大连理工大学
Dalian University Of Technology

信息检索

题目：搭建小型全文检索系统

	组员 1	组员 2
班级	电计 1805	电计 1805
学号	201885097	201857190
姓名	张展瑞	胡泷文

一、实验目标

本次实验的目标是搭建一个完整、可运行的小型全文检索实验系统。通过该系统，用户可以自定义输入需要查询的博客关键字信息，系统后台自动根据关键字进行检索和倒排索引，最终以 WEB 的形式输出索引数据信息。

二、相关原理与工具

整体系统的数据流程分述如下。

首先自己搭建爬虫模块，将爬取的博客数据存储在本地的 mysql 数据库中，通过数据同步将数据同步到 ElasticSearch 中，再通过 Flask 搭建前后端，完成最终用户数据检索交互。

整体系统搭建用到的工具有：

数据库可视化工具：Navicat

ElasticSearch 数据可视化工具：ElasticSearch-head

Web 开发工具：Pycharm 专业版

三、开发环境和运行环境

本项目开发环境与运行环境保持相同：

编译器：Python3.7

前端：HTML + JQuery + Bootstrap

后端：Django3.1.2

数据库：Mysql5.7

分布式检索：ElasticSearch7.15.2

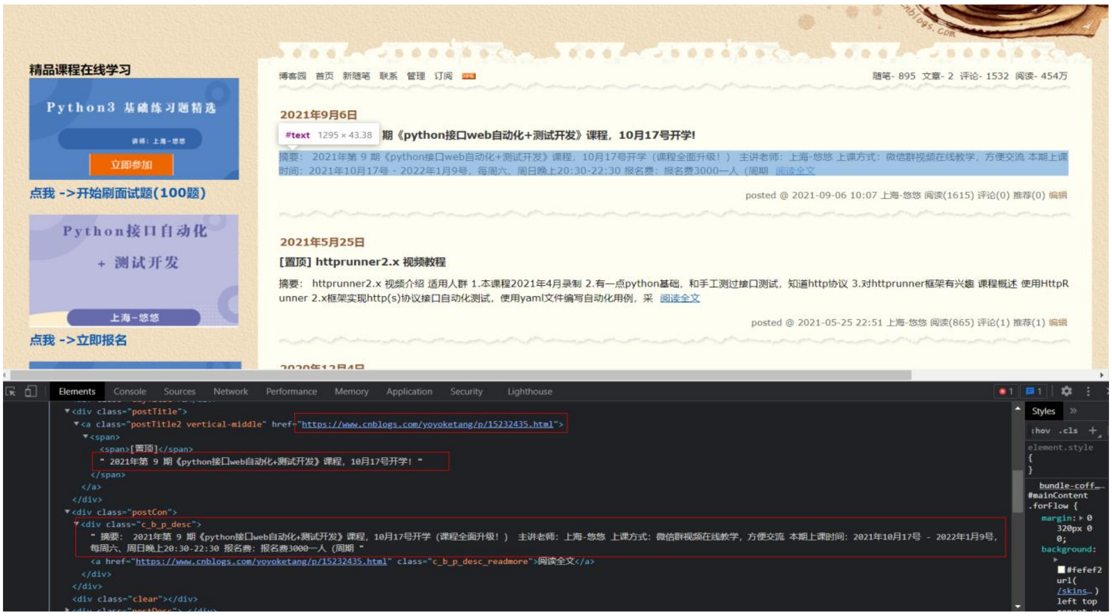
爬虫：requests + BeatifulSoup

四、工作分工

张展瑞：主要负责网站结构分析、爬虫部分与数据持久化

（1）网站结构分析

通过分析目标博客网站结构，根据节点递归解析发现如下图规则：


精品课程在线学习
Python3 基础练习题精选
立即参加
点我 ->开始刷面试题(100题)
Python接口自动化 + 测试开发
上海-悠悠
点我 ->立即报名
博客园 首页 新随笔 联系 管理 订阅
随笔- 895 文章- 2 评论- 1532 阅读- 454万
#text 1295 × 43.38
Elements Console Sources Network Performance Memory Application Security Lighthouse
Styles
:hov .cls +

2021年9月6日

期《python接口web自动化+测试开发》课程，10月17号开学!

摘要：2021年第 9 期《python接口web自动化+测试开发》课程，10月17号开学（课程全面升级！）主讲老师：上海-悠悠 上课方式：微信群视频在线教学，方便交流 本期上课时间：2021年10月17号 - 2022年1月9号，每周六、周日晚上20:30-22:30 报名费：报名费3000一人（周期 阅读全文

posted @ 2021-09-06 10:07 上海-悠悠 阅读(1615) 评论(0) 推荐(0) 编辑

2021年5月25日

[置顶] httprunner2.x 视频教程

摘要：httprunner2.x 视频介绍 适用人群 1.本课程2021年4月录制 2.有一点python基础，和手工测过接口测试，知道http协议 3.对httprunner框架有兴趣 课程概述 使用HttpRunner 2.x框架实现http(s)协议接口自动化测试，使用yaml文件编写自动化用例，采 阅读全文

