ElasticSearch 高级

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

目录 Contents

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

目标
TARGET

- 【应用】 ElasticSearch-bulk批量操作
- 【应用】 ElasticSearch -模糊查询
- 【应用】 ElasticSearch -**queryString**查询
- 【应用】 ElasticSearch-布尔查询
- 【应用】 ElasticSearch-聚合查询
- 【应用】 ElasticSearch-高亮查询
- 【应用】 ElasticSearch 集群管理
- 【掌握】 ElasticSearch-路由原理
- 【掌握】 ElasticSearch-脑裂原因及解决办法

1.今日内容

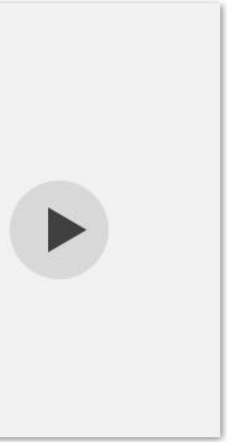
黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

今日内容

掌握程度：了解

要点提示

- 了解今日课程内容



2. ElasticSearch高级操作-bulk批量操作-脚本

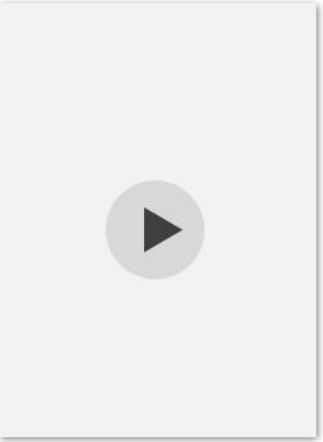
黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

bulk批量操作-脚本

掌握程度：应用

要点提示

◆ 执行bulk批量操作脚本



2. ElasticSearch高级操作-bulk批量操作-脚本

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

知识总结

■ 批量操作-脚本

Bulk 批量操作是将文档的增删改查一些列操作，通过一次请求全都做完。减少网络传输次数。

语法：

```
POST /_bulk
{"action": {"metadata"}}
{"data"}
```

示例：

```
POST _bulk
{"delete":{ "_index":"person", "_id":"5" }}
{"create":{ "_index":"person", "_id":"5" }}
{"name":"六号","age":20,"address":"北京"}
{"update":{ "_index":"person", "_id":"2" }}
{"doc":{"name":"二号"}}
```

3.ElasticSearch高级操作-bulk批量操作-JavaAPI

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

bulk批量操作-JavaAPI

掌握程度：应用

要点提示

- 使用javaAPI执行bulk 批量操作

3.ElasticSearch高级操作-bulk批量操作-JavaAPI

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

知识总结

- bulk批量操作

需求：

使用javaAPI执行bulk 批量操作

实现步骤：

① 删除5号记录
② 添加6号记录
③ 修改3号记录 名称为 “三号”

```
@Test
public void test2() throws IOException {
    //创建bulkrequest对象，整合所有操作
    BulkRequest bulkRequest =new BulkRequest();
    //1. 删除5号记录
    DeleteRequest deleteRequest=new DeleteRequest("person1","5");
    bulkRequest.add(deleteRequest);
    //2. 添加6号记录
    Map<String, Object> map=new HashMap<>();
    map.put("name","六号");
    IndexRequest indexRequest=new IndexRequest("person1").id("6").source(map);
    bulkRequest.add(indexRequest);
    //3. 修改3号记录 名称为 “三号”
    Map<String, Object> mapUpdate=new HashMap<>();
    mapUpdate.put("name","三号");
    UpdateRequest updateRequest=new UpdateRequest("person1","3").doc(mapUpdate);
    bulkRequest.add(updateRequest);
    //执行批量操作
    BulkResponse response = client.bulk(bulkRequest, RequestOptions.DEFAULT);
    System.out.println(response.status());

}
```

4.ElasticSearch高级操作-导入数据-分析&创建索引

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

导入数据-分析&创建索引

掌握程度：应用

要点提示

◆ 创建索引(goods)

4.ElasticSearch高级操作-导入数据-分析&创建索引

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

知识总结

创建索引

参见数据准备.md

- title:商品标题
- price:商品价格
- createTime:创建时间
- categoryName:分类名称。如：家电，手机
- brandName:品牌名称。如：华为，小米
- spec: 商品规格。如： spec:{"屏幕尺寸","5寸"， "内存大小","128G"}
- saleNum:销量
- stock:库存量

```
PUT goods
{
    "mappings": {
        "properties": {
            "title": {
                "type": "text",
                "analyzer": "ik_smart"
            },
            "price": {
                "type": "double"
            },
            "createTime": {
                "type": "date"
            },
            "categoryName": {
                "type": "keyword"
            },
            "brandName": {
                "type": "keyword"
            },
            "spec": {
                "type": "object"
            },
            "saleNum": {
                "type": "integer"
            },
            "stock": {
                "type": "integer"
            }
        }
    }
}
```

5-ElasticSearch高级操作-导入数据-代码实现

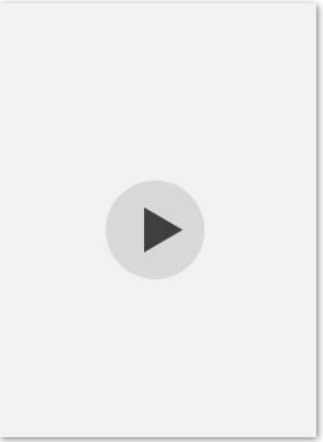
黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

