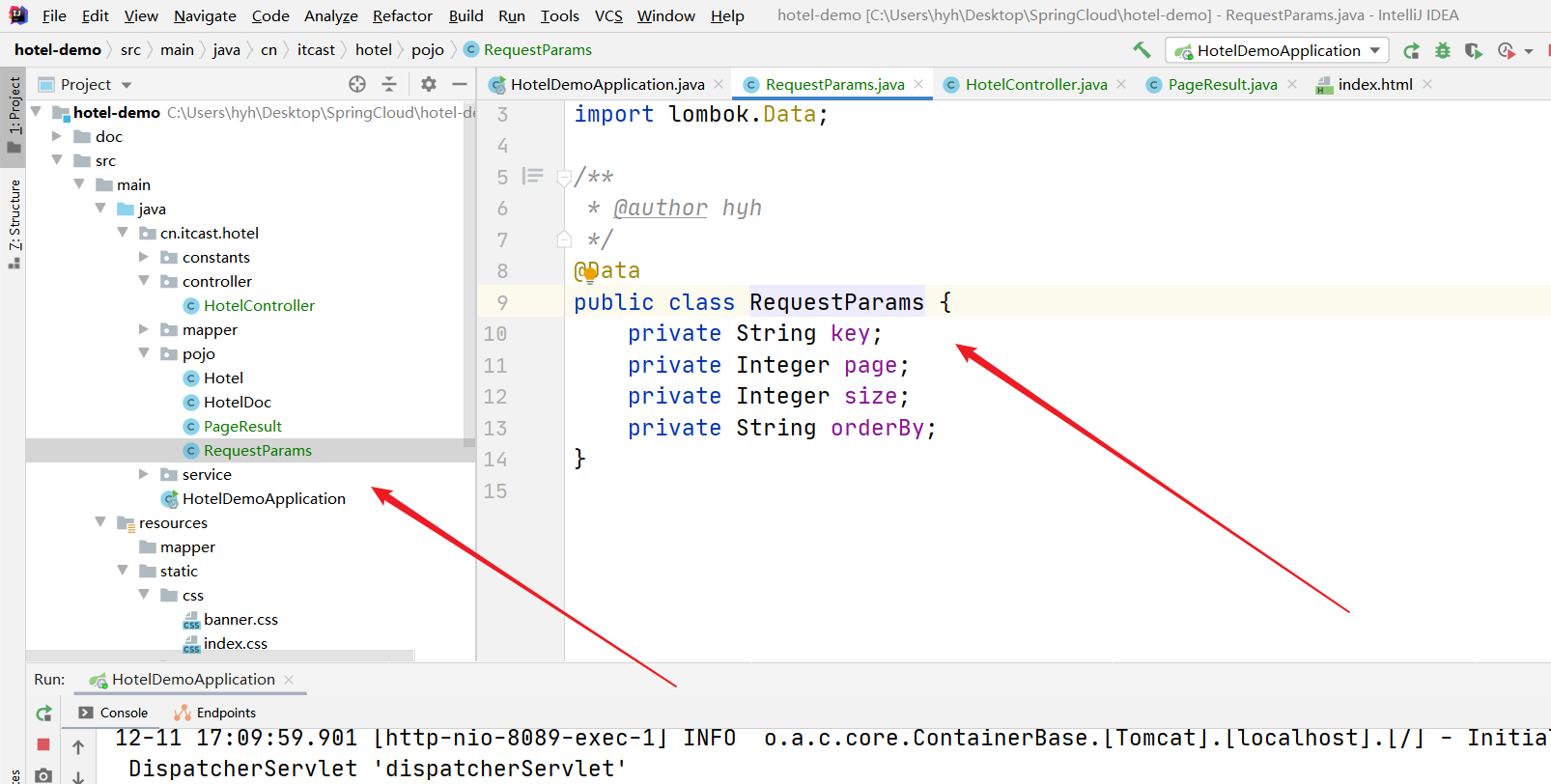


### 点击搜索按钮，可以看到发起了查询数据的请求

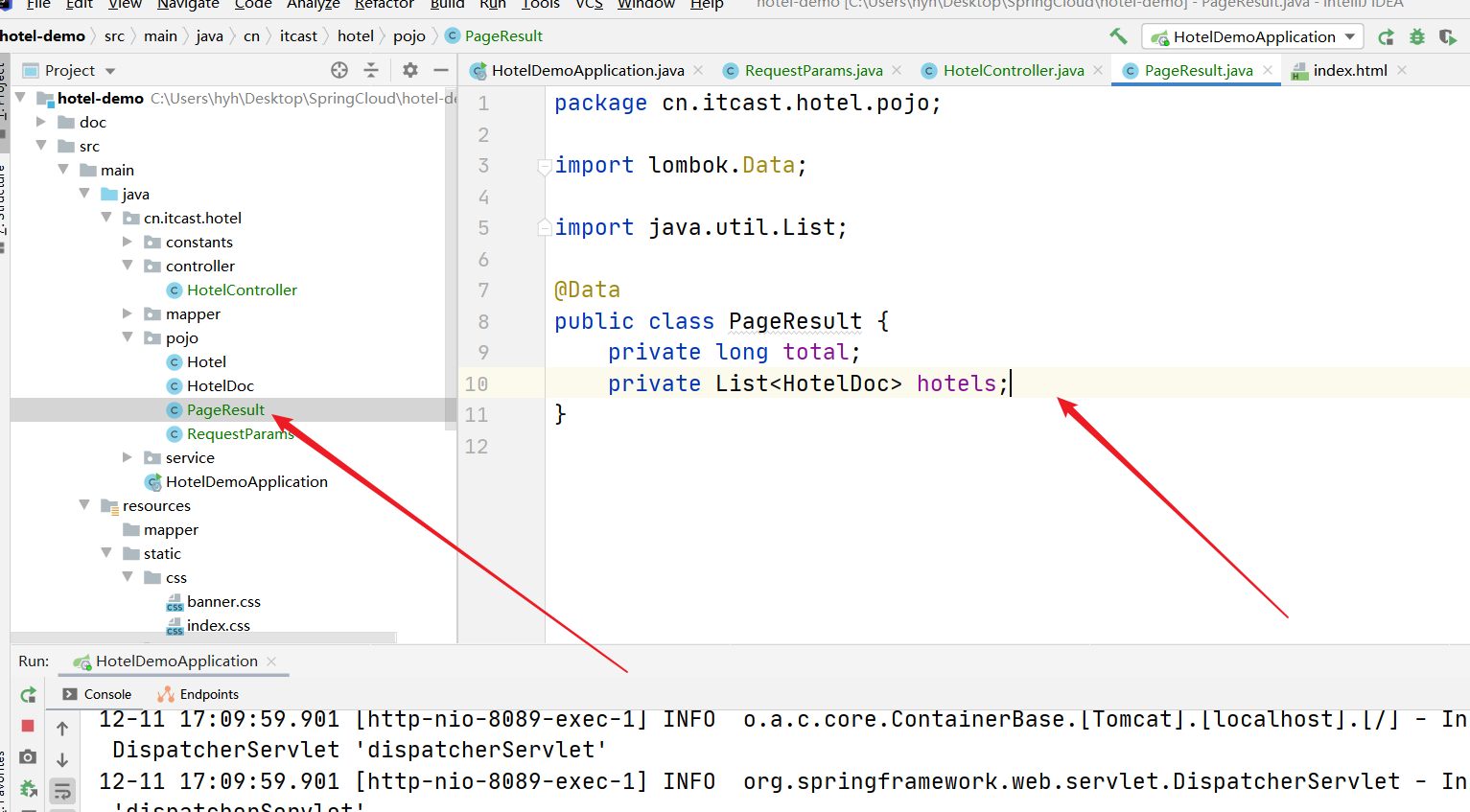
这个请求就是在查询酒店数据，注意在ES中，查询就是搜索！



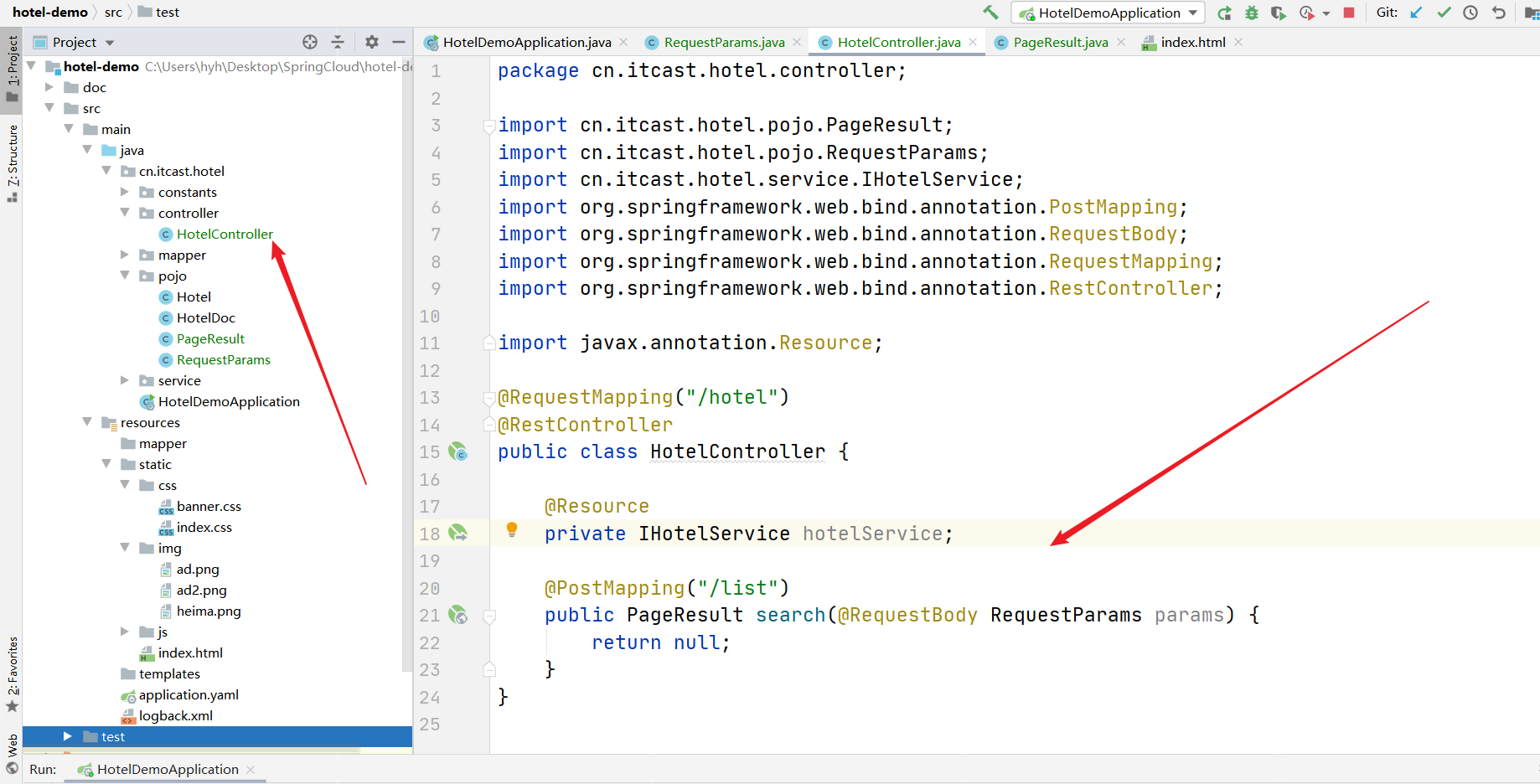
### 定义请求参数类：RequestParams



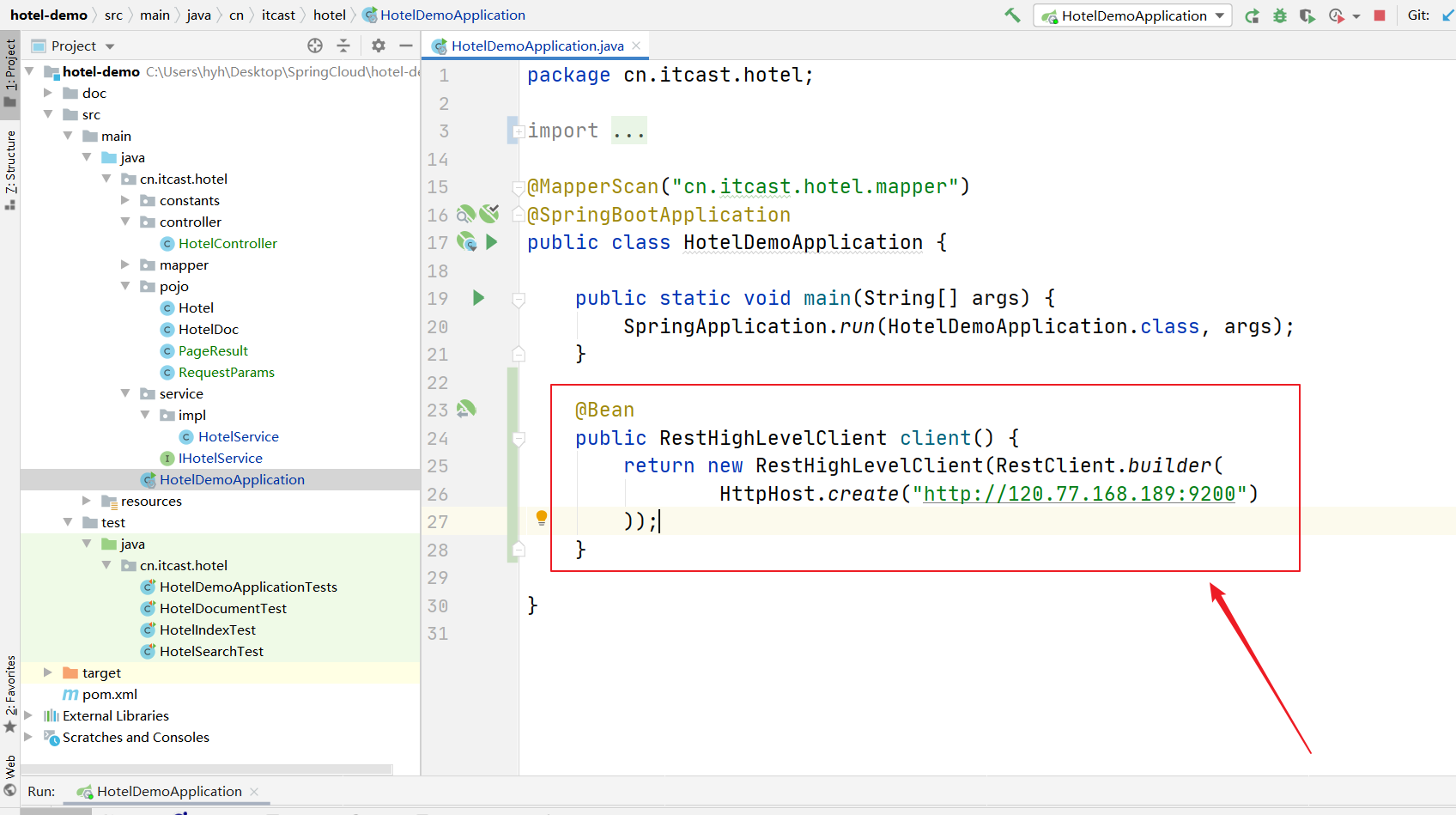
### 定义响应结果类：PageResult



### 定义controller：HotelController



### 创建RestHighLevelClient并注入Spring容器



## 酒店搜索分页（基本搜索 + 分页）

### 需求价绍：



### 代码：见hotel的git提交日志

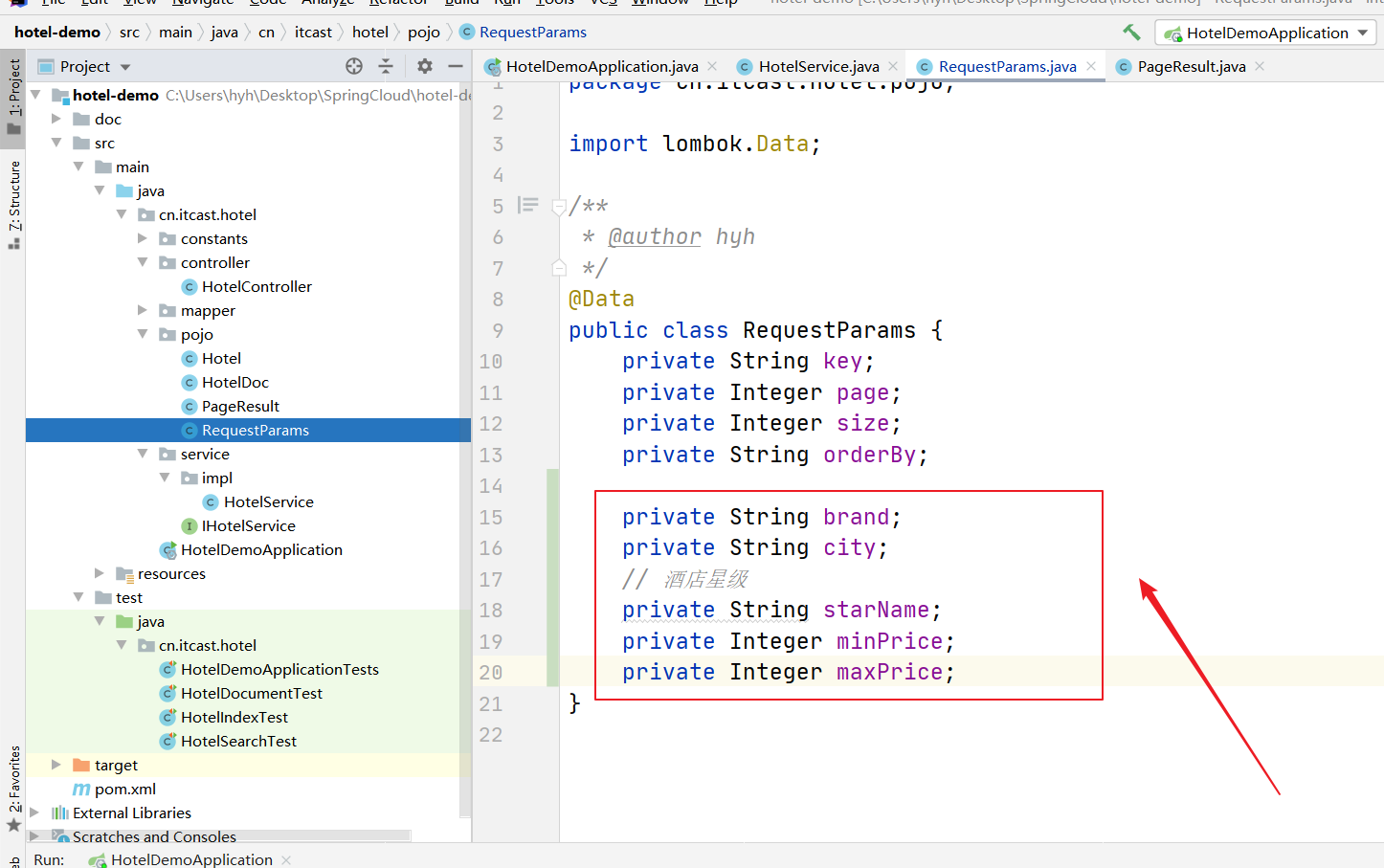
## 酒店搜索结果过滤（条件过滤）

### 需求介绍：



### 代码实现：

#### 1：修改RequestParams类，添加brand、city、sstarName、minPrice、maxPrice



#### 2：修改search方法，在match查询的基础上添加过滤条件，就是组合查询中的boolean查询了

过滤条件包括：

1：city、brand、startName都是精确匹配，即是精确查询中的词条term查询！

2：price是精确查询中的范围range查询

根据需求，这些条件之间是AND的关系，组合多条件用BooleanQuery！实际业务中要考虑业务健壮性，做好非空判断！