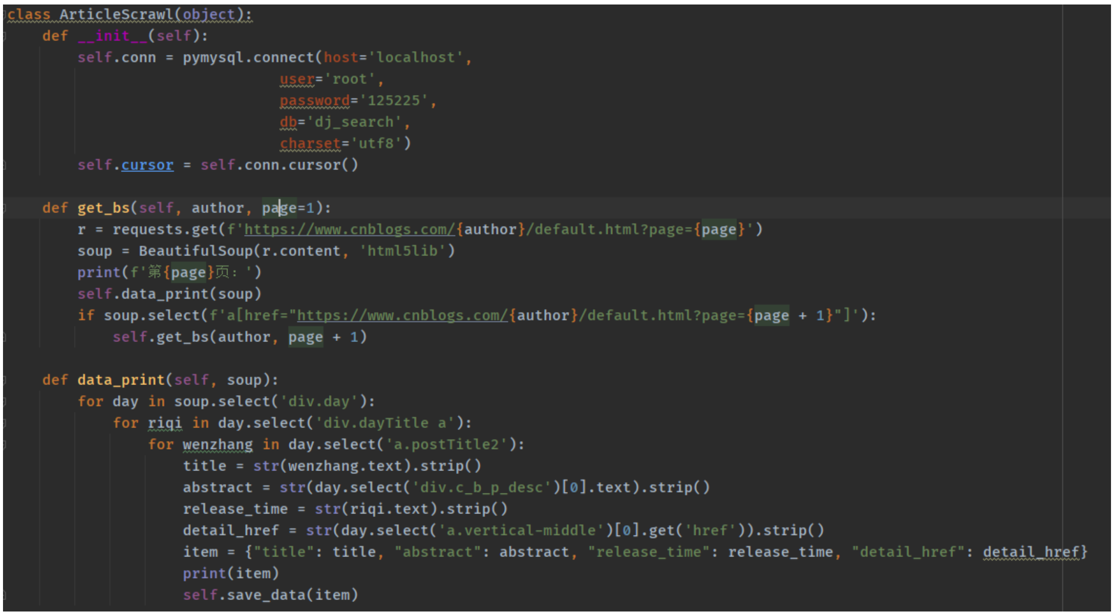
posted @ 2021-05-25 22:51 上海-悠悠 阅读(865) 评论(1) 推荐(1) 编辑

```
▼<div class="postTitle">
  ▼<a class="postTitle2 vertical-middle" href="https://www.cnblogs.com/yoyoketang/p/15232435.html">
    ▼<span>
       <span>[置顶]</span>
       " 2021年第 9 期《python接口web自动化+测试开发》课程，10月17号开学! "
     </span>
   </a>
 </div>
▼<div class="postCon">
  ▼<div class="c_b_p_desc">
     " 摘要: 2021年第 9 期《python接口web自动化+测试开发》课程，10月17号开学（课程全面升级！） 主讲老师：上海-悠悠 上课方式：微信群视频在线教学，方便交流 本期上课时间：2021年10月17号 - 2022年1月9号，
     每周六、周日晚上20:30-22:30 报名费：报名费3000一人（周期 "
     <a href="https://www.cnblogs.com/yoyoketang/p/15232435.html" class="c_b_p_desc_readmore">阅读全文</a>
   </div>
 </div>
 <div class="clear"></div>
```

```
element.style {
}
bundle-coff…
#mainContent
.forFlow {
  margin: ▸ 0
    320px 0
    0;
  background:
    ▸
    #fefef2
    url(
    /skins…)
    left top
```

（2）爬虫模块开发

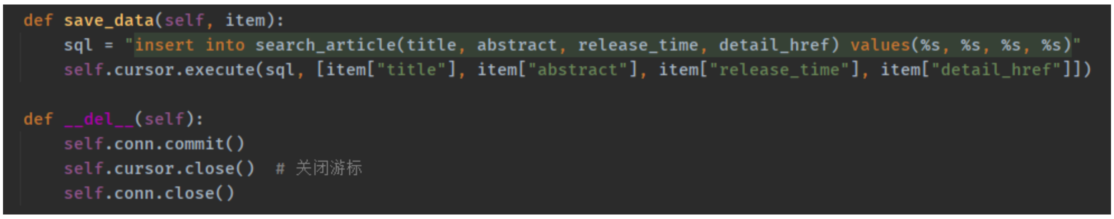
爬虫部分采用 requests 对目标网站发送请求，获取该博主所有博客页数后递归请求每一页数据，并结合 BeautifulSoup 库进行节点解析，提取对应的链接与博客 title 与摘要数据。核心爬虫代码如下：



```python
class ArticleScrawl(object):
    def __init__(self):
        self.conn = pymysql.connect(host='localhost',
                                    user='root',
                                    password='125225',
                                    db='dj_search',
                                    charset='utf8')
        self.cursor = self.conn.cursor()

    def get_bs(self, author, page=1):
        r = requests.get(f'https://www.cnblogs.com/{author}/default.html?page={page}')
        soup = BeautifulSoup(r.content, 'html5lib')
        print(f'第{page}页：')
        self.data_print(soup)
        if soup.select(f'a[href="https://www.cnblogs.com/{author}/default.html?page={page + 1}"]'):
            self.get_bs(author, page + 1)

    def data_print(self, soup):
        for day in soup.select('div.day'):
            for riqi in day.select('div.dayTitle a'):
                for wenzhang in day.select('a.postTitle2'):
                    title = str(wenzhang.text).strip()
                    abstract = str(day.select('div.c_b_p_desc')[0].text).strip()
                    release_time = str(riqi.text).strip()
                    detail_href = str(day.select('a.vertical-middle')[0].get('href')).strip()
                    item = {"title": title, "abstract": abstract, "release_time": release_time, "detail_href": detail_href}
                    print(item)
                    self.save_data(item)
```