导入数据-代码实现

掌握程度：应用

要点提示

- 使用javaAPI将数据从Myql导入ES



5-ElasticSearch高级操作-导入数据-代码实现

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

知识总结

导入数据

案例：需求

将数据库中Goods表的数据导入到ElasticSearch中

案例：实现步骤

① 创建goods索引
② 查询Goods表数据
③ 批量添加到ElasticSearch中

```
@Test
public void test3() throws IOException {
    //1.查询所有数据，mysql
    List<Goods> goodsList = goodsMapper.findAll();
    //2.bulk导入
    BulkRequest bulkRequest=new BulkRequest();
    //2.1 循环goodsList，创建IndexRequest添加数据
    for (Goods goods : goodsList) {
        //2.2 设置spec规格信息 Map的数据   specStr:{}
        String specStr = goods.getSpecStr();
        //将json格式字符串转为Map集合
        Map map = JSON.parseObject(specStr, Map.class);
        //设置spec map
        goods.setSpec(map);
        //将goods对象转换为json字符串
        String data = JSON.toJSONString(goods);
        IndexRequest indexRequest=new
IndexRequest("goods").source(data,XContentType.JSON);
        bulkRequest.add(indexRequest);
    }
    BulkResponse response = client.bulk(bulkRequest, RequestOptions.DEFAULT);
    System.out.println(response.status());
}
```

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

6.ElasticSearch高级操作-导入数据-代码实现-详解（选放）

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

代码实现-详解（选放）

掌握程度：理解

要点提示

- 了解spec字段为什么要转换为JSON

6.ElasticSearch高级操作-导入数据-代码实现-详解（选放）

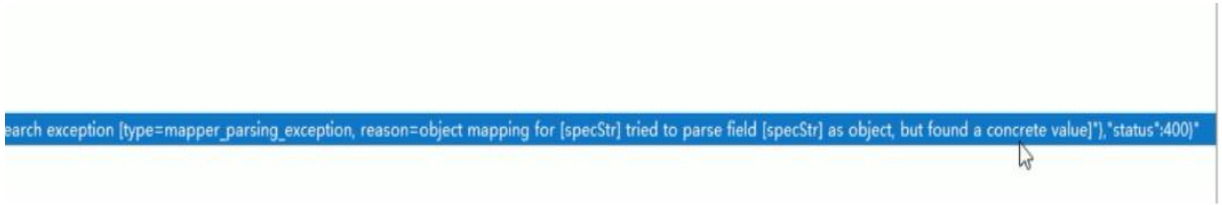
黑马程序员 www.itheima.com 传智播客旗下高端IT教育品牌

知识总结

- **转换成JSON的原因**：

```
#spec配置的数据类型是JSON对象，所以当存放字符串的时候报错
            "spec": {
                "type": "object"
            },
```

错误信息：



7.ElasticSearch查询-matchAll-脚本

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

matchAll-脚本

掌握程度：应用

要点提示

- 使用matchAll命令查询文档
- 使用分页命令实现分页

7.ElasticSearch查询-matchAll-脚本

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

知识总结

- matchAll-脚本

```
# 默认情况下，es一次展示10条数据,通过from和size来控制分页
# 查询结果详解
GET goods/_search
{
  "query": {
    "match_all": {}
  },
  "from": 0,
  "size": 100
}

GET goods
```

8.ElasticSearch查询-matchAll-JavaAPI

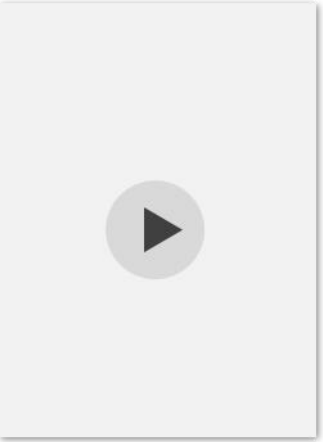
黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

matchAll-JavaAPI

掌握程度：应用

要点提示

- 使用JavaAPI实现matchAll查询，并
- 使用JavaAPI实现分页控制



8.ElasticSearch查询-matchAll-JavaAPI

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

知识总结

- matchAll-JavaAPI

案例：需求

使用JavaAPI实现matchAll查询，并
使用JavaAPI实现分页控制

案例：实现步骤

① 创建goods索引
② 查询Goods表数据
③ 批量添加到ElasticSearch中

```
@Test
public void matchAll() throws IOException {
        //2. 构建查询请求对象，指定查询的索引名称
        SearchRequest searchRequest=new SearchRequest("goods");
        //4. 创建查询条件构建器SearchSourceBuilder
        SearchSourceBuilder sourceBuilder=new SearchSourceBuilder();
        //6. 查询条件
        QueryBuilder queryBuilder= QueryBuilders.matchAllQuery();
        //5. 指定查询条件
        sourceBuilder.query(queryBuilder);
        //3. 添加查询条件构建器 SearchSourceBuilder
        searchRequest.source(sourceBuilder);
        // 8 . 添加分页信息  不设置 默认10条
        // sourceBuilder.from(0);
        // sourceBuilder.size(100);
        //1. 查询,获取查询结果
        SearchResponse searchResponse = client.search(searchRequest,
RequestOptions.DEFAULT);
        //7. 获取命中对象 SearchHits
        SearchHits hits = searchResponse.getHits();
        //7.1 获取总记录数
      Long total= hits.getTotalHits().value;
        System.out.println("总数: "+total);
        //7.2 获取Hits数据  数组
        SearchHit[] hits1 = hits.getHits();
            //获取json字符串格式的数据
        List<Goods> goodsList = new ArrayList<>();
        for (SearchHit searchHit : hits1) {
            String sourceAsString = searchHit.getSourceAsString();
            //转为java对象
            Goods goods = JSON.parseObject(sourceAsString, Goods.class);
            goodsList.add(goods);
        }
        for (Goods goods : goodsList) {
            System.out.println(goods);
        }
    }
```

