

# Springboot2.x

# ElasticSearch7.x

## ---整合实战

主讲：Array老师

# 目录/CONTENT

01

概述

02

技术选型

03

实战代码手把手

04

学习方法和工作方法

# 课程如何观看?



# 讲师简介

## 讲师职称：

现某知名大型互联网公司资深架构师，技术总监，职业规划师，首席面试官，曾在某上市培训机构，高校任教多年。

## 讲师介绍：

Array（Array老师）10多年互联网公司实战经验，知名的大型互联网公司的架构师，高管等职，在企业长期从事于技术的源码阅读和新技术的研究；擅长于职业规划，面试辅导，从事面试官多年；擅长于JAVA,人工智能AI应用,Xmind等等，曾服役于国内某上市培训机构数年，独特的培训思路，培训体系，培训方式，实践的职场技能，职场现状，职场晋升等让你快速适应企业职场的所需。



# 适用人群说明

## 亮点1

零基础到快速上手研发，而非理论/原理性讲解；告别繁琐的原理，直接上手的手把手整合实战课；

## 亮点2

适合零基础的同学或者对Springboot/SpringCloud微服务想快速整合ElasticSearch的同学

## 亮点3

本课都是从商业项目中剥离出来的例子，都是一线的技术企业级的技术，也是目前最流行互联网的技术。

A circular progress indicator with a dark gray border. The interior is white, and the percentage '88%' is displayed in a large, bold, black font. A small segment of the circle's border is highlighted in a lighter gray, indicating the progress level.

88%

### 时间历练

节约大量的时间

A circular progress indicator with a dark gray border. The interior is white, and the percentage '95%' is displayed in a large, bold, black font. A small segment of the circle's border is highlighted in a lighter gray, indicating the progress level.

95%

### 弯道超车

用户可以快速上手和脚手架的使用

# Springboot和es版本关系

We plan on deprecating the TransportClient in Elasticsearch 7.0 and removing it completely in 8.0. Instead, you should be using the Java High Level REST Client, which executes HTTP requests rather than serialized Java requests. The migration guide describes all the steps needed to migrate. The Java High Level REST Client currently has support for the more commonly used APIs, but there are a lot more that still need to be added. You can help us prioritise by telling us which missing APIs you need for your application by adding a comment to this issue: [Java high-level REST client completeness](#). Any missing APIs can always be implemented today by using the low level Java REST Client with JSON request and response bodies.

- (旧) 使用SpringBoot整合Elasticsearch 对应版本的关系:

Spring Boot Version (x)	Spring Data Elasticsearch Version (y)	Elasticsearch Version (z)	
x <= 1.3.5	y <= 1.3.4z <= 1.7.2*	-	
x >= 1.4.x	2.0.0 <=y < 5.0.0**	2.0.0 <= z < 5.0.0**	

- (新) elasticsearch-rest-high-level-client

1. Java high-level REST client 是目前官方推荐使用的客户端,  
2. High Level REST Client 与 Elasticsearch 具有相同的发布周期。  
所以, 能够使用最新版的 Elasticsearch。



# 方式1:

<https://docs.spring.io/spring-data/elasticsearch/docs/3.2.0.RC3/reference/html/#preface.versions>

The following table shows the Elasticsearch versions that are used by Spring Data Elasticsearch:

Spring Data Elasticsearch	Elasticsearch
3.2.x	6.8.1
3.1.x	6.2.2
3.0.x	5.5.0
2.1.x	2.4.0
2.0.x	2.2.0
1.3.x	1.5.2



