# ES

## Tips

所谓一个表的倒排索引就是mysql你去搜索一个表中bankename 包含六年级的班课,你是扫表挨个去比较的,而es是已经对表中数据建好了倒排索引,直接就出来了.

还有什么你搜索subject是4的班课,这样的话mysql也是扫表,而es是把subject是4的课已经建好了倒排索引了,直接就出来了.

nosql数据库列是可以动态添加的.

分布式搜索引擎 nosql 对外提供检索服务通过http的restful的json格式.

对内就是一个nosql的数据库

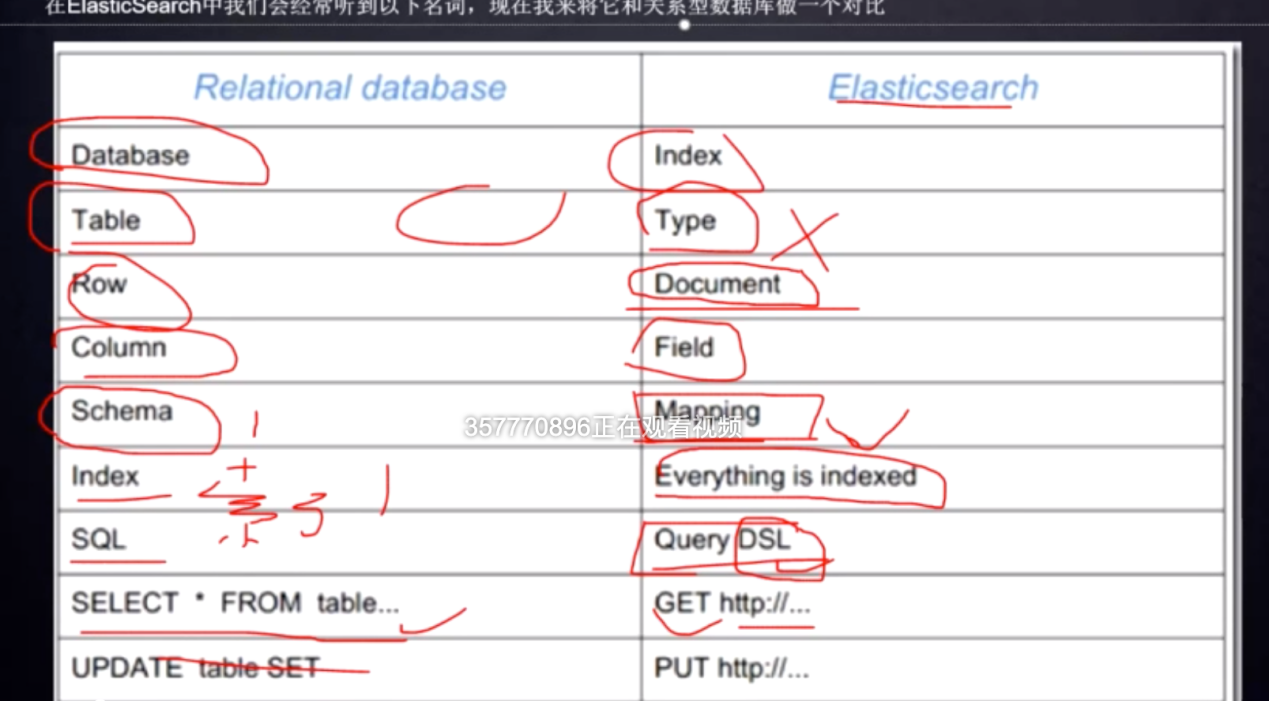
索引 = 数据库 很多个

类型type = 表 es6.x只有一个type 之前的可以创建很多个,es7.x就没有表了

文档doc = 行数据

Field

Nosql 非关系型 没办法链接查询



DSL

阿里云自带openjdk,要删除掉.

Numberofshards 分片数量

Number of relicas 每一片的备份数量

分布索引一定要注意分片数量调整是比较麻烦当然,所以在创建的时候一定要先预估好数据的大小,一般在8核16g的机器上一个分片不要超过500g.