9.ElasticSearch查询-termQuery

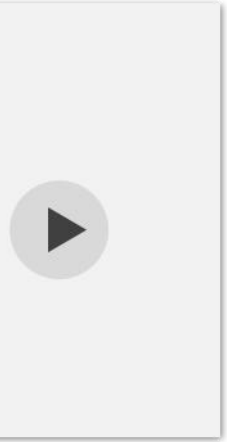
黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

termQuery

掌握程度：掌握

要点提示

- 掌握termQuery的特性



视
结

9.ElasticSearch查询-termQuery

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

知识总结

term查询-脚本

term查询：不会对查询条件进行分词。

语法

```
GET 索引名称/_search
{
  "query": {
    "term": {
      "字段名称": {
        "value": "查询条件"
      }
    }
  }
}
```

注意：term query会去倒排索引中寻找确切的term，它并不知道分词器的存在。这种查询适合**keyword** 、**numeric**、**date**

10.ElasticSearch查询-matchQuery

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

matchQuery

掌握程度：掌握

要点提示

- 掌握matchQuery

视
结

10.ElasticSearch查询-matchQuery

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

知识总结

match查询-脚本

match查询：

- 会对查询条件进行分词。
- 然后将分词后的查询条件和词条进行等值匹配
- 默认取并集（OR）
- match query知道分词器的存在。并且理解是如何被分词的

语法

```
GET 索引名称/_search
{
  "query": {
    "match": {
      "字段名称": "查询条件"
    }
  }
}
```

```
GET 索引名称/_search
{
  "query": {
    "match": {
      "字段名称": {
        "query": "查询条件",
        "operator": "操作（or或and）"
      }
    }
  }
}
```

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

11.ElasticSearch查询-模糊查询-脚本

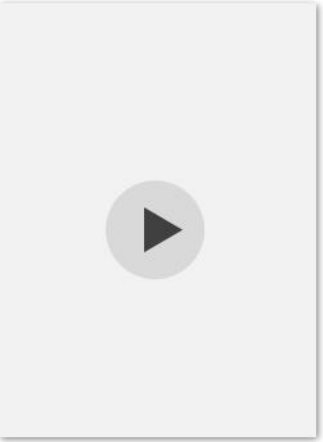
黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

模糊查询-脚本

掌握程度：掌握

要点提示

- 掌握wildcard查询
- 掌握正则查询
- 掌握前缀查询



11.ElasticSearch查询-模糊查询-脚本

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

知识总结

模糊查询-脚本

wildcard查询：

会对查询条件进行分词。还可以使用通配符？（任意单个字符） 和 * （0个或多个字符）

```
"*华*"  包含华字的
"华*"   华字后边多个字符
"华?"  华字后边多个字符
"*华"或"?华" 会引发全表（全索引）扫描 注意效率问题
```

选择语言

```
# wildcard 查询。查询条件分词，模糊查询
GET goods/_search
{
  "query": {
    "wildcard": {
      "title": {
        "value": "华*"
      }
    }
  }
}
```

11.ElasticSearch查询-模糊查询-脚本

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

知识总结

模糊查询-脚本

regexp查询：正则查询

\W：匹配包括下划线的任何单词字符，等价于 [A-Z a-z 0-9_]　开头的反斜杠是转义符

+号多次出现

(.)*为任意字符
正则查询取决于正则表达式的效率

```
GET goods/_search
{
  "query": {
    "regexp": {
      "title": "\\w+(.)*"
    }
  }
}
```

前缀查询： 对keyword类型支持比较好

```
# 前缀查询 对keyword类型支持比较好
GET goods/_search
{
  "query": {
    "prefix": {
      "brandName": {
        "value": "三"
      }
    }
  }
}
```

12.ElasticSearch查询-模糊查询-JavaAPI

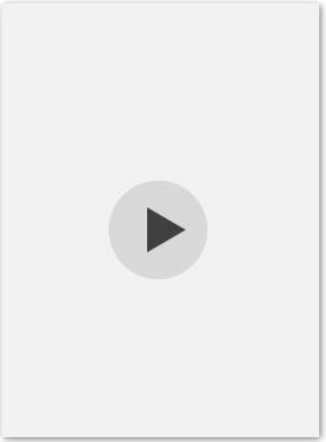
黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

模糊查询-JavaAPI

掌握程度：应用

要点提示

- 使用javaAPI完成模糊查询



12.ElasticSearch查询-模糊查询-JavaAPI

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

知识总结

模糊查询-JavaAPI

```
//模糊查询
WildcardQueryBuilder query = QueryBuilders.wildcardQuery("title", "华*");//华后多个字符
//正则查询
 RegexpQueryBuilder query = QueryBuilders.regexpQuery("title", "\\w+(.)*");
 //前缀查询
 PrefixQueryBuilder query = QueryBuilders.prefixQuery("brandName", "三");
```

视
结

13-ElasticSearch查询-范围&排序查询

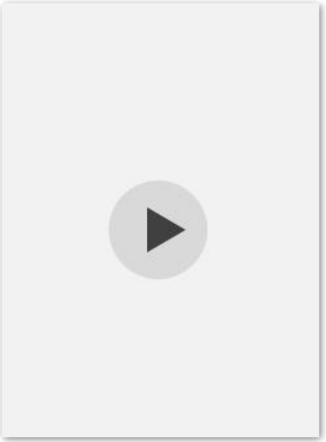
黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

范围&排序查询

掌握程度：理解

要点提示

- 掌握范围查询、排序查询-脚本
- 使用JavaAPI完成范围查询、排序查询



13-ElasticSearch查询-范围&排序查询

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

知识总结

■ 范围&排序查询

范围查询：查找指定字段在指定范围内包含值：gte大于,lte 小于

排序查询：desc 降序，asc升序

```
GET goods/_search
{
  "query": {
    "range": {
      "price": {
        "gte": 2000,
        "lte": 3000
      }
    }
  },
  "sort": [
    {
      "price": {
        "order": "desc"
      }
    }
  ]
}
```