**核心代码：**

**/\*\***

**\* 组装查询条件**

**\***

**\* @param params**

**\* @return**

**\*/**

**private BoolQueryBuilder buildBasicQuery(RequestParams params) {**

**// 查询**

**BoolQueryBuilder boolQuery = new BoolQueryBuilder();**

**// 关键字搜索**

**String searchKey = params.getKey();**

**if (StringUtils.isEmpty(searchKey)) {**

**boolQuery.must(QueryBuilders.matchAllQuery());**

**} else {**

**boolQuery.must(QueryBuilders.matchQuery("all", searchKey));**

**}**

**// 条件过滤-城市city**

**String city = params.getCity();**

**if (!StringUtils.isEmpty(city)) {**

**boolQuery.filter(QueryBuilders.termQuery("city", city));**

**}**

**// 条件过滤-品牌brand**

**String brand = params.getBrand();**

**if (!StringUtils.isEmpty(brand)) {**

**boolQuery.filter(QueryBuilders.termQuery("brand", brand));**

**}**

**// 条件过滤-星级starName**

**String starName = params.getStarName();**

**if (!StringUtils.isEmpty(starName)) {**

**boolQuery.filter(QueryBuilders.termQuery("starName", starName));**

**}**

**// 条件过滤-价格price**

**if (params.getMinPrice() != null && params.getMaxPrice() != null) {**

**int minPrice = params.getMinPrice();**

**int maxPrice = params.getMaxPrice();**

**boolQuery.filter(QueryBuilders**

**.rangeQuery("price").gte(minPrice).lte(maxPrice));**

**}**

**return boolQuery;**

**}**

### 注意点（很重要很重要很重要）：

布尔查询BooleanQuery中must、should是主菜，filter和must\_not是调料（甚至must\_not几乎不用，因为filter和must\_not是一体两面）。意思就是must先搜索出结果，然后filter、must\_not再对must搜索出来的结果在做一次筛选！