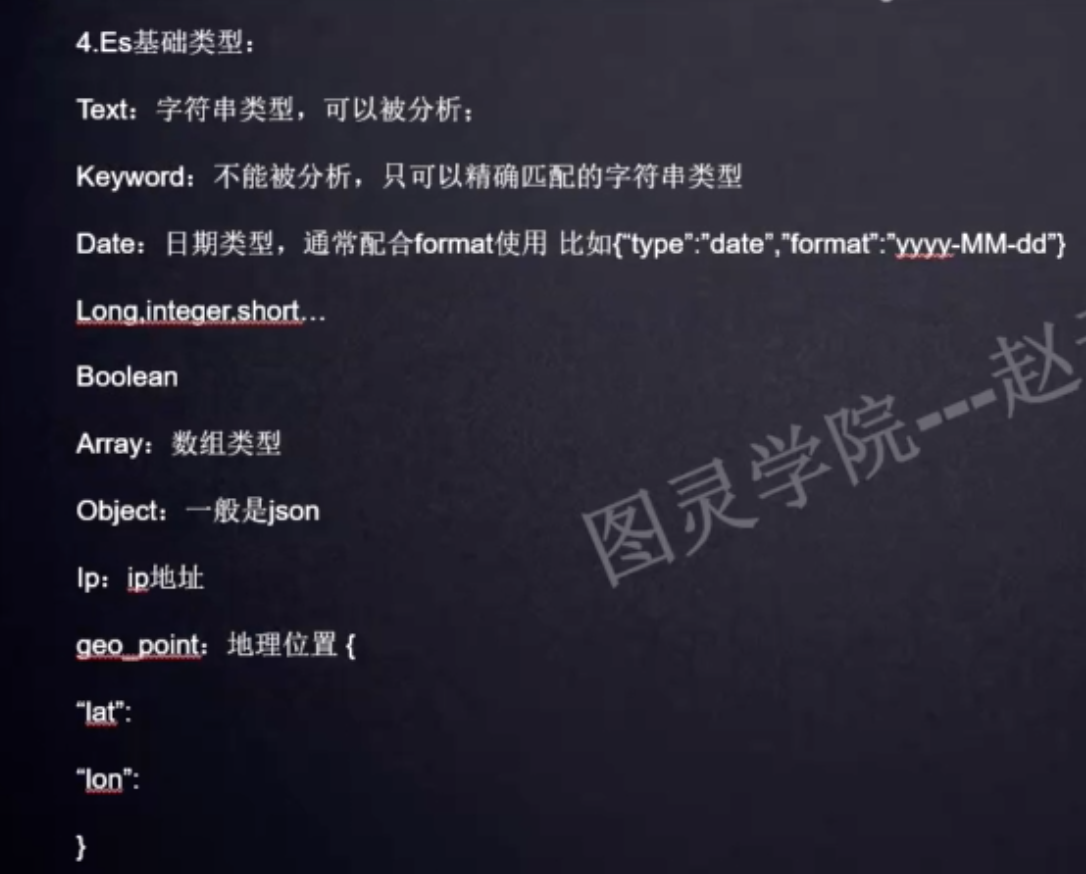
如果调整了分片那么就要重建索引.

es也是集群的,可以弄多个节点.

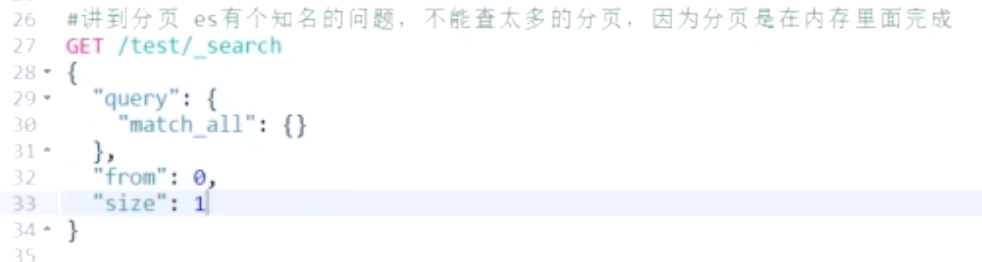
不会以某一个机器为主,不同的分片的主所在的机器可能不一样.



