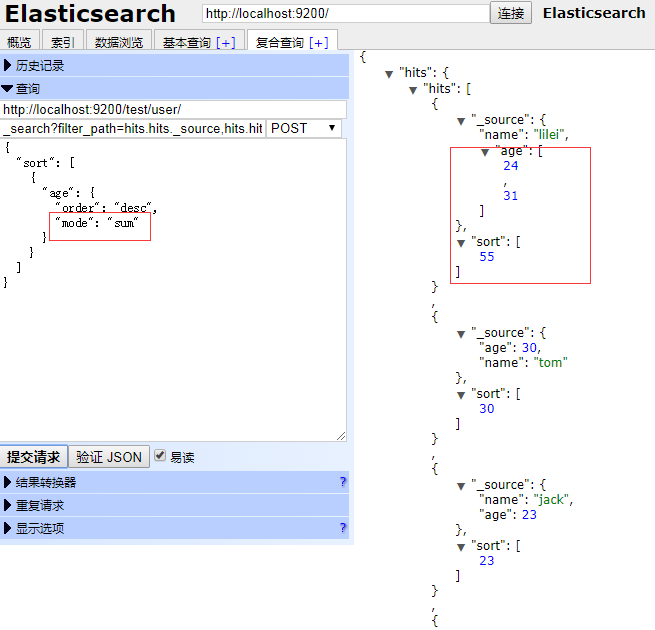
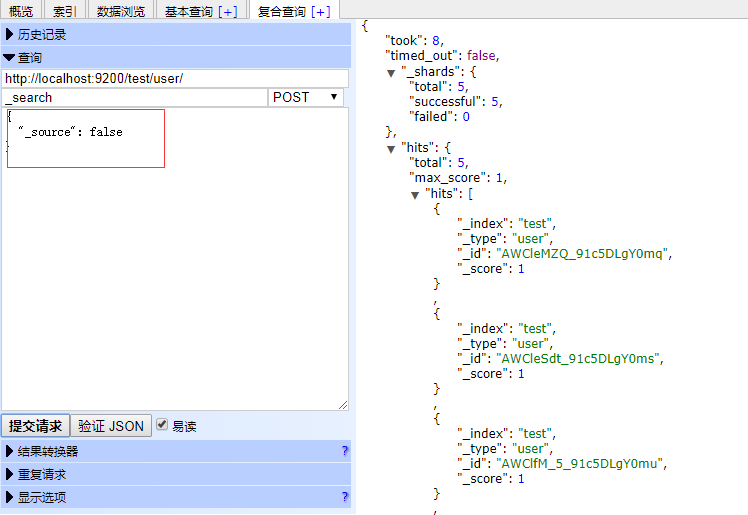
ElasticSearch搜索

1、Sort:

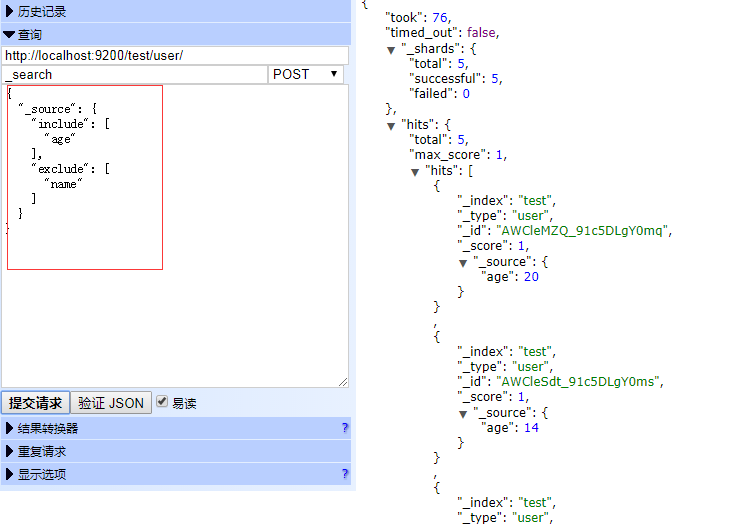
当一个字段的内容有多个值，系统支持一些计算（min,max,sum,avg,median(中间值)）后再排序