代码：

```
 //范围查询 以price 价格为条件
RangeQueryBuilder query = QueryBuilders.rangeQuery("price");
//指定下限
query.gte(2000);
//指定上限
query.lte(3000);
sourceBuilder.query(query);
//排序  价格 降序排列
sourceBuilder.sort("price",SortOrder.DESC);
```

14.ElasticSearch查询-queryString查询

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

queryString查询

掌握程度：掌握

要点提示

- 掌握queryString查询
- 掌握simple_query_string查询

14.ElasticSearch查询-queryString查询

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

知识总结

queryString查询-脚本

queryString：

- 会对查询条件进行分词。
- 然后将分词后的查询条件和词条进行等值匹配
- 默认取并集（OR）
- 可以指定多个查询字段
- query_string：识别query中的连接符（or 、and）

```
GET 索引名称/_search
{
  "query": {
    "query_string": {
      "fields": ["字段1","字段2"...],
      "query": "查询条件1 OR 查询条件2"
    }
  }
}
```

java代码

```
QueryStringQueryBuilder query = QueryBuilders.queryStringQuery("华为手机").field("title").field("categoryName")
.field("brandName").defaultOperator(Operator.AND);
```

14.ElasticSearch查询-queryString查询

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

知识总结

simple_query_string查询-脚本

simple_query_string和query_string的区别：

- 不识别query中的连接符（or 、and），
- 查询时会将 “华为” 、"and"、 “手机” 分别进行查询然后将分词后的）

```
GET goods/_search
{
  "query": {
    "simple_query_string": {
      "fields": ["title","categoryName","brandName"],
      "query": "华为 AND 手机"
    }
  }
}
```

15-ElasticSearch查询-布尔查询-脚本

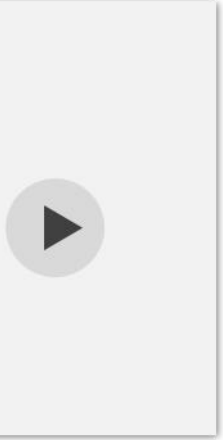
黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

布尔查询-脚本

掌握程度：掌握

要点提示

- 掌握布尔查询-脚本
- must和filter的区别



15-ElasticSearch查询-布尔查询-脚本

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

知识总结

布尔查询-脚本

boolQuery：对多个查询条件连接。连接方式：

- must（and）：条件必须成立
- must_not（not）：条件必须不成立
- should（or）：条件可以成立
- filter：条件必须成立，性能比must高。不会计算得分 （**得分:**即条件匹配度,匹配度越高，得分越高）

```
GET 索引名称/_search
{
  "query": {
    "bool": {
      "must": [ {}],
      "filter": [{}],
      "must_not": [{}],
      "should": [{}]
    }
  }
}
```

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

上午复习

- ElasticSearch批量操作命令
- ElasticSearch查询-matchAll-脚本
- ElasticSearch查询-模糊查询-三种方式
- ElasticSearch查询-布尔查询-连接方式
- ElasticSearch查询- term查询和match查询的区别
- ElasticSearch查询-queryString查询的两种方式

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

上午复习

- ElasticSearch批量操作命令POST _bulk{"delete":{"_index":"person1","_id":"5"}}
- ElasticSearch查询-matchAll-脚本 GET goods/_search{"query": {"match_all": {}}}
- ElasticSearch查询-模糊查询-三种方式 wildcard查询、正则查询、前缀查询
- ElasticSearch查询-布尔查询-连接方式 must、must_not、should、filter
- ElasticSearch查询- term查询和match查询的区别 term查询：不会对查询条件进行分词。match查询：会对查询条件进行分词
- ElasticSearch查询-queryString查询的两种方式 queryString查询、simple_query_string查询

16.ElasticSearch查询-布尔查询-JavaAPI

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

布尔查询-JavaAPI

掌握程度：掌握

要点提示

◆ 掌握布尔查询-JavaAPI

视
结

16.ElasticSearch查询-布尔查询-JavaAPI

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

知识总结

- 布尔查询-JavaAPI

案例：需求

1. 查询品牌名称为:华为
2. 查询标题包含：手机
3. 查询价格在：2000-3000

案例：实现步骤

① 构建boolQuery
② 构建各个查询条件
③ 测试

```
//1.构建boolQuery
BoolQueryBuilder boolQuery = QueryBuilders.boolQuery();
//2.构建各个查询条件
//2.1 查询品牌名称为:华为
TermQueryBuilder termQueryBuilder = QueryBuilders.termQuery("brandName", "华为");
boolQuery.must(termQueryBuilder);
//2.2. 查询标题包含：手机
MatchQueryBuilder matchQuery = QueryBuilders.matchQuery("title", "手机");
boolQuery.filter(matchQuery);
//2.3 查询价格在：2000-3000
RangeQueryBuilder rangeQuery = QueryBuilders.rangeQuery("price");
rangeQuery.gte(2000);
rangeQuery.lte(3000);
boolQuery.filter(rangeQuery);
sourceBuilder.query(boolQuery);
```

17.ElasticSearch查询-聚合查询-脚本

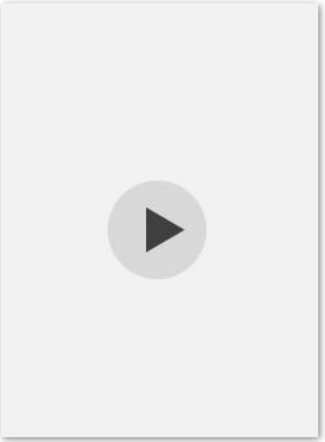
黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

