# 基础

结构

索引 数据库

mapping 表

doc文档 一条记录

<!-- elasticsearch的客户端 -->

<dependency>

<groupId>org.elasticsearch.client</groupId>

<artifactId>transport</artifactId>

<version>5.4.3</version>

</dependency>

**其他搜索引擎**

Lucene java编写

Solr 基于Lucene

Elasticsearch

**为什么不用mysql做全文搜索？**

1. where name like '%tom%'; 不走索引，性能差
2. where name like 'tom%'; 走索引，但是数据量大依然扛不住
3. 全文搜索 insert update都会扫全表做索引平衡

# 命令操作

**精简版**

1.删掉nba索引

DELETE

http://192.168.1.206:9200/nba

2.新建一个索引，并且指定mapping

PUT

http://192.168.1.206:9200/nba

{

    "mappings": {

        "properties": {

            "name": {

                "type": "text"

            },

            "team\_name": {

                "type": "text"

            },

            "position": {

                "type": "text"

            },

            "play\_year": {

                "type": "long"

            },

            "jerse\_no": {

                "type": "keyword"

            }

        }

    }

}

PUT

http://192.168.1.206:9200/nba/\_doc/1

{

    "name": "哈登",

    "team\_name": "火箭",

    "position": "得分后卫",

    "play\_year": 10,

    "jerse no": "13"

}

PUT

http://192.168.1.206:9200/nba/\_doc/2

{

    "name": "库里",

    "team\_name": "勇士",

    "position": "控球后卫",

    "play\_year": 10,

    "jerse\_no": "30"

}

PUT

http://192.168.1.206:9200/nba/\_doc/3

{

    "name": "詹姆斯",

    "team\_name":"湖人",

    "position":"小前锋",

    "play\_year":15,

    "jerse\_no":"23"

}

term（词条）查询和full text（全文）查询

**词条查询**：词条查询不会分析查询条件，只有当词条和查询字符串完全匹配时，才匹配搜索。

**全文查询**：ElasticSearch引擎会先分析查询字符串，将其拆分成多个分词，只要已分析的字段中

包含词条的任意一个，或全部包含，就匹配查询条件，返回该文档；如果不包含任意一个分词，表

示没有任何文档匹配查询条件

POST

http://192.168.1.206:9200/nba/\_search

{

    "query":{

        "term":{

            "jerse\_no":"23"

        }

    }

}

**linux:**

curl -X GET "http://localhost:9200"

**postman:**

**创建索引**

http://192.168.1.206:9200/nba

**创建mapping（表）**

可以新增字段覆盖上去，但是不能修改

{

    "properties":{//指定字段的一些信息

        "name":{

            "type":"text"//会被分词

        },

        "team\_name":{

            "type":"text"

        },

        "position":{

            "type":"keyword"//关键字 不会被分词

        },

        "play\_year":{

            "type":"keyword"

        },

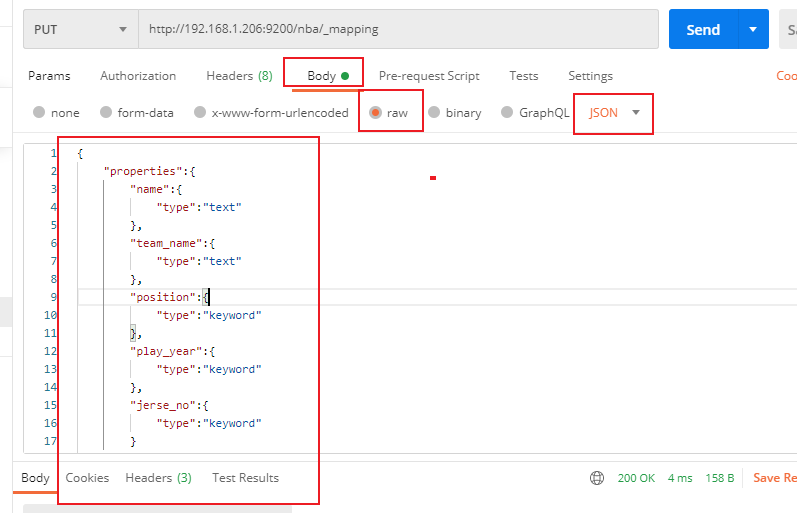
        "jerse\_no":{

            "type":"keyword"