1. 显示数据时不显示\_source即不返回原始文档



1. 排除source里不需要返回的信息



1. 脚本的支持

{

"script\_fields": {

"test1": {

"script": "doc['age']\*2"

}

}

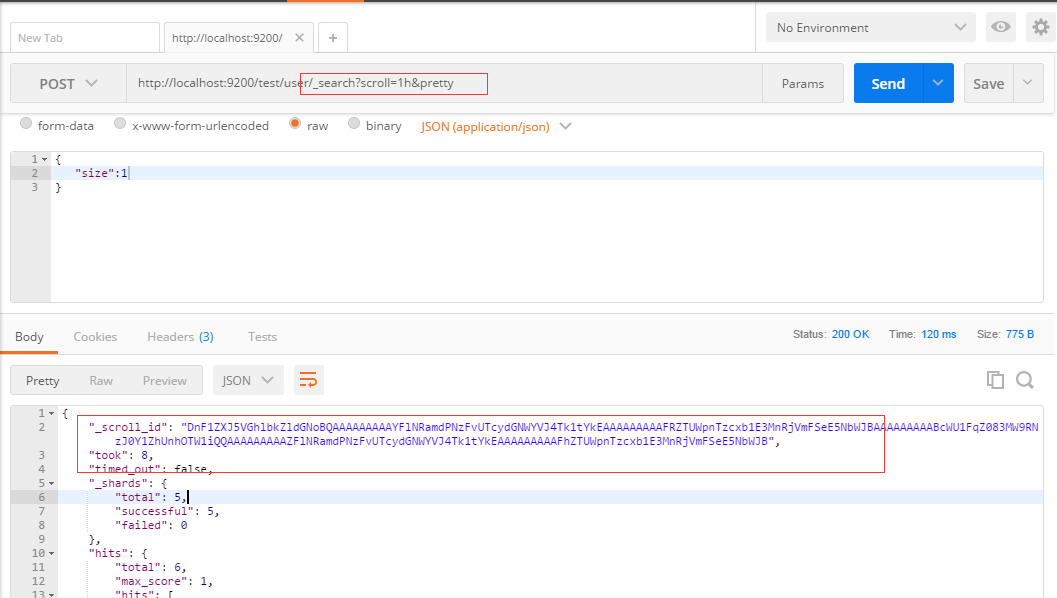
}

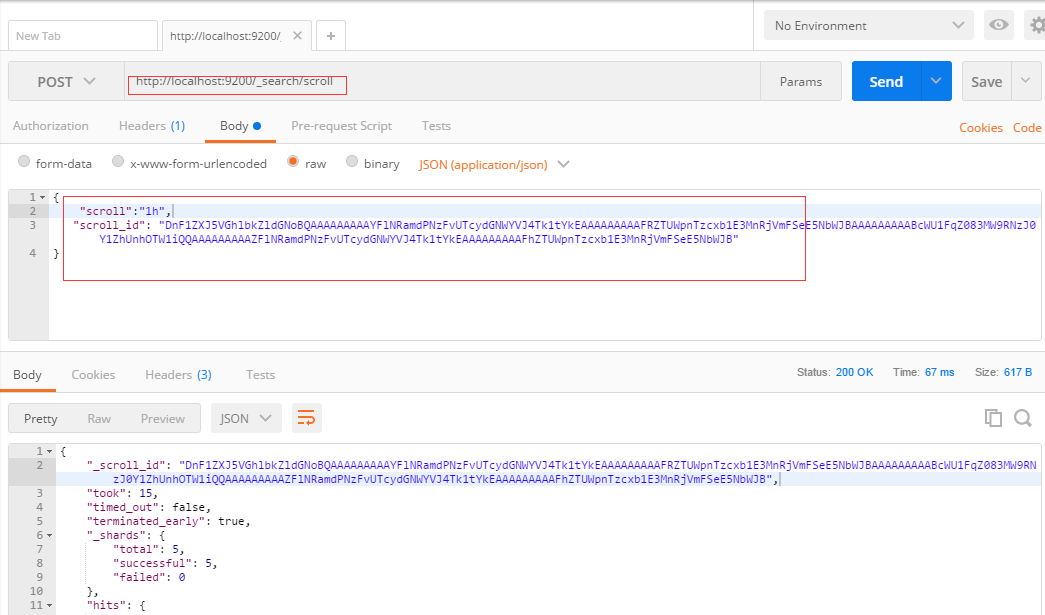
1. 评分算法

<http://blog.csdn.net/molong1208/article/details/50623948>

1. 滚动查询(深度分页scroll)

最好用postman测试





其中这个scroll\_id是迭代查询中返回的

1. [Nested分析](http://blog.csdn.net/u012332735/article/details/62222953)

nested结构是Elasticsearch提供关系存储的一种特殊的结构,是NOSQL的一种高级特性

参考：<http://blog.csdn.net/u012332735/article/details/62222953>

1. inner hits

<https://www.elastic.co/guide/en/elasticsearch/reference/current/search-request-inner-hits.html>

(1)先设置属性

{

"doc": {

"properties": {

"comments": {

"type": "nested"