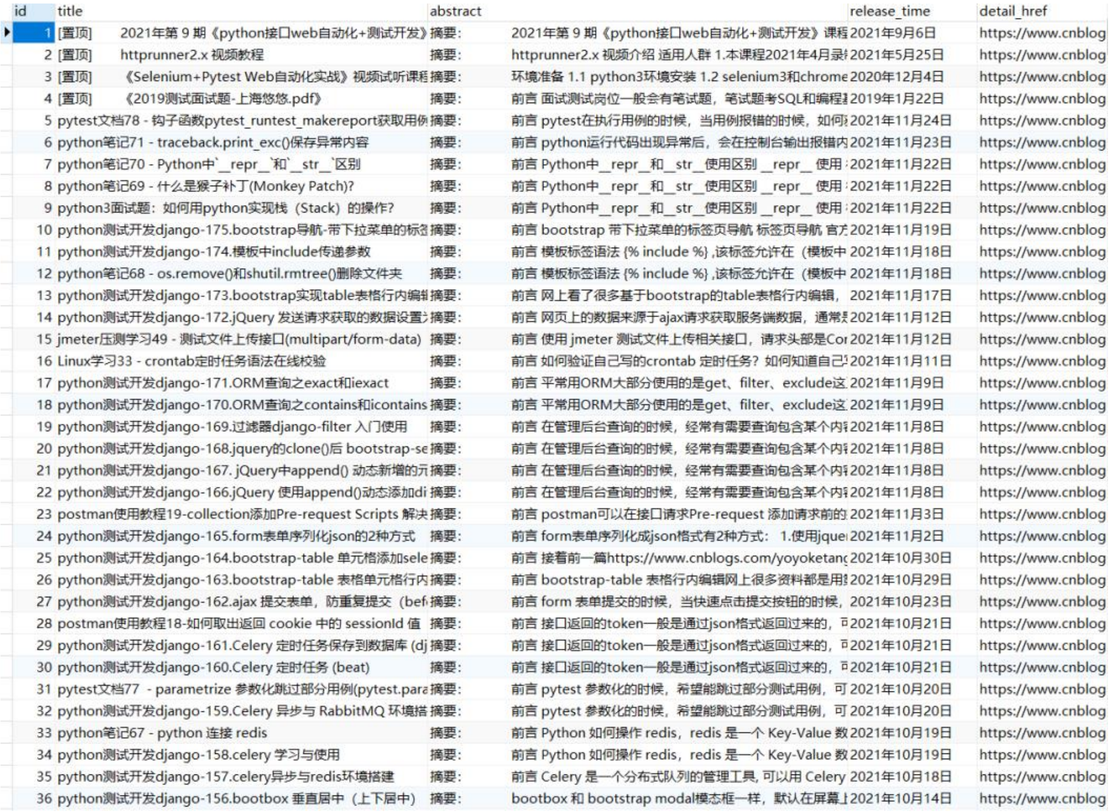
（3）数据持久化

将爬虫爬取到的每篇博客的标题、发布时间、摘要以及详情链接等字段存入本地 mysql 数据库做数据持久化处理。核心代码如下：



```python
def save_data(self, item):
    sql = "insert into search_article(title, abstract, release_time, detail_href) values(%s, %s, %s, %s)"
    self.cursor.execute(sql, [item["title"], item["abstract"], item["release_time"], item["detail_href"]])

def __del__(self):
    self.conn.commit()
    self.cursor.close()  # 关闭游标
    self.conn.close()
```