        }

    }

}



**新增文档**

{

    "name":"库里",

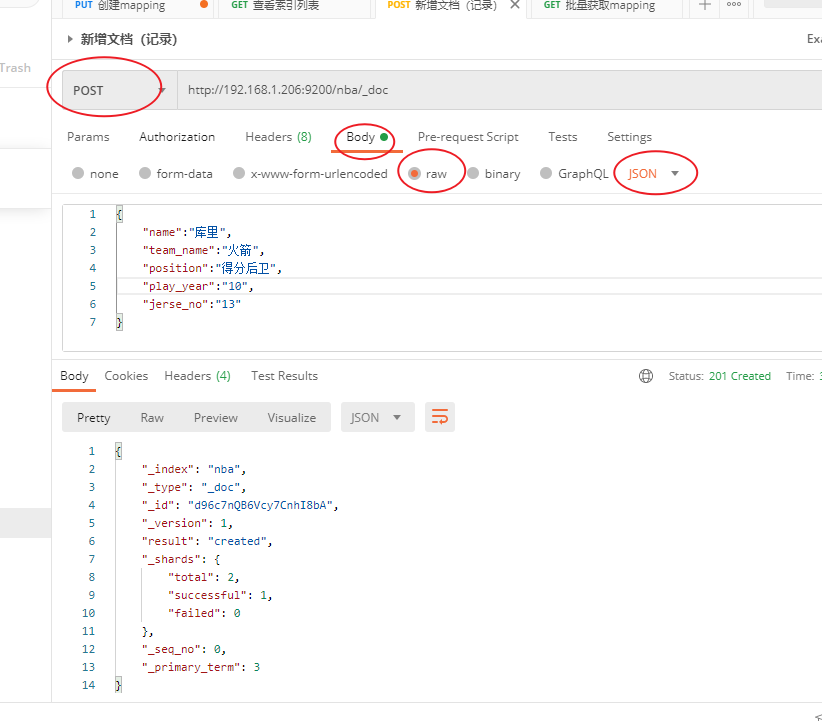
    "team\_name":"火箭",

    "position":"得分后卫",

    "play\_year":"10",

    "jerse\_no":"13"

}



**获取文档**

http://192.168.1.206:9200/nba/\_doc/d96c7nQB6Vcy7CnhI8bA

# 安装

## ElasticSearch

安装

docker pull elasticsearch:7.4.2 本体

docker pull kibana:7.4.2 可视化工具

配置es

mkdir -p /ghaya/elasticsearch/config

mkdir -p /ghaya/elasticsearch/data

mkdir -p /ghaya/elasticsearch/plugins

chmod -R 777 /ghaya/elasticsearch

echo "http.host:0.0.0.0">>/mydata/elasticsearch/config/elasticsearchyml

挂载并启动es

docker run --name elasticsearch -p 9200:9200 -p 9300:9300 \

-e "discovery.type=single-node" \

-e ES\_JAVA\_OPTS="-Xms64m -Xmx128m" \

-v /ghaya/elasticsearch/config/elasticsearch.yml:/usr/share/elasticsearch/config/elasticsearch.yml \

-v /ghaya/elasticsearch/data:/usr/share/elasticsearch/data \

-v /ghaya/elasticsearch/plugins:/usr/share/elasticsearch/plugins \

-d elasticsearch:7.4.2

## kibana

启动kibana可视化界面

docker run --name kibana -e ELASTICSEARCH\_HOSTS=http://192.168.1.206:9200 -p 5601:5601 \

-d kibana:7.4.2