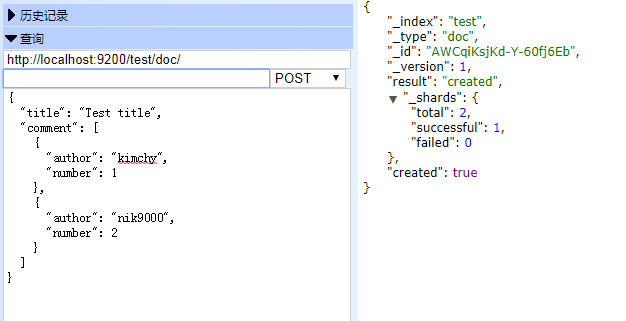
}

}

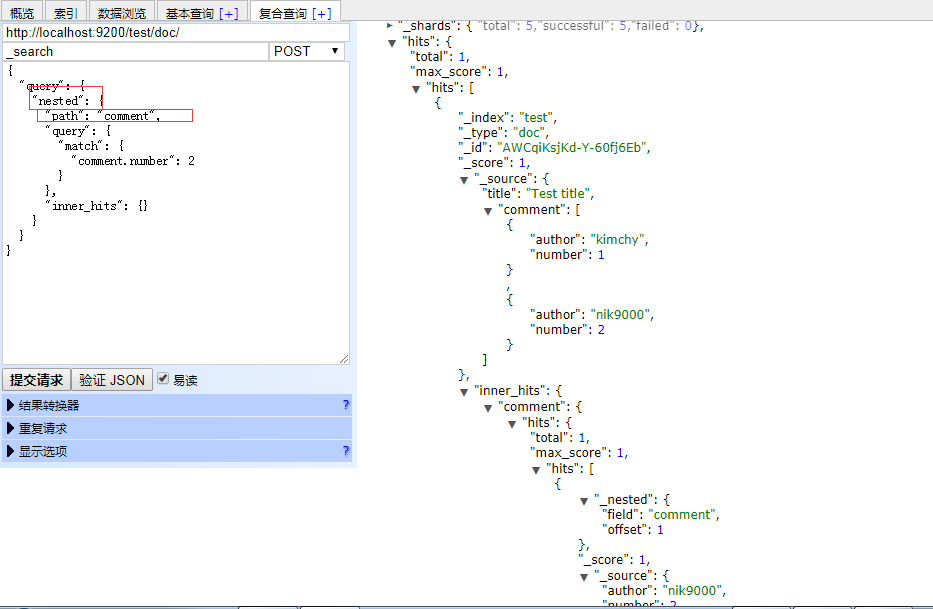
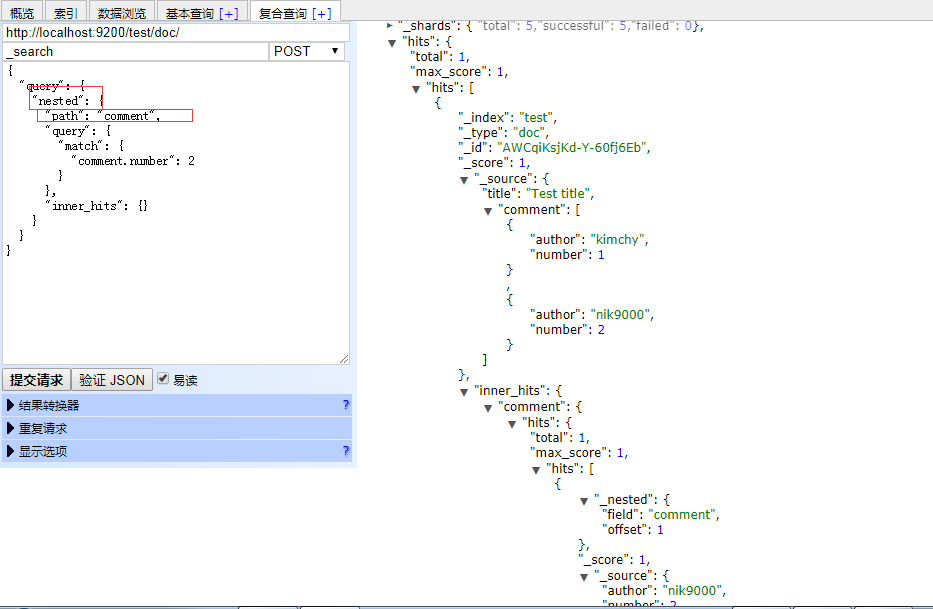
}

}

(2)插入数据



(3)查询数据



1. 搜索相关函数

(1)preference

在es查询的时候我们可以控制Preference，来完成特定shard或节点上的数据查询，默认情况下查询是随机的。

假如现在我们有一份索引5个shard和3个副本，当一个查询请求过来的时候，查询操作如何执行，在没有使用路由的情况下5个shard的数据肯定都要查询，然后查询5个shard时候到底查的是主shard还是replia都是随机的。

通过偏好查询，我们可以有更多的选择定制，比如我们可以设置只查询主shard，或者只查replia，或者仅仅查某几个节点的数据。

下面来介绍下Preference相关的参数：

（1）\_primary：只查询主shard，也就是说不管你有多少个副本，只对主shard进行检索，这种场景可以用在所有副本不可用的时候，强制读取主shard数据。

（2）\_primary\_first：优先读取主shard，如果主shard无效或者失败，则会读取其他shard

（3）\_replica：只查询replia

（4）\_replica\_first：优先查询replia，如果replia无效就查询其他的shard。

（5）\_local：尽可能在本地执行查询，不跨网络

（6）\_prefer\_nodes:abc,xyz 在指定的节点id上执行查询

（7）\_shards:2,3 查询指定分片上的数据，此外这种写法还可以和前面的用法组合，如：\_shards:2,3|\_primary ，查询分片2和3且在主节点上的数据

（8）\_only\_nodes ：限制在特定的node上执行操作

（9）Custom (string) value ：使用自定义的值来保证同一个值的数据，在一个shard里面，感觉有点像routing字段的功能，暂时没太理解如何使用这个功能，因为官网给的例子，只有查询，在索引的时候没有看到设置preference的功能：

GET /\_search?preference=xyzabc123

{

"query": {

"match": {

"title": "elasticsearch"

}

}

}

大多数时候是不需要刻意使用Preference功能的，除非在一些特殊情况下，如shard不可读，shard失败，shard未分配情况，或者某些节点挂了，我们可以指定读取正常shard的数据，虽然不完整，但也比整个服务不可用体验要好