Mysql 入库数据展示如下：

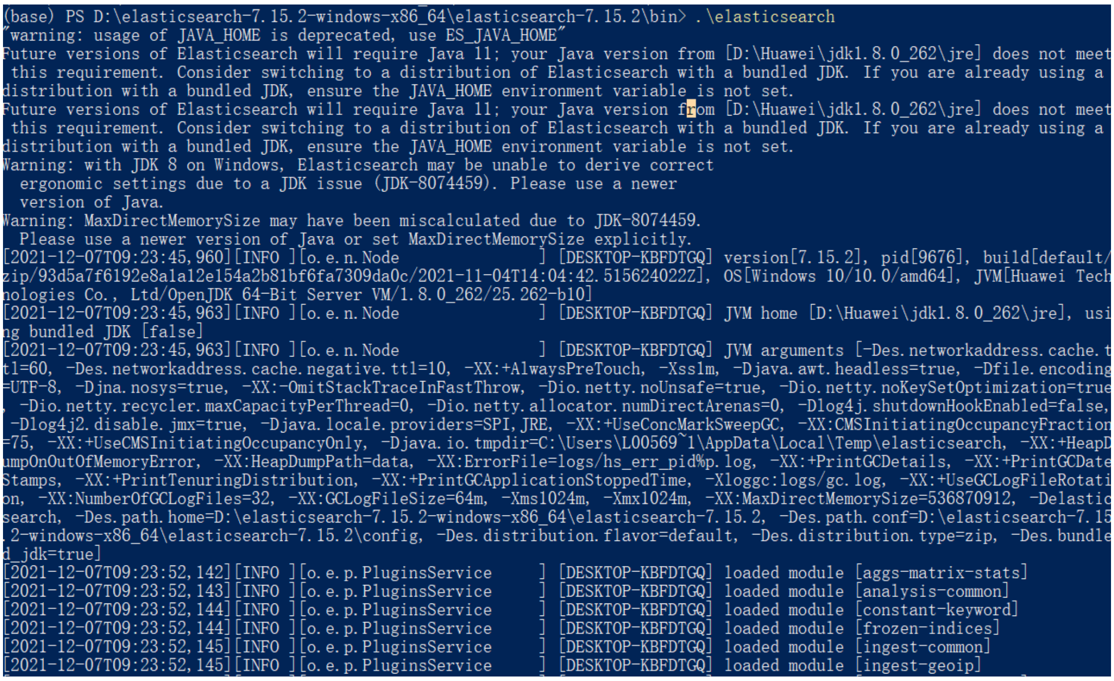


id	title	abstract	release_time	detail_href
1	[置顶] 2021年第 9 期《python接口web自动化+测试开发》	摘要: 2021年第 9 期《python接口web自动化+测试开发》课程	2021年9月6日	https://www.cnblog
2	[置顶] httprunner2.x 视频教程	摘要: httprunner2.x 视频介绍 适用人群 1.本课程2021年4月录	2021年5月25日	https://www.cnblog
3	[置顶] 《Selenium+Pytest Web自动化实战》视频试听课程	摘要: 环境准备 1.1 python3环境安装 1.2 selenium3和chrome	2020年12月4日	https://www.cnblog
4	[置顶] 《2019测试面试题-上海悠悠.pdf》	摘要: 前言 面试测试岗位一般会有笔试题，笔试题考SQL和编程	2019年1月22日	https://www.cnblog
5	pytest文档78 - 钩子函数pytest_runtest_makereport获取用例	摘要: 前言 pytest在执行用例的时候，当用例报错的时候，如何	2021年11月24日	https://www.cnblog
6	python笔记71 - traceback.print_exc()保存异常内容	摘要: 前言 python运行代码出现异常后，会在控制台输出报错内	2021年11月23日	https://www.cnblog
7	python笔记70 - Python中`__repr__`和`__str__`区别	摘要: 前言 Python中__repr__和__str__使用区别 __repr__ 使用	2021年11月22日	https://www.cnblog
8	python笔记69 - 什么是猴子补丁(Monkey Patch)?	摘要: 前言 Python中__repr__和__str__使用区别 __repr__ 使用	2021年11月22日	https://www.cnblog
9	python3面试题：如何用python实现栈（Stack）的操作?	摘要: 前言 Python中__repr__和__str__使用区别 __repr__ 使用	2021年11月22日	https://www.cnblog
10	python测试开发django-175.bootstrap导航-带下拉菜单的标签	摘要: 前言 bootstrap 带下拉菜单的标签页导航 标签页导航 官方	2021年11月19日	https://www.cnblog
11	python测试开发django-174.模板中include传递参数	摘要: 前言 模板标签语法 {% include %} ,该标签允许在（模板中	2021年11月18日	https://www.cnblog
12	python笔记68 - os.remove()和shutil.rmtree()删除文件夹	摘要: 前言 模板标签语法 {% include %} ,该标签允许在（模板中	2021年11月18日	https://www.cnblog
13	python测试开发django-173.bootstrap实现table表格行内编辑	摘要: 前言 网上看了很多基于bootstrap的table表格行内编辑，	2021年11月17日	https://www.cnblog
14	python测试开发django-172.jQuery 发送请求获取的数据设置	摘要: 前言 网页上的数据来源于ajax请求获取服务端数据，通常	2021年11月12日	https://www.cnblog
15	jmeter压测学习49 - 测试文件上传接口(multipart/form-data)	摘要: 前言 使用 jmeter 测试文件上传相关接口，请求头部是Cor	2021年11月12日	https://www.cnblog
16	Linux学习33 - crontab定时任务语法在线校验	摘要: 前言 如何验证自己写的crontab 定时任务？如何知道自己	2021年11月11日	https://www.cnblog
17	python测试开发django-171.ORM查询之exact和iexact	摘要: 前言 平常用ORM大部分使用的是get、filter、exclude这	2021年11月9日	https://www.cnblog
18	python测试开发django-170.ORM查询之contains和icontains	摘要: 前言 平常用ORM大部分使用的是get、filter、exclude这	2021年11月9日	https://www.cnblog
19	python测试开发django-169.过滤器django-filter 入门使用	摘要: 前言 在管理后台查询的时候，经常有需要查询包含某个内	2021年11月8日	https://www.cnblog
20	python测试开发django-168.jquery的clone()后 bootstrap-se	摘要: 前言 在管理后台查询的时候，经常有需要查询包含某个内	2021年11月8日	https://www.cnblog
21	python测试开发django-167. jQuery中append() 动态新增的元	摘要: 前言 在管理后台查询的时候，经常有需要查询包含某个内	2021年11月8日	https://www.cnblog
22	python测试开发django-166.jQuery 使用append()动态添加di	摘要: 前言 在管理后台查询的时候，经常有需要查询包含某个内	2021年11月8日	https://www.cnblog
23	postman使用教程19-collection添加Pre-request Scripts 解决	摘要: 前言 postman可以在接口请求Pre-request 添加请求前的	2021年11月3日	https://www.cnblog
24	python测试开发django-165.form表单序列化json的2种方式	摘要: 前言 form表单序列化成json格式有2种方式： 1.使用jque	2021年11月2日	https://www.cnblog
25	python测试开发django-164.bootstrap-table 单元格添加sele	摘要: 前言 接着前一篇https://www.cnblogs.com/yoyoketang	2021年10月30日	https://www.cnblog
26	python测试开发django-163.bootstrap-table 表格单元格行内	摘要: 前言 bootstrap-table 表格行内编辑网上很多资料都是用	2021年10月29日	https://www.cnblog
27	python测试开发django-162.ajax 提交表单，防重复提交（bef	摘要: 前言 form 表单提交的时候，当快速点击提交按钮的时候，	2021年10月23日	https://www.cnblog
28	postman使用教程18-如何取出返回 cookie 中的 sessionId 值	摘要: 前言 接口返回的token一般是通过json格式返回过来的，可	2021年10月21日	https://www.cnblog
29	python测试开发django-161.Celery 定时任务保存到数据库 (dj	摘要: 前言 接口返回的token一般是通过json格式返回过来的，可	2021年10月21日	https://www.cnblog
30	python测试开发django-160.Celery 定时任务 (beat)	摘要: 前言 接口返回的token一般是通过json格式返回过来的，可	2021年10月21日	https://www.cnblog
31	pytest文档77 - parametrize 参数化跳过部分用例(pytest.para	摘要: 前言 pytest 参数化的时候，希望能跳过部分测试用例，可	2021年10月20日	https://www.cnblog
32	python测试开发django-159.Celery 异步与 RabbitMQ 环境搭	摘要: 前言 pytest 参数化的时候，希望能跳过部分测试用例，可	2021年10月20日	https://www.cnblog
33	python笔记67 - python 连接 redis	摘要: 前言 Python 如何操作 redis，redis 是一个 Key-Value 数	2021年10月19日	https://www.cnblog
34	python测试开发django-158.celery 学习与使用	摘要: 前言 Python 如何操作 redis，redis 是一个 Key-Value 数	2021年10月19日	https://www.cnblog
35	python测试开发django-157.celery异步与redis环境搭建	摘要: 前言 Celery 是一个分布式队列的管理工具, 可以用 Celery	2021年10月18日	https://www.cnblog
36	python测试开发django-156.bootbox 垂直居中（上下居中）	摘要: bootbox 和 bootstrap modal模态框一样，默认在屏幕上	2021年10月14日	https://www.cnblog