以上代码的例子就是这样啊！显示must中全文检索all字段，再filter筛选出符合条件的结果！

## 我周边的酒店（排序中的地理坐标排序\_geo\_distance）

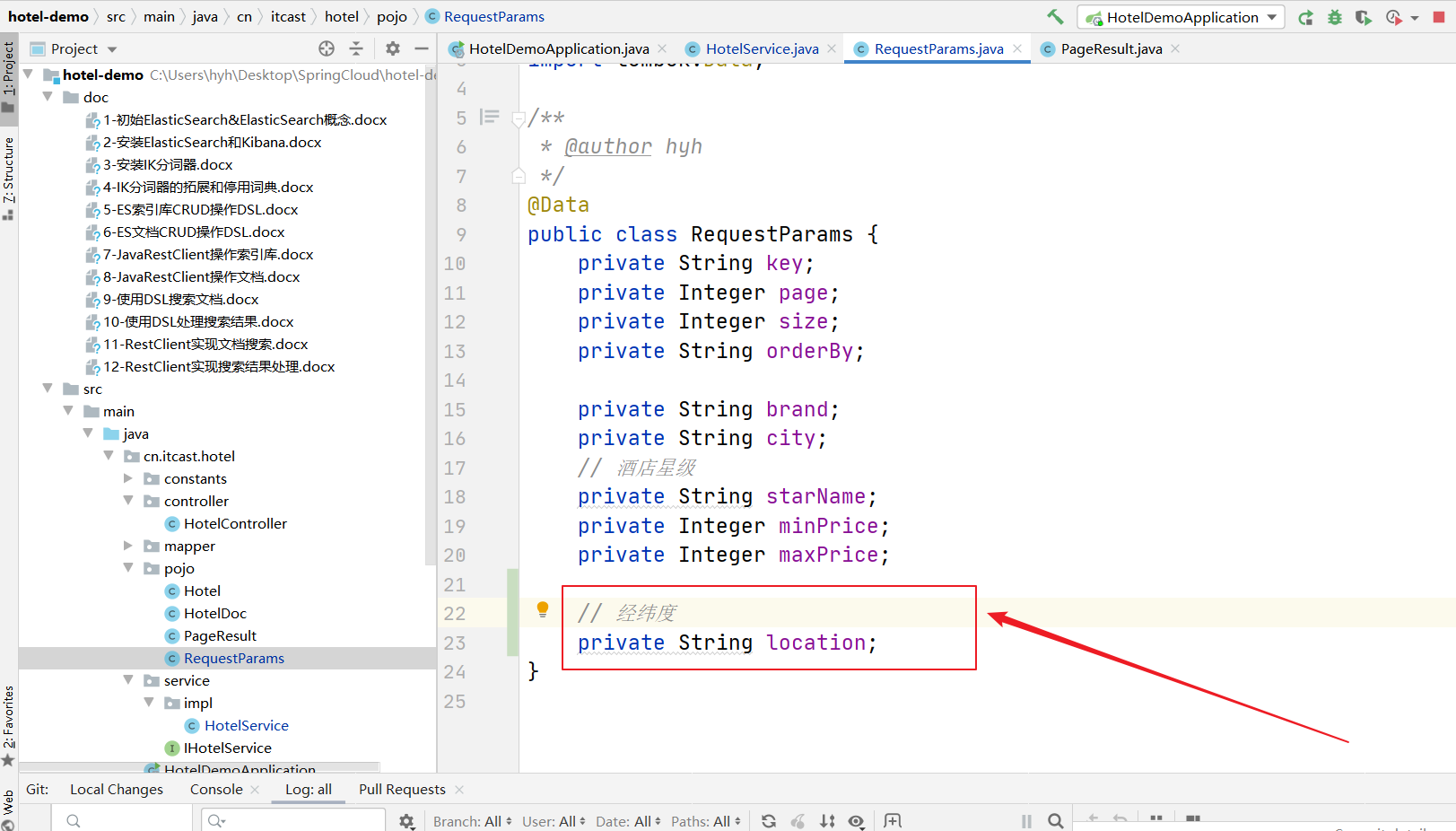
### 需求介绍

前端页面顶级定位后，会将你所在的位置发送给后台



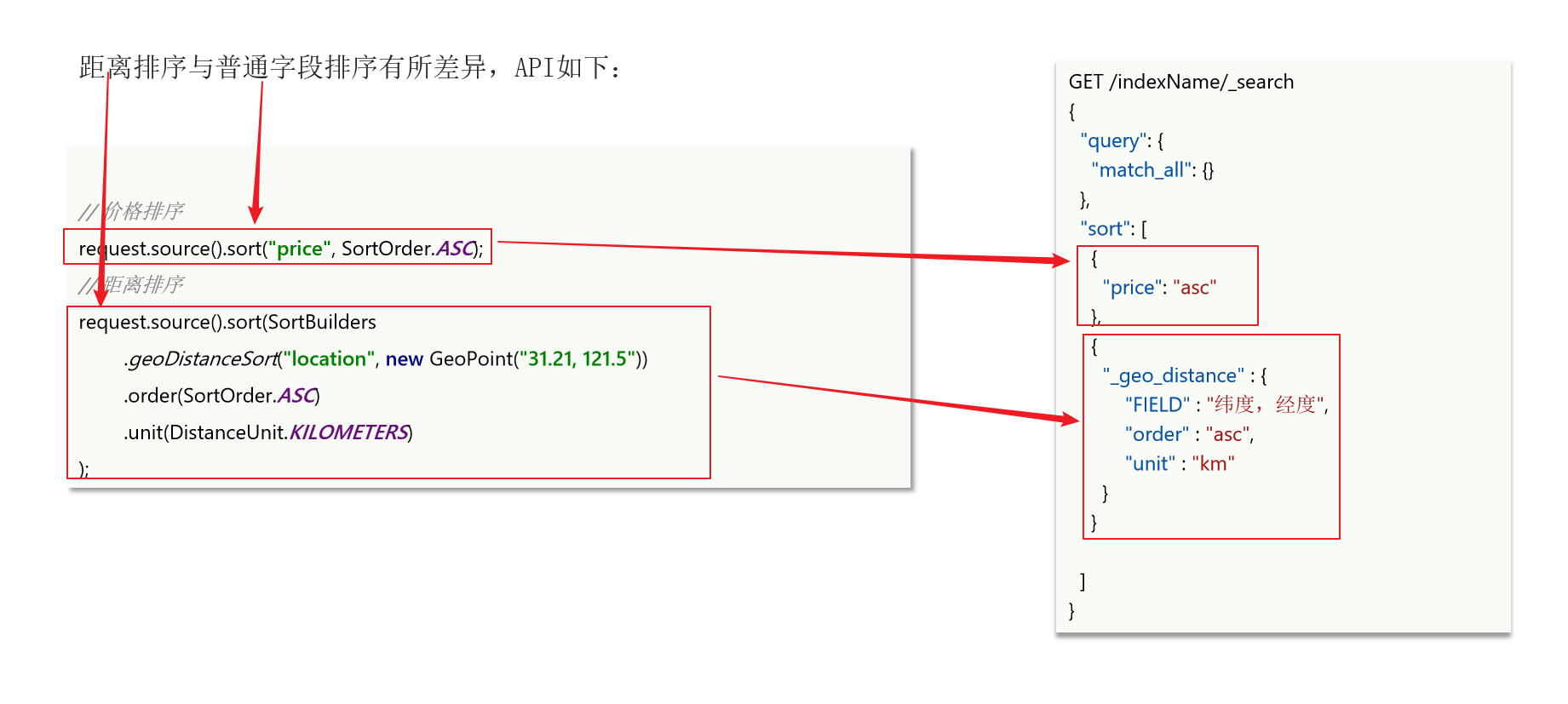
### 代码实现

#### 1：修改RequestParams参数，接收location字段！

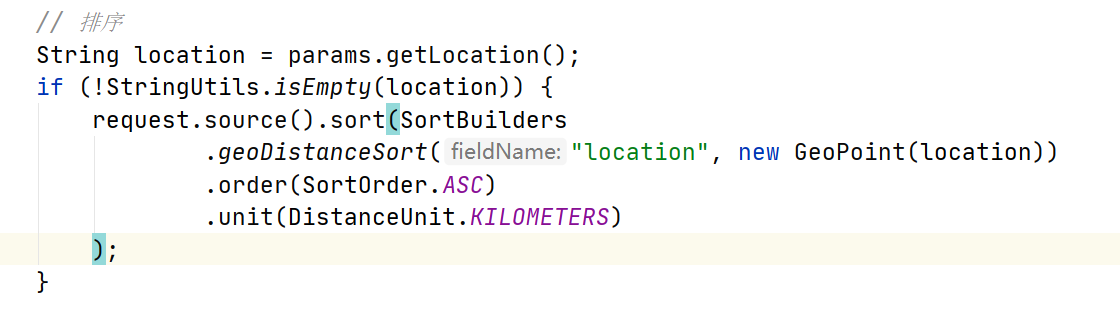