## 方式2: Elasticsearch官方推荐

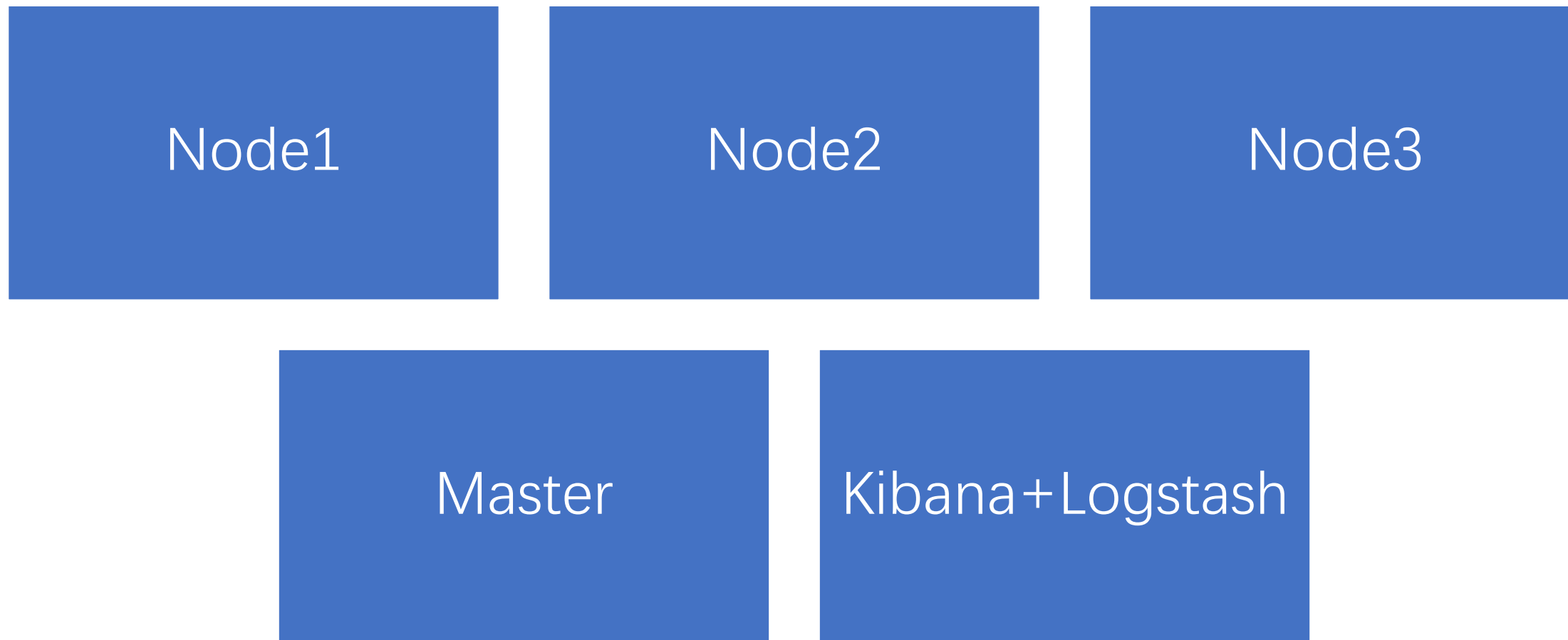
```
<!--  
https://mvnrepository.com/artifact/org.elasticsearch.client/el  
asticsearch-rest-high-level-client -->  
<dependency>  
  <groupId>org.elasticsearch.client</groupId>  
  <artifactId>elasticsearch-rest-high-level-  
client</artifactId>  
  <version>7.3.2</version>  
</dependency>
```



# ◆ Elasticsearch7.x常用操作



# ELK尤其是ElasticSearch7.x集群安装





# 搜索原理

## 正排索引

文档id到文档内容、单词的关联关系

## 倒排索引

单词到文档id的关联关系

## 单词词典

记录所有文档的单词，一般比较大  
记录单词到倒排列表的关联信息  
一般为B+Tree

# 倒排列表

01

文档id

用于获取原始信息

03

位置

记录单词在文档的分词位置，用于词语搜索，高亮显示

总结

单词频率

记录该单词在文档出现的次数，用于后续相关性算分

02

偏移

记录单词在文档的开始和结束的位置

04

总结



用户输入关键词 → 通过倒排索引先从单词字典中的b+tree结构定位到某个子节点，该子节点对应到倒排列表中的某个位置 → 通过倒排列表的对应位置的信息获取对应的文档id → 根据文档id查询到完整的内容并返回结果  
在es中，每个字段都会有自己的倒排索引

所有教程资料均受著作权法保护，已经版权局备案，未经（Array老师）许可任何人不得擅自使用，否则将依法承担高达人民币50万元赔偿责任



# 中文分词搜索

es中的测试分词api接口为\_analyze



Character Filter

针对原始文本进行处理，  
去除html标记符



Tokenizer

将原始文本按一定的规则  
切分为单词



Token Filters

针对tokenizer处理的单词  
进行再加工，比如转小写，  
删除或新增某些字符

- 1.分词是指将文本转换成一系列的单词的过程，也可以叫文本分析，在es里面称为Analysis。
- 2.为什么要使用分词呢？可以简单的理解为提高搜索的准确度，分词器越合适，形成的倒排索引在查询时得到的结果也越准确



## Es体系介绍

- **es**中的测试分词**api**接口为**\_analyze**
- **es**中的测试分词**api**接口为**\_analyze**

官网介绍:

<https://www.elastic.co/guide/en/elasticsearch/reference/7.3/analysis-standard-analyzer.html>

### Example output

```
POST _analyze
{
  "analyzer": "standard",
  "text": "The 2 QUICK Brown-Foxes jumped over the lazy dog's bone."
}
```



# 默认接口

http://192.168.186.128:9200/\_analyze

GET

192.168.186.128:9200/\_analyze

Send

Params

Authorization

Headers (9)

Body

Pre-request Script

Tests

Settings

Cookies Code

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL BETA **JSON**

```
1 {  
2   "analyzer": "standard",  
3   "text": "The 2 QUICK Brown-Foxes jumped over the lazy dog's bone."  
4 }
```