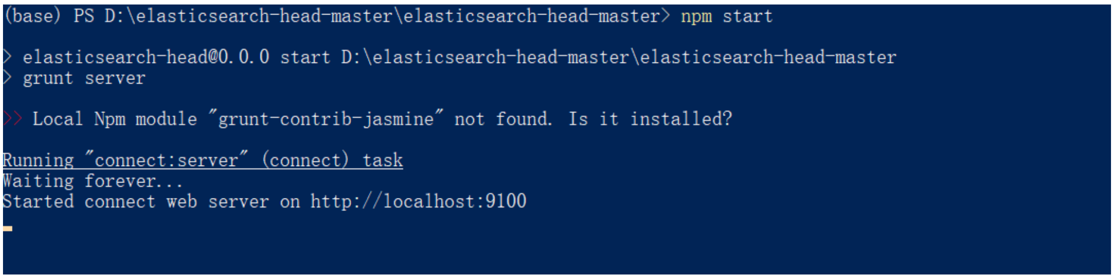
胡泷文：主要负责全文检索系统的前后端设计与开发、Mysql 与 es 的数据同步

（1）首先启动本地 es 环境

先启动 es 服务，然后启动 es-head 服务运行截图如下：

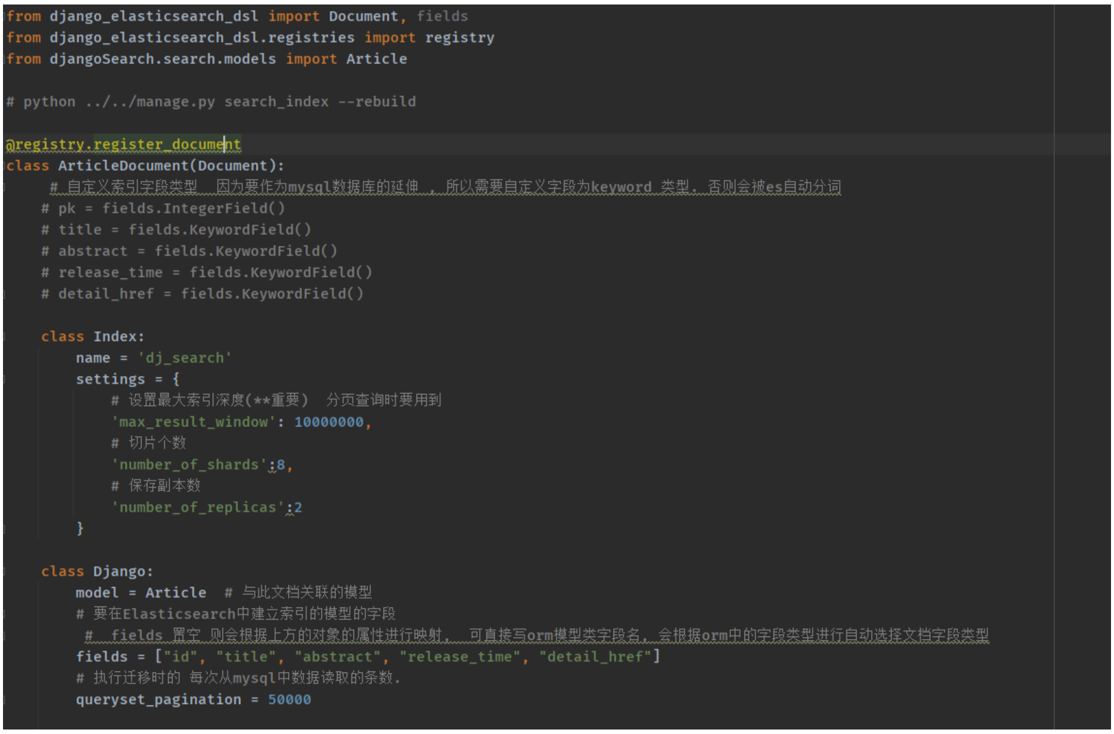


```
(base) PS D:\elasticsearch-7.15.2-windows-x86_64\elasticsearch-7.15.2\bin> .\elasticsearch
"warning: usage of JAVA_HOME is deprecated, use ES_JAVA_HOME"
Future versions of Elasticsearch will require Java 11; your Java version from [D:\Huawei\jdk1.8.0_262\jre] does not meet
 this requirement. Consider switching to a distribution of Elasticsearch with a bundled JDK. If you are already using a
distribution with a bundled JDK, ensure the JAVA_HOME environment variable is not set.
Future versions of Elasticsearch will require Java 11; your Java version from [D:\Huawei\jdk1.8.0_262\jre] does not meet
 this requirement. Consider switching to a distribution of Elasticsearch with a bundled JDK. If you are already using a
distribution with a bundled JDK, ensure the JAVA_HOME environment variable is not set.
Warning: with JDK 8 on Windows, Elasticsearch may be unable to derive correct
  ergonomic settings due to a JDK issue (JDK-8074459). Please use a newer
  version of Java.
Warning: MaxDirectMemorySize may have been miscalculated due to JDK-8074459.
  Please use a newer version of Java or set MaxDirectMemorySize explicitly.
[2021-12-07T09:23:45,960][INFO ][o.e.n.Node               ] [DESKTOP-KBFDTGQ] version[7.15.2], pid[9676], build[default/
zip/93d5a7f6192e8a1a12e154a2b81bf6fa7309da0c/2021-11-04T14:04:42.515624022Z], OS[Windows 10/10.0/amd64], JVM[Huawei Tech
nologies Co., Ltd/OpenJDK 64-Bit Server VM/1.8.0_262/25.262-b10]
[2021-12-07T09:23:45,963][INFO ][o.e.n.Node               ] [DESKTOP-KBFDTGQ] JVM home [D:\Huawei\jdk1.8.0_262\jre], usi
ng bundled JDK [false]
[2021-12-07T09:23:45,963][INFO ][o.e.n.Node               ] [DESKTOP-KBFDTGQ] JVM arguments [-Des.networkaddress.cache.t
tl=60, -Des.networkaddress.cache.negative.ttl=10, -XX:+AlwaysPreTouch, -Xss1m, -Djava.awt.headless=true, -Dfile.encoding
=UTF-8, -Djna.nosys=true, -XX:-OmitStackTraceInFastThrow, -Dio.netty.noUnsafe=true, -Dio.netty.noKeySetOptimization=true