1. 索引加权（indices\_boost）

当在多个索引中搜索时， 可以使用参数 indices\_boost 来提升整个索引的权重，在下面例子中，当要为最近索引的文档分配更高权重时，可以这么做：

GET /docs\_2014\_\*/\_search IMG_256

{

"indices\_boost": { IMG_257

"docs\_2014\_10": 3,

"docs\_2014\_09": 2

},

"query": {

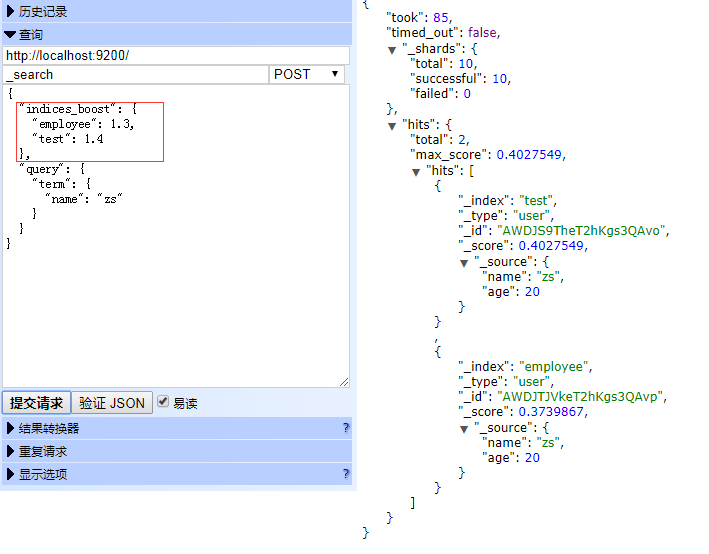
"match": {

"text": "quick brown fox"