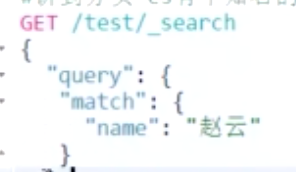
结构化创建索引,数据会有约束.



基本类型 text是可以被分词的,keyword是不会被分词的.



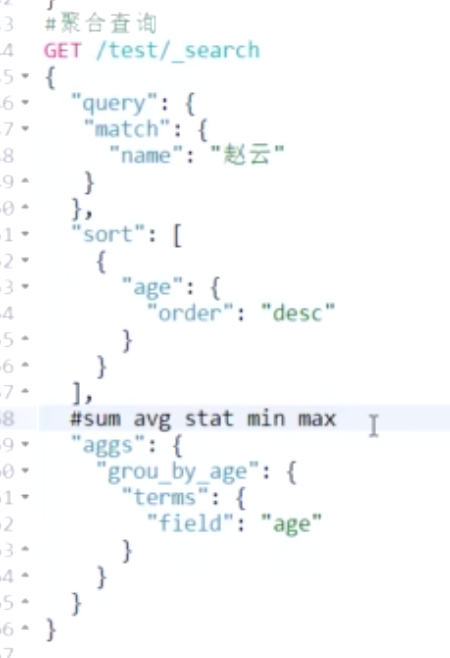
分页查询



条件查询



排序查询



聚合查询



创建索引的时候指定英文分词.

英文分词可以吧什么runing apples给分成run和apple,英文分词会把这些东西标准化.

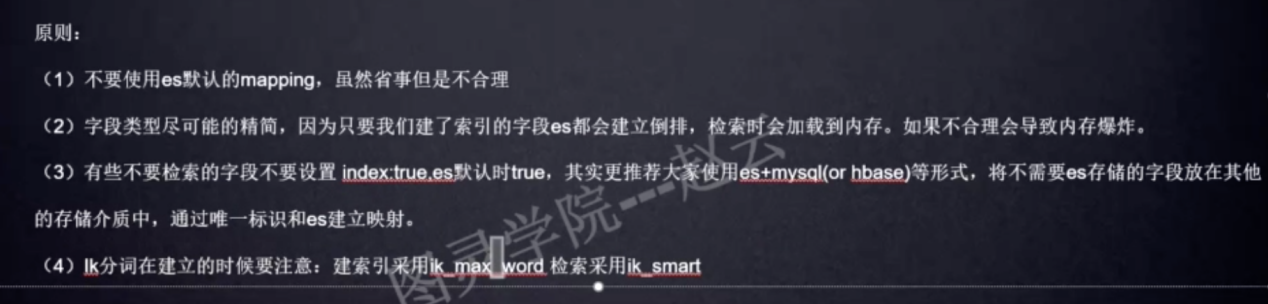
es的分词

标准 垃圾 就是用空格分了下英文,对中文就是每个字分的

英文分词 english 分英文会标准化,很好 也是es自带的

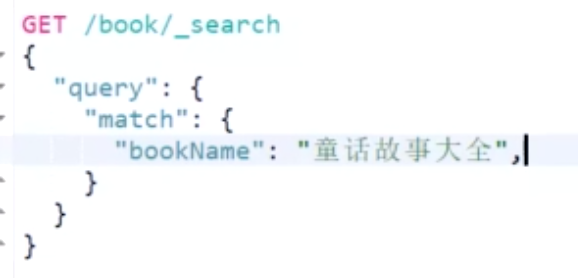
中文分词 ik分词器 需要安装ik插件

中英文都有就用ik就好了

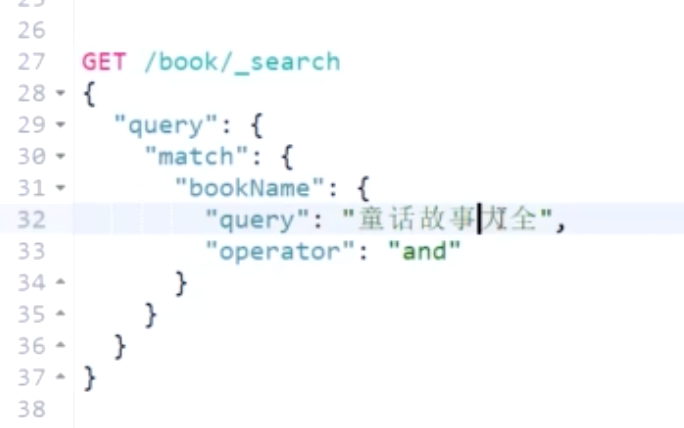


推荐只把需要索引的id字段存到es,不需要的字段配合mysql使用.