聚合查询-脚本

掌握程度：应用

要点提示

- 掌握指标聚合的特性
- 掌握桶聚合的特性



视
结

17.ElasticSearch查询-聚合查询-脚本

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

知识总结

聚合查询-脚本

- 指标聚合：相当于MySQL的聚合函数。max、min、avg、sum等
- 桶聚合：相当于MySQL的 group by 操作。不要对text类型的数据进行分组，会失败

```
# 聚合查询
# 指标聚合 聚合函数
GET goods/_search
{
  "query": {
    "match": {
      "title": "手机"
    }
  },
  "aggs": {
    "max_price": {
      "max": {
        "field": "price"
      }
    }
  }
}
```

```
# 桶聚合  分组
GET goods/_search
{
  "query": {
    "match": {
      "title": "手机"
    }
  },
  "aggs": {
    "goods_brands": {
      "terms": {
        "field": "brandName",
        "size": 100
      }
    }
  }
}
```

18.ElasticSearch查询-聚合查询-javaAPI

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

聚合查询-javaAPI

掌握程度：应用

要点提示

◆ 聚合查询-javaAPI

18.ElasticSearch查询-聚合查询-javaAPI

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

知识总结

- 聚合查询-JavaAPI

案例：需求

聚合查询：桶聚合，分组查询

1.查询title包含手机的数据

2.查询品牌列表

案例：实现步骤

```
@Test
public void testAggQuery() throws IOException {
    SearchRequest searchRequest=new SearchRequest("goods");
    SearchSourceBuilder sourceBuilder=new SearchSourceBuilder();
    //1. 查询title包含手机的数据
    MatchQueryBuilder queryBuilder = QueryBuilders.matchQuery("title", "手机");
    sourceBuilder.query(queryBuilder);
    //2. 查询品牌列表  只展示前100条
    AggregationBuilder
aggregation=AggregationBuilders.terms("goods_brands").field("brandName").size(100);
    sourceBuilder.aggregation(aggregation);
    searchRequest.source(sourceBuilder);
    SearchResponse searchResponse = client.search(searchRequest, RequestOptions.DEFAULT);
    //7. 获取命中对象 SearchHits
    SearchHits hits = searchResponse.getHits();
    //7.1 获取总记录数
    Long total= hits.getTotalHits().value;
    System.out.println("总数: "+total);
    // aggregations 对象
    Aggregations aggregations = searchResponse.getAggregations();
    //将aggregations 转化为map
    Map<String, Aggregation> aggregationMap = aggregations.asMap();
    //通过key获取goods_brands 对象 使用Aggregation的子类接收  buckets属性在Terms接口中体现
    //        Aggregation goods_brands1 = aggregationMap.get("goods_brands");
    Terms goods_brands =(Terms) aggregationMap.get("goods_brands");
    //获取buckets 数组集合
    List<? extends Terms.Bucket> buckets = goods_brands.getBuckets();
    Map<String,Object>map=new HashMap<>();
    //遍历buckets   key 属性名，doc_count 统计聚合数
    for (Terms.Bucket bucket : buckets) {
        System.out.println(bucket.getKey());
        map.put(bucket.getKeyAsString(),bucket.getDocCount());
    }
    System.out.println(map);
}
```

19-ElasticSearch查询-高亮查询-脚本

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

高亮查询-脚本

掌握程度：应用

要点提示

- 高亮三要素
- 高亮字段、前缀、后缀
- 默认前后缀

19-ElasticSearch查询-高亮查询-脚本

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

知识总结

高亮查询-脚本

高亮三要素：
- 高亮字段
- 前缀
- 后缀

默认前后缀：手机

```
GET goods/_search
{
  "query": {
    "match": {
      "title": "电视"
    }
  },
  "highlight": {
    "fields": {
      "title": {
        "pre_tags": "<font color='red'>",
        "post_tags": "</font>"
      }
    }
  }
}
```

20.ElasticSearch查询-高亮查询-JavaAPI

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

高亮查询-JavaAPI

掌握程度：应用

要点提示

- 设置高亮字段
- 前后缀
- 将高亮了的字段数据，替换原有数据

20.ElasticSearch查询-高亮查询-JavaAPI

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

知识总结

- 高亮查询-JavaAPI

实施步骤：

高亮查询：

1.设置高亮段

2.前缀 后缀

3.将高亮了的字段数据，替换原有数据

```
@Test
public void testHighLightQuery() throws IOException {
    SearchRequest searchRequest = new SearchRequest("goods");
    SearchSourceBuilder sourceBulider = new SearchSourceBuilder();
    // 1. 查询title包含手机的数据
    MatchQueryBuilder query = QueryBuilders.matchQuery("title", "手机");
    sourceBulider.query(query);
    //设置高亮
    HighlightBuilder highlighter = new HighlightBuilder();
    //设置三要素
    highlighter.field("title");
    //设置前后缀标签
    highlighter.preTags("<font color='red'>");
    highlighter.postTags("</font>");
    //加载已经设置好的高亮配置
    sourceBulider.highlighter(highlighter);
    searchRequest.source(sourceBulider);
    SearchResponse searchResponse = client.search(searchRequest, RequestOptions.DEFAULT);
    SearchHits searchHits = searchResponse.getHits();
    //获取记录数
    long value = searchHits.getTotalHits().value;
    System.out.println("总记录数："+value);
    List<Goods> goodsList = new ArrayList<>();
    SearchHit[] hits = searchHits.getHits();
    for (SearchHit hit : hits) {
        String sourceAsString = hit.getSourceAsString();
        //转为java
        Goods goods = JSON.parseObject(sourceAsString, Goods.class);
        // 获取高亮结果，替换goods中的title
        Map<String, HighlightField> highlightFields = hit.getHighlightFields();
        HighlightField HighlightField = highlightFields.get("title");
        Text[] fragments = HighlightField.fragments();
        //highlight title替换 替换goods中的title
        goods.setTitle(fragments[0].toString());
        goodsList.add(goods);
    }
    for (Goods goods : goodsList) {
        System.out.println(goods);
    }
}
```