}

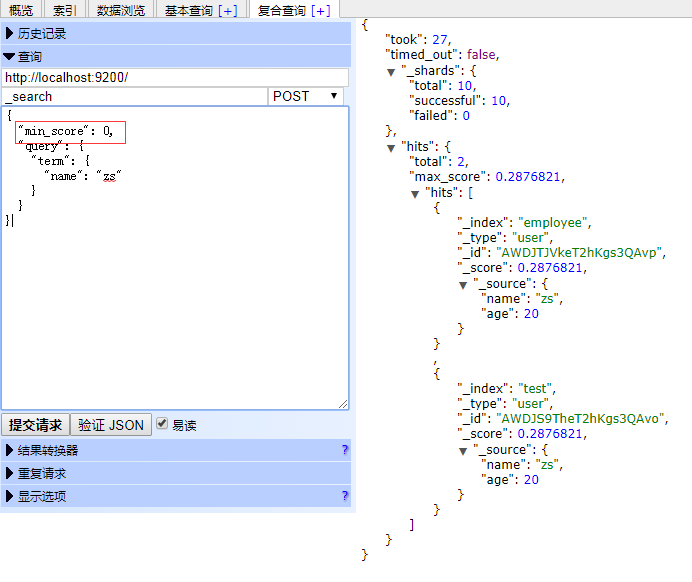
}

}



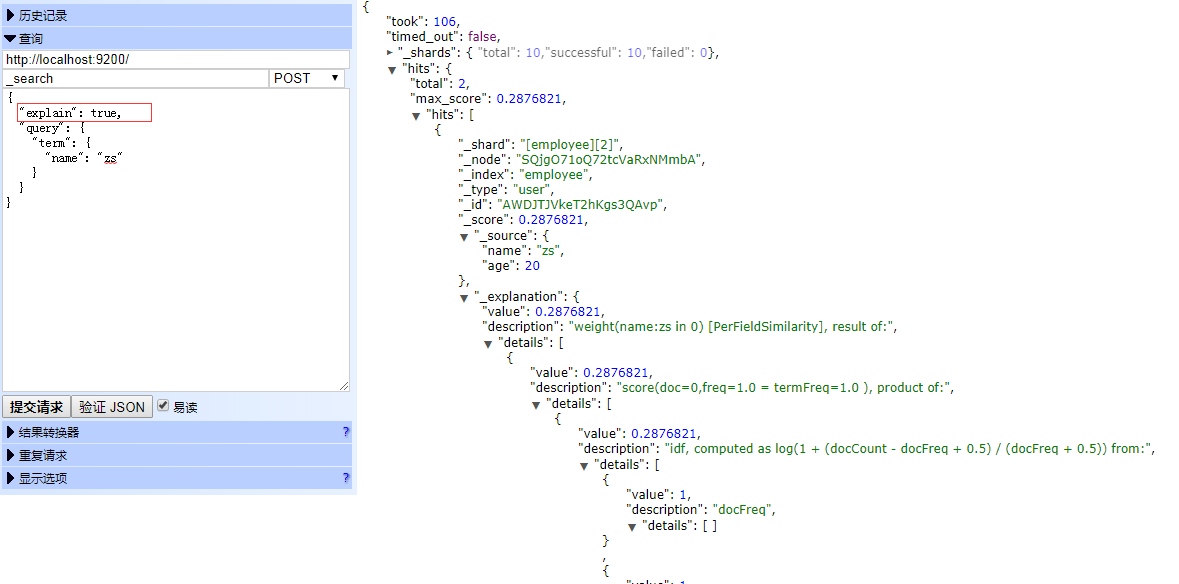
1. 最小分值(min\_score)

在搜索是指定最小评分分值

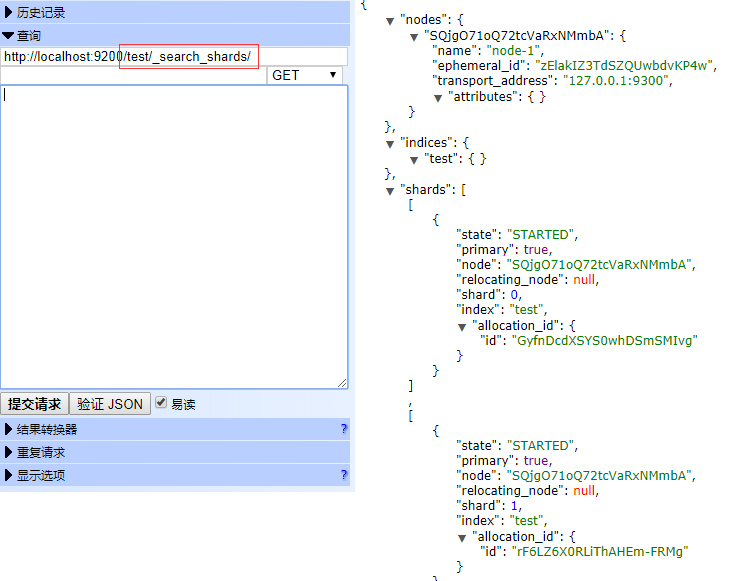


1. 分值解释(explain)

解释命中的查询评分是怎么计算出来的



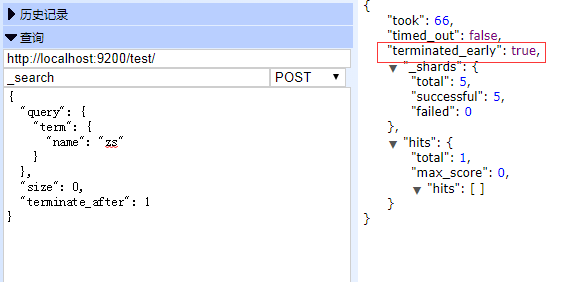
1. 查询分片情况(\_search\_shards)



1. 是否存在查询

terminate\_after设置为1，size设置为0

如果存在则查询结果里会显示："terminated\_early": true,



1. 判断查询语句是否合法

<http://localhost:9200/test/_validate/query/> post



(8)查询模板