, -Dio.netty.recycler.maxCapacityPerThread=0, -Dio.netty.allocator.numDirectArenas=0, -Dlog4j.shutdownHookEnabled=false,
 -Dlog4j2.disable.jmx=true, -Djava.locale.providers=SPI,JRE, -XX:+UseConcMarkSweepGC, -XX:CMSInitiatingOccupancyFraction
=75, -XX:+UseCMSInitiatingOccupancyOnly, -Djava.io.tmpdir=C:\Users\L00569~1\AppData\Local\Temp\elasticsearch, -XX:+HeapD
umpOnOutOfMemoryError, -XX:HeapDumpPath=data, -XX:ErrorFile=logs/hs_err_pid%p.log, -XX:+PrintGCDetails, -XX:+PrintGCDate
Stamps, -XX:+PrintTenuringDistribution, -XX:+PrintGCApplicationStoppedTime, -Xloggc:logs/gc.log, -XX:+UseGCLogFileRotati
on, -XX:NumberOfGCLogFiles=32, -XX:GCLogFileSize=64m, -Xms1024m, -Xmx1024m, -XX:MaxDirectMemorySize=536870912, -Delastic
search, -Des.path.home=D:\elasticsearch-7.15.2-windows-x86_64\elasticsearch-7.15.2, -Des.path.conf=D:\elasticsearch-7.15
.2-windows-x86_64\elasticsearch-7.15.2\config, -Des.distribution.flavor=default, -Des.distribution.type=zip, -Des.bundle
d_jdk=true]
[2021-12-07T09:23:52,142][INFO ][o.e.p.PluginsService     ] [DESKTOP-KBFDTGQ] loaded module [aggs-matrix-stats]
[2021-12-07T09:23:52,143][INFO ][o.e.p.PluginsService     ] [DESKTOP-KBFDTGQ] loaded module [analysis-common]
[2021-12-07T09:23:52,144][INFO ][o.e.p.PluginsService     ] [DESKTOP-KBFDTGQ] loaded module [constant-keyword]
[2021-12-07T09:23:52,144][INFO ][o.e.p.PluginsService     ] [DESKTOP-KBFDTGQ] loaded module [frozen-indices]
[2021-12-07T09:23:52,145][INFO ][o.e.p.PluginsService     ] [DESKTOP-KBFDTGQ] loaded module [ingest-common]
[2021-12-07T09:23:52,145][INFO ][o.e.p.PluginsService     ] [DESKTOP-KBFDTGQ] loaded module [ingest-geoip]
```



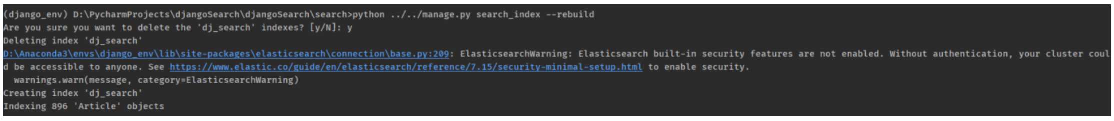
```
(base) PS D:\elasticsearch-head-master\elasticsearch-head-master> npm start

> elasticsearch-head@0.0.0 start D:\elasticsearch-head-master\elasticsearch-head-master
> grunt server

>> Local Npm module "grunt-contrib-jasmine" not found. Is it installed?

Running "connect:server" (connect) task
Waiting forever...
Started connect web server on http://localhost:9100
```