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

21.ElasticSearch重建索引&索引别名

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

重建索引&索引别名

掌握程度：应用

要点提示

- 索引一旦创建，只允许添加字段，不允许改变字段
- 改变字段，需要重建倒排索引，影响内部缓存结构，性能太低。
- 重建一个新的索引，并将原有索引的数据导入到新索引中

视
结

21.ElasticSearch重建索引&索引别名

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

知识总结

重建索引

随着业务需求的变更，索引的结构可能发生改变。

ElasticSearch的索引一旦创建，只允许添加字段，不允许改变字段。因为改变字段，需要重建倒排索引，影响内部缓存结构，性能太低。

那么此时，就需要重建一个新的索引，并将原有索引的数据导入到新索引中。

21.ElasticSearch重建索引&索引别名

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

知识总结

- 重建索引

```
# 创建新的索引 student_index_v2
PUT student_index_v2
{
  "mappings": {
    "properties": {
      "birthday":{
        "type": "text"
      }
    }
  }
}
```

```
# 将student_index_v1 数据拷贝到 student_index_v2
# _reindex 拷贝数据
POST _reindex
{
  "source": {
    "index": "student_index_v1"
  },
  "dest": {
    "index": "student_index_v2"
  }
}
```

22.ElasticSearch 集群-集群介绍

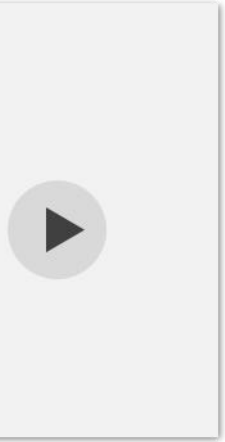
黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

ElasticSearch 集群-集群介绍

掌握程度：理解

要点提示

- 集群和分布式的概念
- 集群和分布式解决的问题



22.ElasticSearch 集群-集群介绍

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

知识总结

集群介绍

集群和分布式

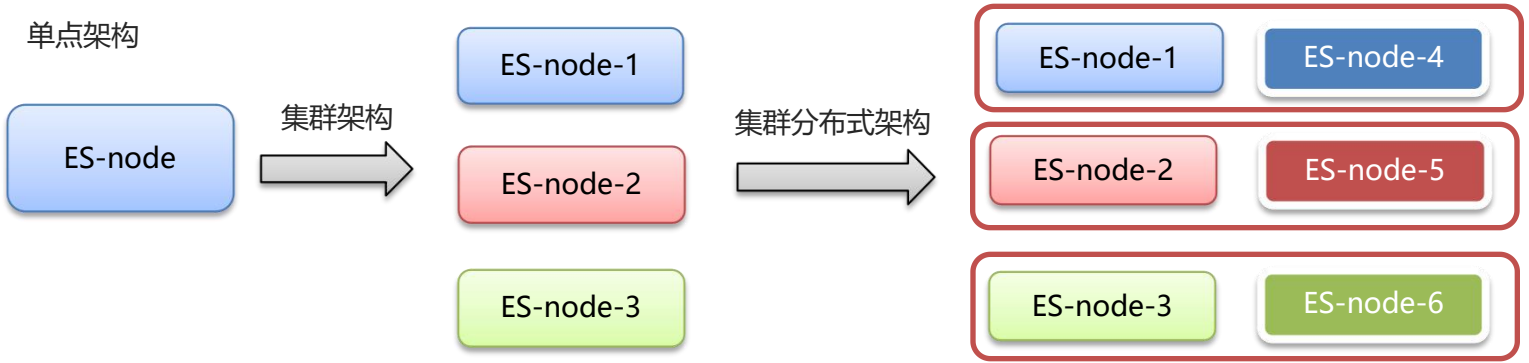
- 集群：多个人做一样的事。
- 分布式：多个人做不一样的事。

集群解决的问题：

- 让系统高可用
- 分担请求压力

分布式解决的问题：

- 分担存储和计算的压力，提速
- 解耦



23.ElasticSearch 集群-ES集群相关概念

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

ES集群相关概念

掌握程度：理解

要点提示

- 集群、节点的概念
- 索引概念
- 分片：主分片、副分片概念

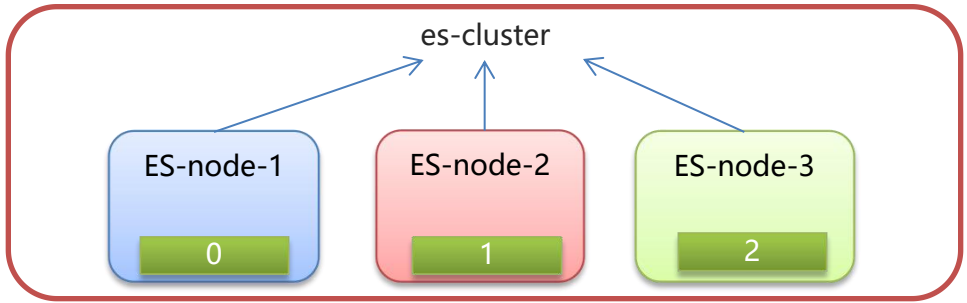
23.ElasticSearch 集群-ES集群相关概念

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

知识总结

ES集群相关概念

- 集群（cluster）：一组拥有共同的 cluster name 的 节点。
- 节点（node）：集群中的一个 Elasticearch 实例
- 索引（index）：es存储数据的地方。相当于关系数据库中的database概念
- 分片（shard）：索引可以被拆分为不同的部分进行存储，称为分片。在集群环境下，一个索引的不同分片可以拆分到不同的节点中
- 主分片（Primary shard）：相对于副本分片的定义。
- 副本分片（Replica shard）每个主分片可以有一个或者多个副本，数据和主分片一样。



