1、通过索引一个文档来创建索引，会自动创建映射、setting

put localhost:9200/flight/segment/1

1）、{

"dep\_port":"PEK",

"arr\_port":"SYX",

"base\_fare":2200,

"flight\_date":"2019-07-01",

"dep\_time":"2019-07-01 08:00",

"cabin":{

"cabin\_code":"H",

"cabin\_class":"Economy"

}

}

2）、{

"dep\_port":"PEK",

"arr\_port":"CTU",

"base\_fare":1690,

"flight\_date":"2019-01-03",

"dep\_time":"2019-01-03 08:00",

"cabin":{

"cabin\_code":"Y",

"cabin\_class":"Economy",

"isOpen":false

}

}

3）、{

"dep\_port":"CTU",

"arr\_port":"PEK",

"base\_fare":1690,

"flight\_date":"2019-01-05",

"dep\_time":"2019-01-05 08:00",

"cabin":{

"cabin\_code":"B",

"cabin\_class":"Economy",

"rule":"不得签转"

}

}

2、创建结构化索引：

put localhost:9200/flight

{

        "mappings": {

            "segment": {

                "properties": {

                    "dep\_port": {

//不分词

                        "type": "keyword"

                    },

                    "arr\_port": {

//分词

                        "type": "text",

                        "fields": {

                            "keyword": {

//不分词，用于排序

                                "type": "keyword"

                            }

                        },

//默认为false，为true表示支持聚合操作，比如分组

                        "fielddata":"true"

                    },

                    "base\_fare": {

//精确值

                        "type": "long"

                    },

                    "dep\_time": {

                        "type": "text",

                        "fields": {

                            "keyword": {

                                "type": "keyword"

                            }

                        }

                    },

                    "arr\_time": {

                        "type": "keyword"

                    },

                    "flight\_date": {

//精确值

                        "type": "date"

                    },

                    "cabin": {

                        "properties": {

                            "cabin\_class": {

                                "type": "text",

                                "fields": {

                                    "keyword": {

                                        "type": "keyword"

                                    }

                                }

                            },

                            "cabin\_code": {

                                "type": "text",

                                "fields": {

                                    "keyword": {

                                        "type": "keyword"

                                    }

                                }

                            },

                            "rule": {

                                "type": "text",

                                "fields": {

                                    "keyword": {

                                        "type": "keyword"

                                    }

                                },

//指定分析器

                                "analyzer":"english"

                            },

                            "isOpen": {

/\*逻辑类型\*/

                                "type": "boolean"

                            }

                        }

                    }

                }

            }

        },

        "settings": {

            "number\_of\_shards": "3",

            "number\_of\_replicas": "1"

        }

}

3、查询表达式搜索：\_search API

get/post localhost:9200/flight/segment/\_search 或者不要type（6.0只允许一个，是否指定type不重要）：localhost:9200/flight/\_search

{

"query":{

"match": {

"flight\_date" : "2019-07"

}

}

}

4、查询字符串搜索：

<http://localhost:9200/flight/segment/_search?q=arr_port:SYX>

5、删除指定id的文档

delete  localhost:9200/flight/segment/6hzUHmwBbuUvY8OjsVZb 6hzUHmwBbuUvY8OjsVZb为id

6、查看某一个索引的情况：aliases、mappings、settings：

get <http://localhost:9200/flight?pretty>

7、查看所有索引的基本信息：

get  <http://localhost:9200/_cat/indices?v>

8、删除索引：

delete <http://localhost:9200/flight>

9、全文搜索相关性得分：

get http://localhost:9200/flight/segment/\_search

{

"query": {

"match": {"dep\_time" : "2019-01-03" }

}

}

按其他字段排序，不计算相关性的得分：

{

"query": {

"match": {"dep\_time" : "2019-01-03" }

},

"sort":{

"dep\_time.keyword" : {

"order":"desc"

}

}

}

精确匹配（过滤），相关性得分都一样：

{

"query": {

"bool":{

"filter": {

"match" : {"flight\_date" : "2019-01-05" }

}

}

}

}

10、不同分析器测试：

自带的：chinese、english、standard 插件：中文分词：ik\_max\_word、ik\_smart

get <http://localhost:9200/_analyze>

{

"analyzer": "chinese",

"text": "Text to analyze 张三去坐飞机"

}