#### 2：修改search业务逻辑，如果location优质，添加根据\_geo\_distance距离排序的功能

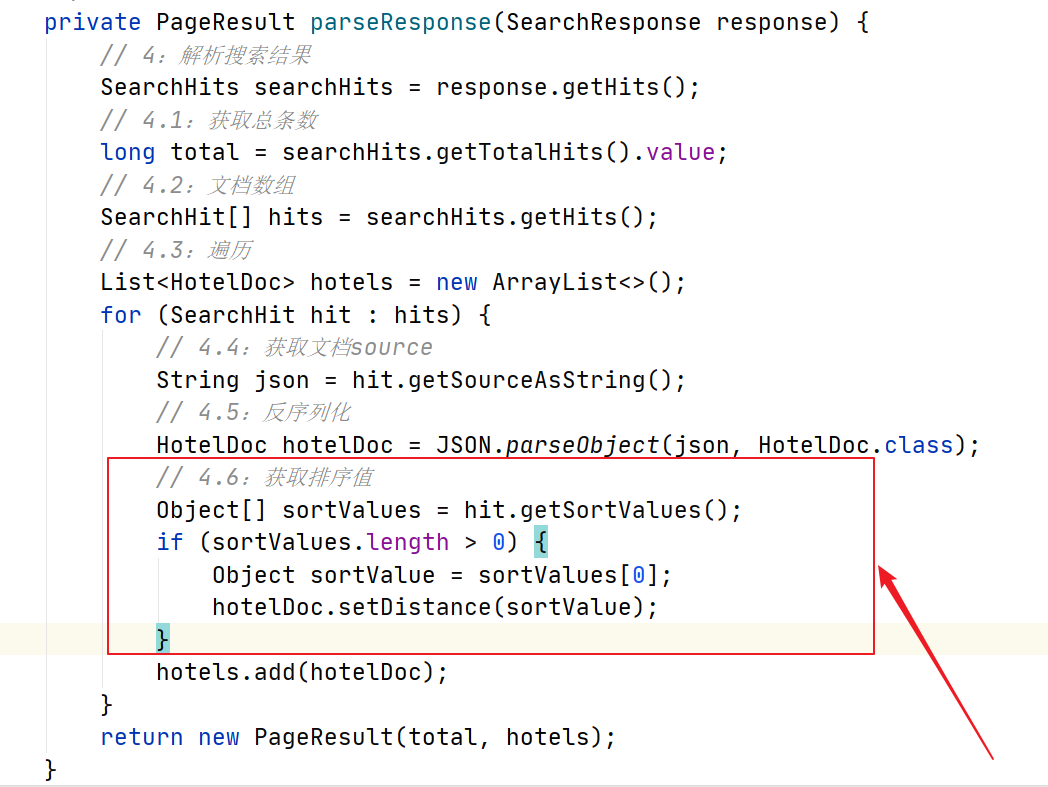
距离排序与普通字段排序有所差异，API如下：



地理坐标排序核心代码如下：



#### 3：修改结果解析逻辑，从搜索结果解析出sort，拿到排序值！



## 酒店竞价排名-广告置顶（function score）

我们知道，在默认请情况下，酒店的搜索结果是按照相关性排名来去排序的，也就是说跟搜索关键字相关性越高，排名越靠前，如果要让指定的酒店排在前面，那么就要去影响相关性打分，就要用到function\_score查询了.。