（2）搭建 Django 项目，编写配置文件 settings.py，将数据库，es 等第三方服务接口导入并配置对应驱动与相应的端口号，然后根据 mysql 表结构编写博客对应的实体类，与 es 的索引类，部分核心代码如下：



```python
from django_elasticsearch_dsl import Document, fields
from django_elasticsearch_dsl.registries import registry
from djangoSearch.search.models import Article

# python ../../manage.py search_index --rebuild

@registry.register_document
class ArticleDocument(Document):
     # 自定义索引字段类型  因为要作为mysql数据库的延伸 , 所以需要自定义字段为keyword 类型, 否则会被es自动分词
    # pk = fields.IntegerField()
    # title = fields.KeywordField()
    # abstract = fields.KeywordField()
    # release_time = fields.KeywordField()
    # detail_href = fields.KeywordField()

    class Index:
        name = 'dj_search'
        settings = {
            # 设置最大索引深度(**重要)  分页查询时要用到
            'max_result_window': 10000000,
            # 切片个数
            'number_of_shards':8,
            # 保存副本数
            'number_of_replicas':2
        }

    class Django:
        model = Article  # 与此文档关联的模型
        # 要在Elasticsearch中建立索引的模型的字段
         #  fields 置空 则会根据上方的对象的属性进行映射,  可直接写orm模型类字段名, 会根据orm中的字段类型进行自动选择文档字段类型
        fields = ["id", "title", "abstract", "release_time", "detail_href"]
        # 执行迁移时的 每次从mysql中数据读取的条数.
        queryset_pagination = 50000
```

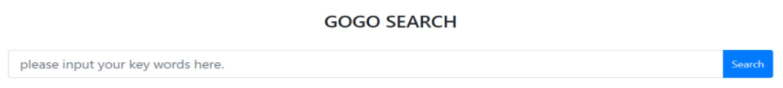
（3）命令行输入命令，完成 mysql 与 es 的数据迁移



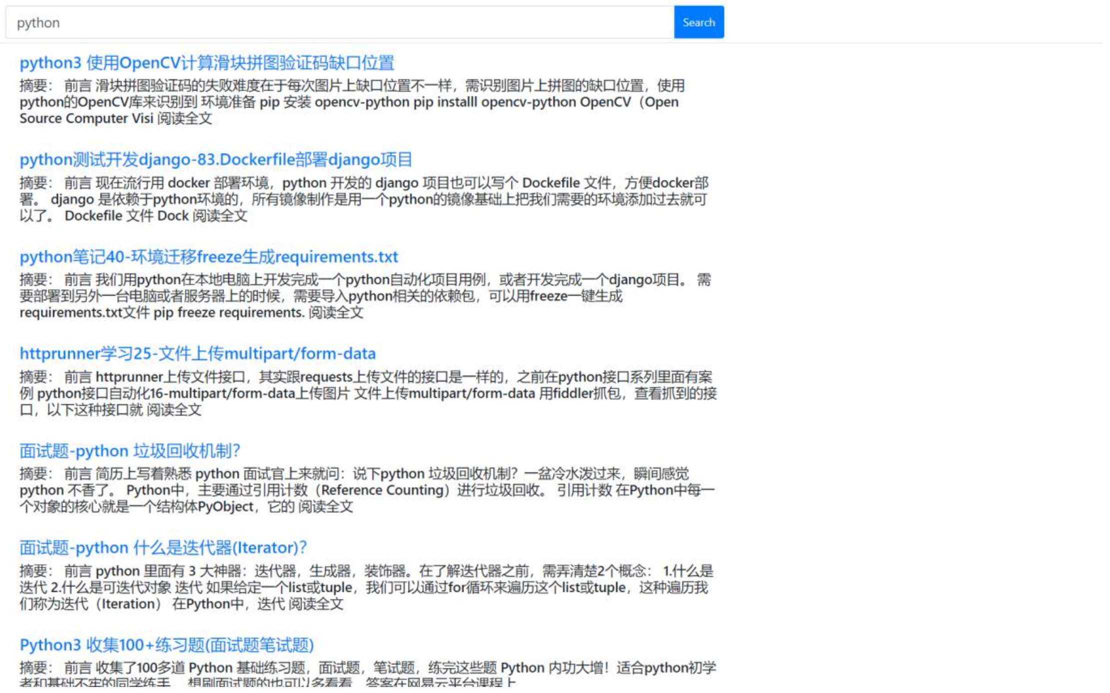
```
(django_env) D:\PycharmProjects\djangoSearch\djangoSearch\search>python ../../manage.py search_index --rebuild
Are you sure you want to delete the 'dj_search' indexes? [y/N]: y
Deleting index 'dj_search'
D:\Anaconda3\envs\django_env\lib\site-packages\elasticsearch\connection\base.py:209: ElasticsearchWarning: Elasticsearch built-in security features are not enabled. Without authentication, your cluster coul
d be accessible to anyone. See https://www.elastic.co/guide/en/elasticsearch/reference/7.15/security-minimal-setup.html to enable security.
  warnings.warn(message, category=ElasticsearchWarning)
Creating index 'dj_search'
Indexing 896 'Article' objects
```

（4）编写前端页面与后端业务逻辑接口，完成用户交互全文检索功能。

五、实验结果展示



GOGO SEARCH



python3 使用OpenCV计算滑块拼图验证码缺口位置

摘要： 前言 滑块拼图验证码的失败难度在于每次图片上缺口位置不一样，需识别图片上拼图的缺口位置，使用python的OpenCV库来识别到 环境准备 pip 安装 opencv-python pip installl opencv-python OpenCV（Open Source Computer Visi 阅读全文

python测试开发django-83.Dockerfile部署django项目

摘要： 前言 现在流行用 docker 部署环境，python 开发的 django 项目也可以写个 Dockefile 文件，方便docker部署。 django 是依赖于python环境的，所有镜像制作是用一个python的镜像基础上把我们需要的环境添加过去就可以了。 Dockefile 文件 Dock 阅读全文

python笔记40-环境迁移freeze生成requirements.txt

摘要： 前言 我们用python在本地电脑上开发完成一个python自动化项目用例，或者开发完成一个django项目。 需要部署到另外一台电脑或者服务器上的时候，需要导入python相关的依赖包，可以用freeze一键生成requirements.txt文件 pip freeze requirements. 阅读全文

httprunner学习25-文件上传multipart/form-data

摘要： 前言 httprunner上传文件接口，其实跟requests上传文件的接口是一样的，之前在python接口系列里面有案例 python接口自动化16-multipart/form-data上传图片 文件上传multipart/form-data 用fiddler抓包，查看抓到的接口，以下这种接口就 阅读全文

面试题-python 垃圾回收机制?