因为检索时会用到内存,不合理内存会爆炸.



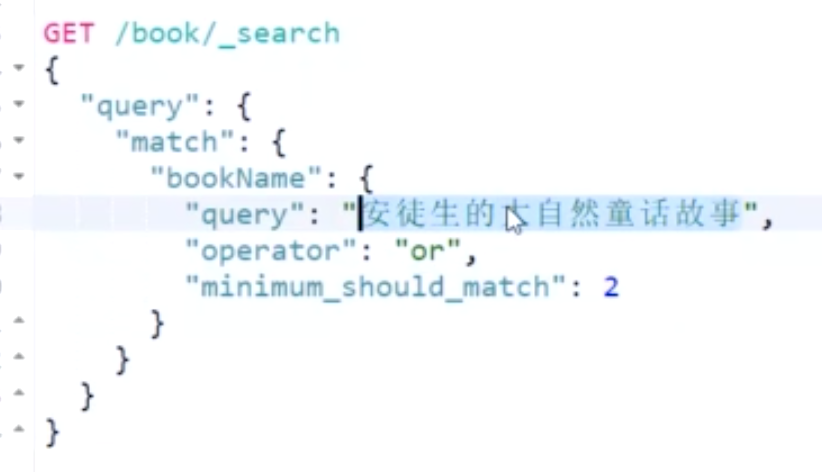
如果这样查的话,他会先分词, 分成童话故事和大全,包含其中一个就会查出来.



而如果这样的话,加了个and,就是先分词,查询出既包含大全也包含童话故事的.



这个term查询就不会分词,必须包含童话故事大全才行.



加了minshouldmatch表示,至少要匹配两个分词的才会查出来.



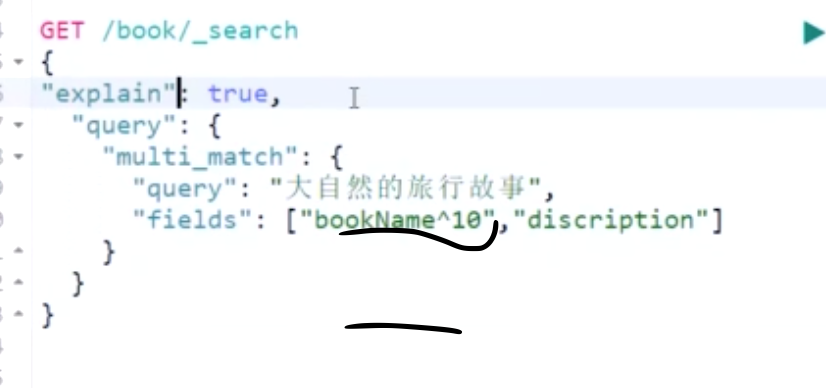
可以看到查询结果会有个东西加score,就是表示数据和你的查询条件的匹配程度.



加入了explain就可以知道每一条的记录是怎么算出来的了



多字段查询,这两个字段有一个包含上面的分词就会被查出来,这样的分算的也更明显,字段符合的越多的分肯定也越高.



我们还可以设置权重,肯定书名符合的越高的越在前,我们就把书名的权重设置成10



Query\_string

这个就是可以手动指定分词.