问题是我们要给给哪些酒店去增加打分呢？我们可以给要置顶的酒店文档添加一个标记，然后利用function score给带有标记的文档增加权重！



### 1：挑选出几个你喜欢的酒店，给这几个酒店的文档数据添加isAD字段，值为true。

这里是我们自己随便挑几个，将来再真实开发中，肯定会有一个后台管理系统，将来谁掏了钱，我们就在后台管理系统中给他标一个true就行了

**POST /hotel/\_update/2056126831**

**{**

**"doc": {**

**"isAD": true**

**}**

**}**

**POST /hotel/\_update/1989806195**

**{**

**"doc": {**

**"isAD": true**

**}**

**}**

**POST /hotel/\_update/2056105938**

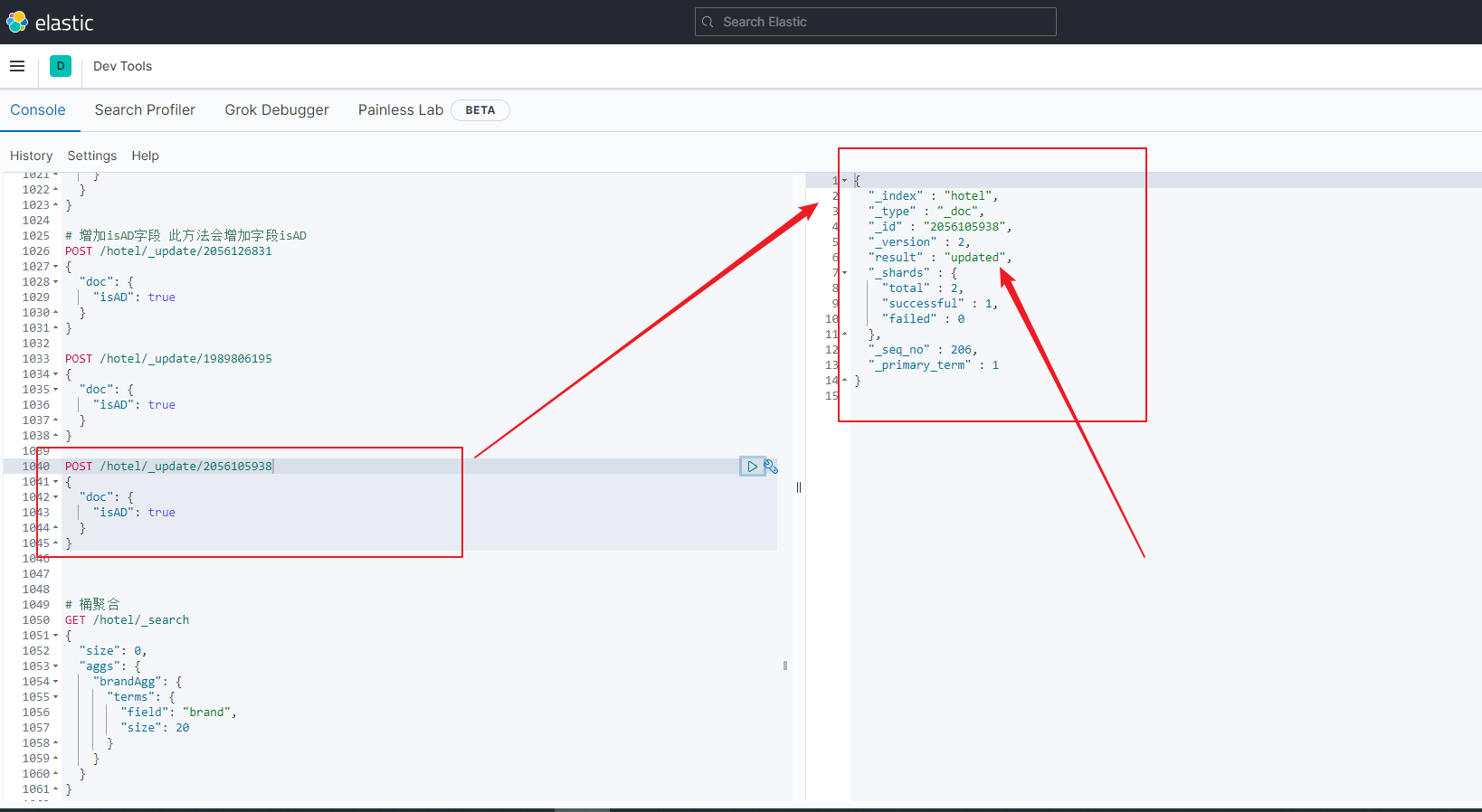
**{**

**"doc": {**

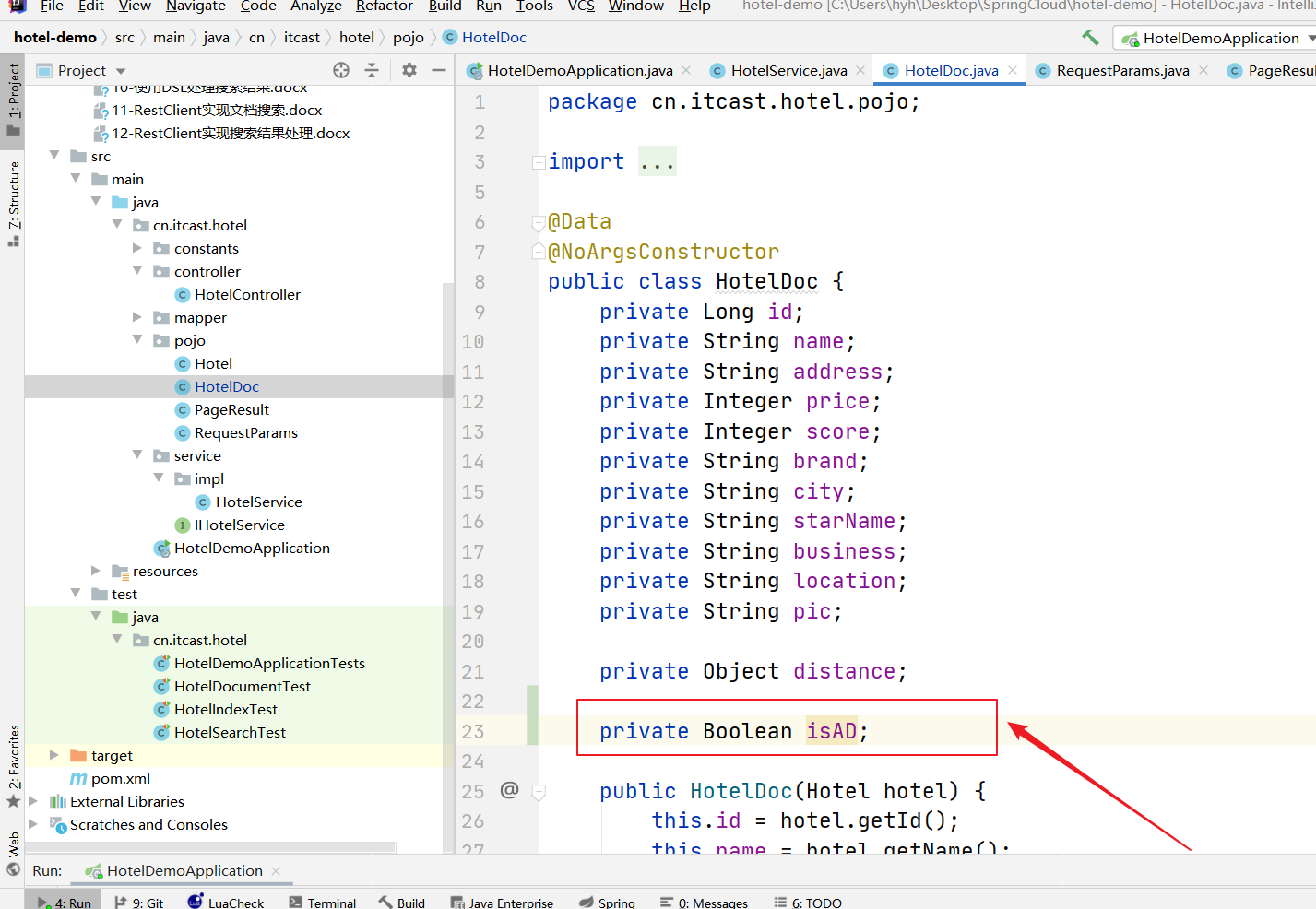