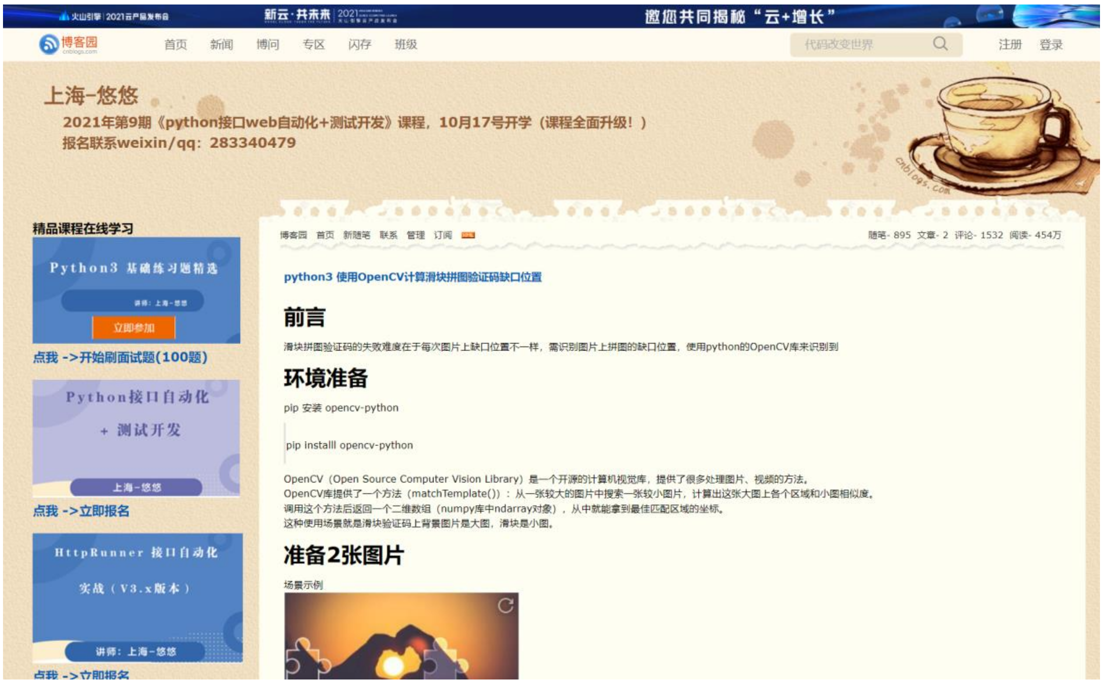
摘要： 前言 简历上写着熟悉 python 面试官上来就问：说下python 垃圾回收机制？一盆冷水泼过来，瞬间感觉python 不香了。 Python中，主要通过引用计数（Reference Counting）进行垃圾回收。 引用计数 在Python中每一个对象的核心就是一个结构体PyObject，它的 阅读全文

面试题-python 什么是迭代器(Iterator)?

摘要： 前言 python 里面有 3 大神器：迭代器，生成器，装饰器。在了解迭代器之前，需弄清楚2个概念： 1.什么是迭代 2.什么是可迭代对象 迭代 如果给定一个list或tuple，我们可以通过for循环来遍历这个list或tuple，这种遍历我们称为迭代（Iteration） 在Python中，迭代 阅读全文

Python3 收集100+练习题(面试题笔试题)

摘要： 前言 收集了100多道 Python 基础练习题，面试题，笔试题，练完这些题 Python 内功大增！适合python初学者和基础不牢的同学练手，想刷面试题的也可以多看看，答案在网易云平台课程上



上海-悠悠

2021年第9期《python接口web自动化+测试开发》课程，10月17号开学（课程全面升级！）
报名联系weixin/qq：283340479

随笔- 895 文章- 2 评论- 1532 阅读- 454万

python3 使用OpenCV计算滑块拼图验证码缺口位置

前言

滑块拼图验证码的失败难度在于每次图片上缺口位置不一样，需识别图片上拼图的缺口位置，使用python的OpenCV库来识别到

环境准备

pip 安装 opencv-python

```
pip installl opencv-python
```

OpenCV（Open Source Computer Vision Library）是一个开源的计算机视觉库，提供了很多处理图片、视频的方法。
OpenCV库提供了一个方法（matchTemplate()）：从一张较大的图片中搜索一张较小图片，计算出这张大图上各个区域和小图相似度。
调用这个方法后返回一个二维数组（numpy库中ndarray对象），从中就能拿到最佳匹配区域的坐标。
这种使用场景就是滑块验证码上背景图片是大图，滑块是小图。

准备2张图片

场景示例

六、总结

本次实验涉及知识点较多，包含了 django 的前后端开发、爬虫模块编写、mysql 的持久化操作、ElasticSearch 的数据迁移与全文检索，通过完成整个项目的开发，对所学知识有了更加深刻的理解与应用。