Body Cookies Headers (3) Test Results

Status: 200 OK Time: 71ms Size: 383 B Save Response

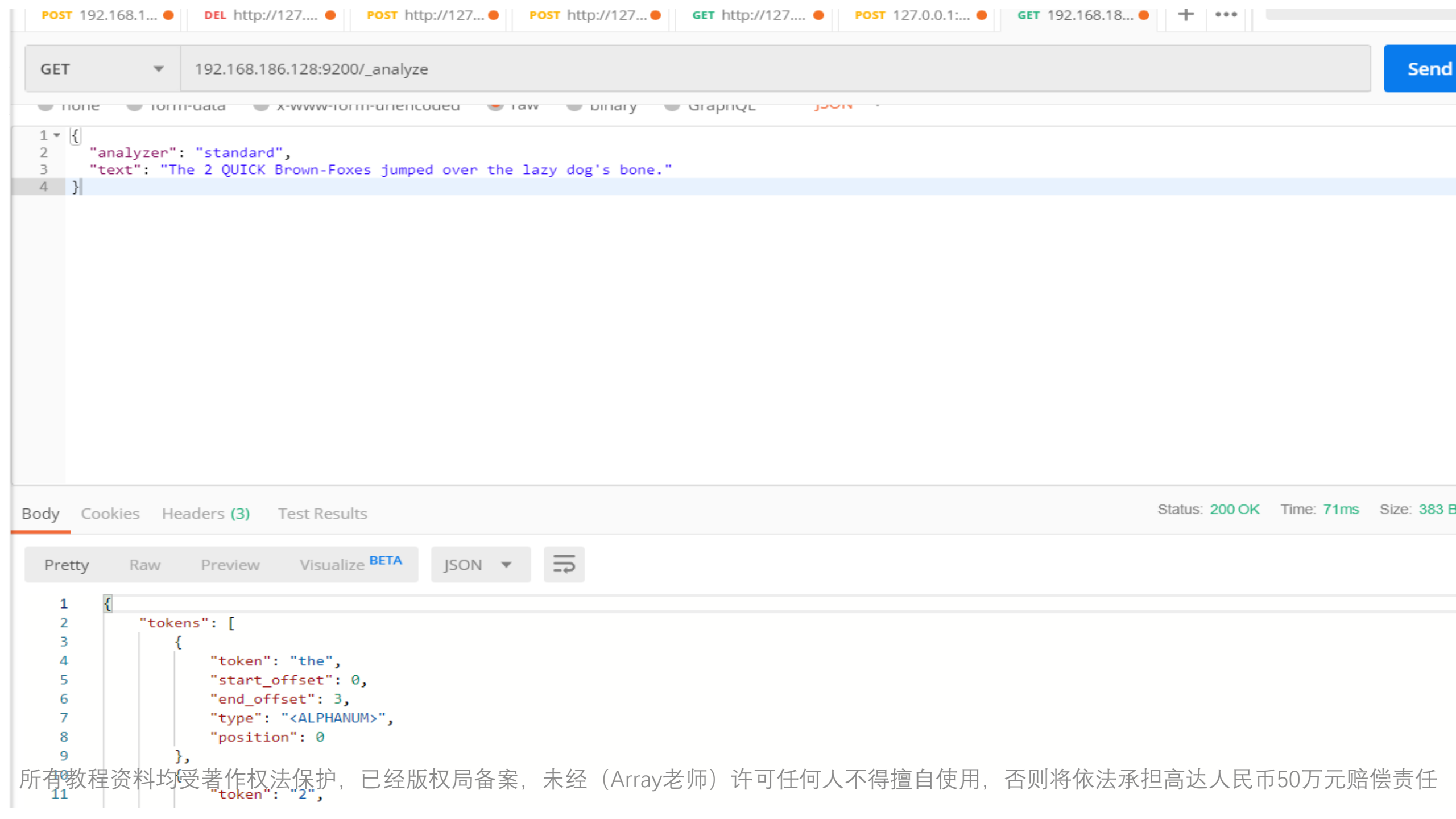
Pretty Raw Preview Visualize BETA

JSON



```
1 {  
2   "tokens": [  
3     {  
4       "token": "the",  
5       "start_offset": 0,  
6       "end_offset": 3,  
7       "type": "<ALPHANUM>",  
8       "position": 0
```

所有教程资料均受著作权法保护，已经版权局备案，未经（Array老师）许可任何人不得擅自使用，否则将依法承担高达人民币50万元赔偿责任







# elasticsearch内置分词器

## standard

默认分词器，小写处理，去掉stop word（要配置），按词切分，支持多语言，

## simple

按照非字母切分，小写处理

## whitespace

按空格切分

## language

提供多种常见语言分词器

## stop

同simple多了stopword处理，去除一些不必要的字段

## keyword

不分词，直接将输入作为一个单词输出

## parttern

通过正则表达自定义分隔符，默认是\W+,非字词符号作为分隔符

## elasticsearch-analysis-ik

解压到plugin的ik目录下，重启es





# 番茄教学法



老师内容

照搬内容

融会贯通

举一反三

# 高薪启蒙

/ 高薪启蒙研究院 / 专业只为高薪 / 零基础项目实战 / 职场辅导 /

## 感谢您的欣赏！