**"isAD": true**

**}**

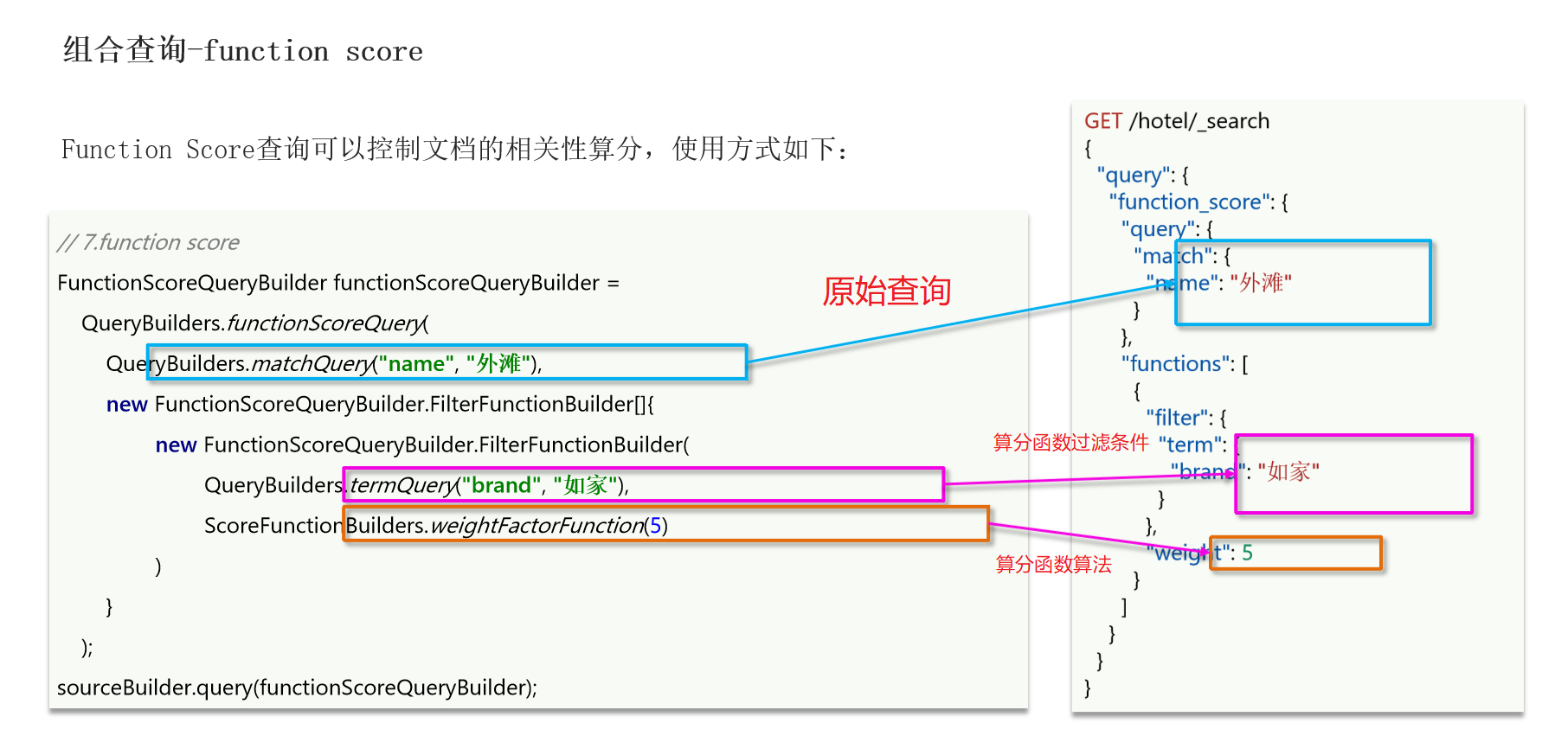
**}**

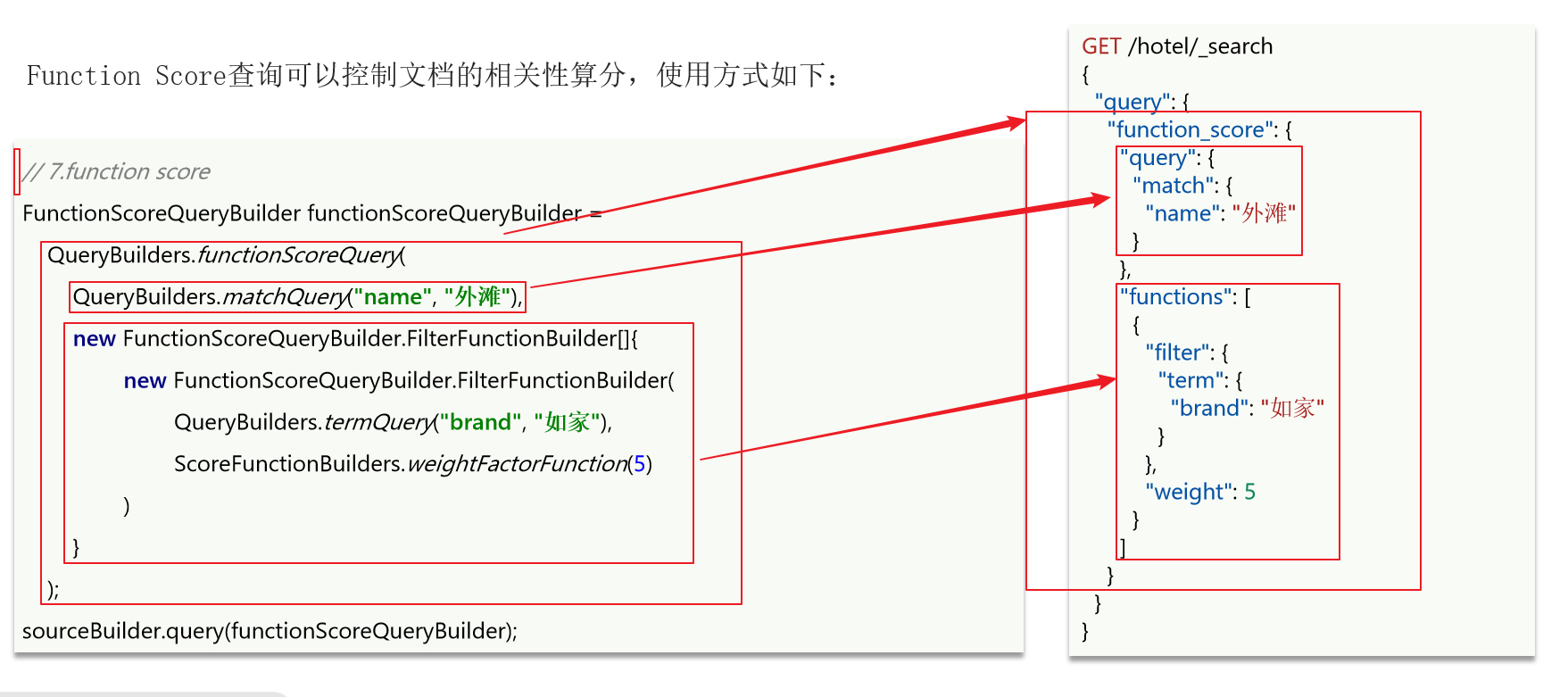


### 2：给HotelDoc类添加Boolean类型的isAD字段



### 3：修改search方法，添加function score功能，给isAD值为true的酒店增加权重!





4：算分控制核心代码如下：



### 5：浏览器查看结果