24.ElasticSearch 集群-集群搭建

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

集群搭建

掌握程度：应用

要点提示

- 集群搭建

24.ElasticSearch 集群-集群搭建

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

知识总结

参见ElasticSearch 集群-集群搭建.md

视
结

25.ElasticSearch 集群-kibina管理集群

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

kibina管理集群

掌握程度：应用

要点提示

◆ S

视
结

25.ElasticSearch 集群-kibina管理集群

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

知识总结

Kibina配置文件

```
vim  kibana-7.4.0-linux-x86_64-cluster/config/kibana.yml
```

kibana.yml

```
#支持中文
i18n.locale: "zh-CN"
#5602避免与之前的冲突
server.port: 5602
server.host: "0.0.0.0"
server.name: "kibana-itcast-cluster"
elasticsearch.hosts:
["http://localhost:9201","http://localhost:9202","http://localhost:9203"]
elasticsearch.requestTimeout: 99999
```

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

26-ElasticSearch 集群-JavaAPI 访问集群

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

JavaAPI 访问集群

掌握程度：应用

要点提示

- 使用脚本测试集群
- 编写测试类测试集群

26-ElasticSearch 集群-JavaAPI 访问集群

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

知识总结

JavaAPI 访问集群

脚本测试：

```
# 创建索引、并添加映射
PUT cluster_test
{
  "mappings": {
    "properties": {
      "name":{
        "type": "text"
      }
    }
  }
}
#查看索引
GET cluster_test
#添加文档
POST /cluster_test/_doc/1
{
  "name":"张三"
}
#查看文档
GET cluster_test/_search
```

javaAPI测试：

```
@Resource(name="clusterClient")
RestHighLevelClient clusterClient;
@Test
public void testCluster() throws IOException {
    //设置查询的索引、文档
    GetRequest indexRequest=new GetRequest("cluster_test","1");
    GetResponse response = clusterClient.get(indexRequest, RequestOptions.DEFAULT);
    System.out.println(response.getSourceAsString());
}
```

27.ElasticSearch 集群-分片配置

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

分片配置

掌握程度：应用

要点提示

- 在创建索引时，如果不指定分片配置，则默认主分片1，副本分片1。
- 创索引时设置分片
- 集群自平衡

27.ElasticSearch 集群-分片配置

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

知识总结

集群原理-分片配置

- 在创建索引时，如果不指定分片配置，则默认主分片1，副本分片1。
- 在创建索引时，可以通过settings设置分片

```
"settings": {
  "number_of_shards": 3,
  "number_of_replicas": 1
},
```

- 分片与自平衡：当节点挂掉后，挂掉的节点分片会自平衡到其他节点中
- 在Elasticsearch 中，每个查询在每个分片的单个线程中执行，但是，可以并行处理多个分片。
- 分片数量一旦确定好，不能修改。
- 索引分片推荐配置方案：
    1. 每个分片推荐大小10-30GB
    2. 分片数量推荐 = 节点数量 * 1~3倍

思考：比如有1000GB数据，应该有多少个分片？多少个节点

40个分片　　20个节点

28-ElasticSearch 集群-路由原理

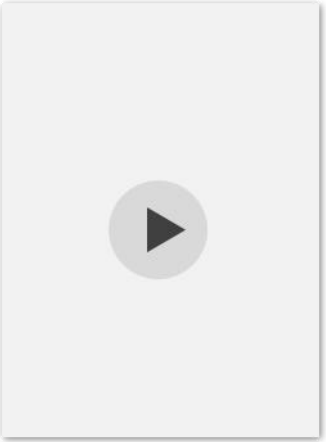
黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

集群-路由原理

掌握程度：理解

要点提示

- 文档存入对应的分片，ES计算分片编号的过程，称为路由。
- 路由算法： shard_index = hash(id) % number_of_primary_shards



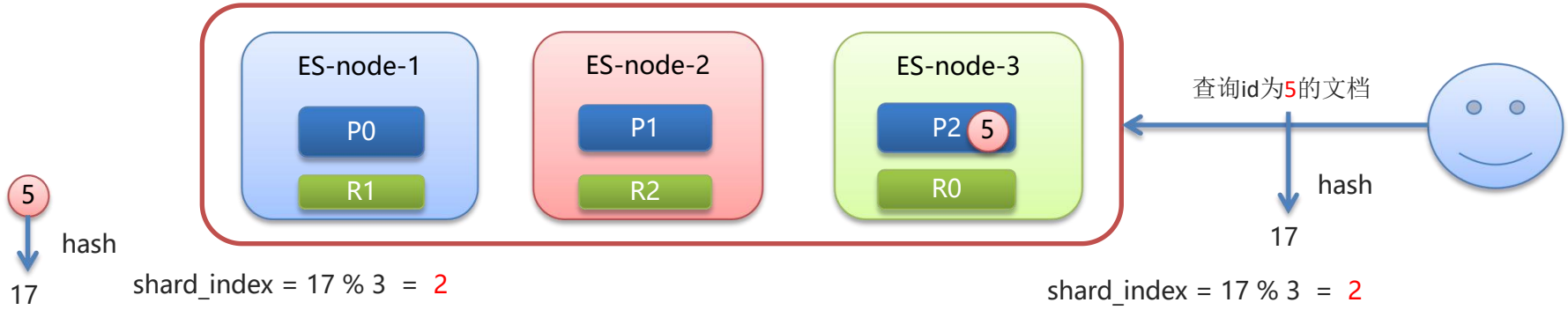
28-ElasticSearch 集群-路由原理

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

知识总结

集群原理-路由原理

- 文档存入对应的分片，ES计算分片编号的过程，称为路由。
- Elasticsearch 是怎么知道一个文档应该存放到哪个分片中呢?
- 查询时，根据文档id查询文档， Elasticsearch 又该去哪个分片中查询数据呢?
- 路由算法： shard_index = hash(id) % number_of_primary_shards