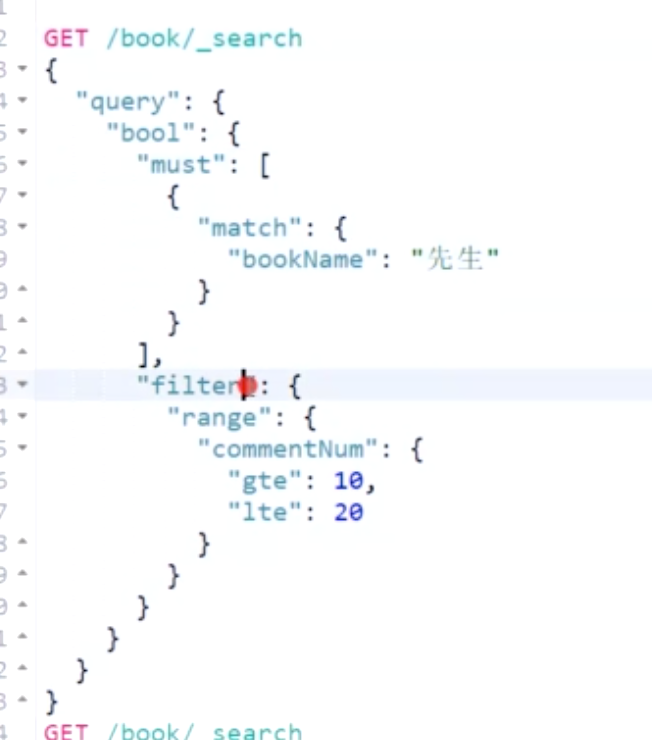


这个是布尔查询,

适用于多个条件, must\_not must should

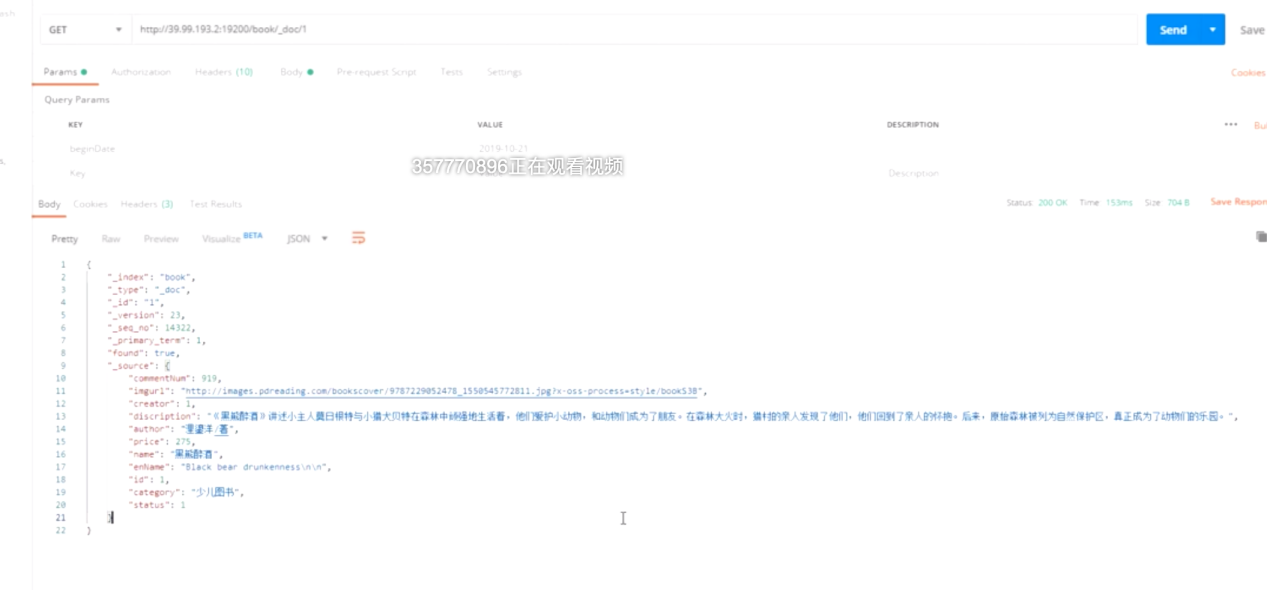


布尔查询, 里面用filter 搞范围查询



先must查询,然后在fliter过滤, 注意必须是must,不能用should.

还可以配置同义词,比如你搜苹果,apple也会出来,估计也是个插件.



看上图的url 直接是ip端口 索引/type/id 就直接查询了

postman也是可以的.

Canal 这种监听mysqlbinlog的,其实就是假装自己是个slave,然后获取master的同步信息.

这种不仅可以同步es,同步redis也是可以的咯.