29-ElasticSearch 集群-脑裂

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

集群-脑裂

掌握程度：理解

要点提示

- 理解脑裂现象的原因
- 脑裂现象的解决办法

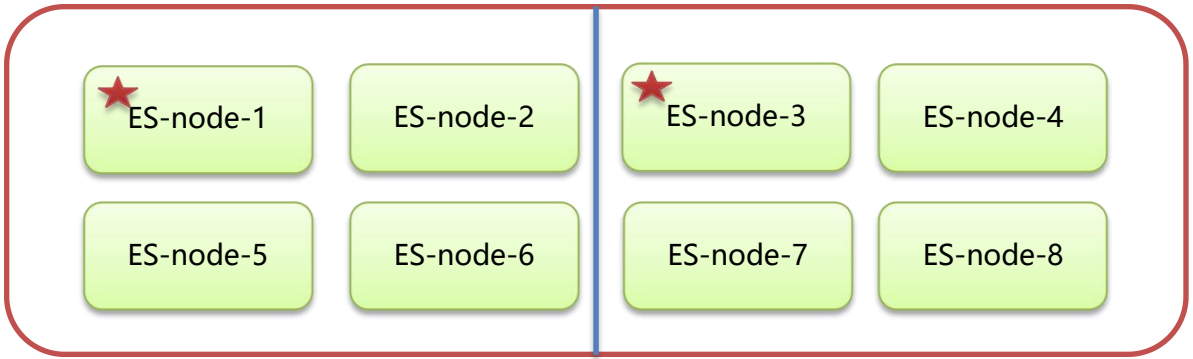
29-ElasticSearch 集群-脑裂

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

知识总结

脑裂

- 一个正常es集群中只有一个主节点（Master），主节点负责管理整个集群。如创建或删除索引，跟踪哪些节点是群集的一部分，并决定哪些分片分配给相关的节点。
- 集群的所有节点都会选择同一个节点作为主节点。
- 脑裂问题的出现就是因为从节点在选择主节点上出现分歧导致一个集群出现多个主节点从而使集群分裂，使得集群处于异常状态。



29-ElasticSearch 集群-脑裂

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

知识总结

脑裂原因

1. 网络原因：网络延迟
    - 一般es集群会在内网部署，也可能在外网部署，比如阿里云。
    - 内网一般不会出现此问题，外网的网络出现问题的可能性大些。
2. 节点负载
    - 主节点的角色既为master又为data。数据访问量较大时，可能会导致Master节点停止响应（假死状态）。

```
#是不是有资格主节点
node.master: true
#是否存储数据
node.data: true
```

3. JVM内存回收
    - 当Master节点设置的JVM内存较小时，引发JVM的大规模内存回收，造成ES进程失去响应。

29-ElasticSearch 集群-脑裂

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

知识总结

避免脑裂

1.网络原因：discovery.zen.ping.timeout 超时时间配置大一点。默认是3S

2. 节点负载：角色分离策略
    - 候选主节点配置为
        - node.master: true
        - node.data: false
    - 数据节点配置为
        - node.master: false
        - node.data: true

3. JVM内存回收：修改 config/jvm.options 文件的 -Xms 和 -Xmx 为服务器的内存一半。

30-ElasticSearch 集群-集群扩容

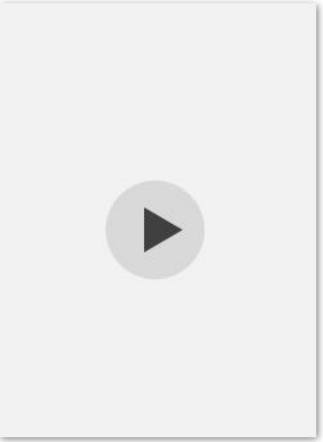
黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

集群扩容

掌握程度：应用

要点提示

- 参照集群搭建进行扩容



30-ElasticSearch 集群-集群扩容

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

知识总结

- 集群扩容

参见集群搭建.md

视
结

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

今日复习

- ElasticSearch批量操作命令
- ElasticSearch查询-matchAll-脚本
- ElasticSearch查询-模糊查询-三种方式
- ElasticSearch查询-布尔查询-连接方式
- ElasticSearch查询- term查询和match查询的区别
- ElasticSearch查询-queryString查询的两种方式
- ElasticSearch查询-聚合查询
- ElasticSearch 集群-路由原理
- ElasticSearch 集群-脑裂

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

今日复习

- ElasticSearch批量操作命令POST _bulk{"delete":{"_index":"person1","_id":"5"}}
- ElasticSearch查询-matchAll-脚本 GET goods/_search{"query": {"match_all": {}}}
- ElasticSearch查询-模糊查询-三种方式 wildcard查询、正则查询、前缀查询
- ElasticSearch查询-布尔查询-连接方式 must、must_not、should、filter
- ElasticSearch查询- term查询和match查询的区别 term查询：不会对查询条件进行分词。match查询：会对查询条件进行分词
- ElasticSearch查询-queryString查询的两种方式 queryString查询、simple_query_string查询
- ElasticSearch查询-聚合查询的两种方式 指标聚合、桶聚合
- ElasticSearch 集群-路由原理 通过路由算法：shard_index = hash(id) % number_of_primary_shards完成路由
- ElasticSearch 集群-脑裂的三个原因 1.网络延迟较高、2.节点负载：主节点的角色既为master又为data 3. JVM内存设置太小

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

目标检测

TARGETS DETECTION

- ElasticSearch-bulk批量操作
- ElasticSearch -模糊查询
- ElasticSearch -**queryString**查询
- ElasticSearch-布尔查询
- ElasticSearch-聚合查询
- ElasticSearch-高亮查询
- ElasticSearch 集群管理
- ElasticSearch-路由原理
- ElasticSearch-脑裂原因及解决办法

黑马程序员
www.itheima.com
传智播客旗下高端IT教育品牌

传智播客旗下高端IT教育品牌