**数据采集源码说明**

1. **说明**
2. **环境及用到的第三方框架或工具：**

jdk1.8， eclipse Kepler， windows 10， mysql 5.7.20，

程序采用spring4.0管理主要实例以及hibernate事务，采用hibernate4.2进行数据持久化，采用c3p数据库连接池管理与数据库的连接，采用jsoup进行网页分析；采用json-lib进行json数据解析;

1. **数据来源：**

航班数据来源于<https://flightaware.com/>

机场所在地数据来源于百度经纬度查询api

机场名数据来源于<http://airport.anseo.cn/>

代理ip爬取数据来源于<http://www.xicidaili.com/wn/>

天气数据来源于<http://www.tianqihoubao.com> （新增部分）｛由于国外的地区名字与先前的数据库中的地名匹配不上，所以只爬取了国内的天气数据！｝

1. **相关配置及其说明：**

数据库信息，以及爬虫线程相关修改在com/drsn/flights/config/applicationContext.xml中配置

航空公司ICAO码爬取，航班号爬取，具体航班信息爬取，代理ip爬取这4个线程的启动个数可 以在com/drsn/flights/config/applicationContext.xml中的cfgBean处设置，但 航空公司ICAO码爬取和代理ip爬取这两个线程是同时进行的，只需要设置航空公司ICAO码爬取的线程数即可，并且只能是1个线程或者0个，具体航班信息爬取，代理从数据库获取这两个线程也是同时进行的，只需要设置具体航班信息爬取的线程数即可，但只有后者只能是1个线程或者0个，建议用两台电脑，一台航空公司ICAO码爬取的线程数设置为1，航班号爬取线程数设置200-500或根据配置设置，另一台电脑具体航班信息爬取线程数设置200-500或根据配置设置，注意：要连接同一个数据库;

1. **程序主要思路**

本程序主要分为航空公司ICAO码爬取，航班号爬取，具体航班信息爬取，代理ip爬取这4个线程，并交由一个线程管理器进行维护，启动关闭各个线程，检测线程是否异常中止，然后重启异常中止的线程。

航班信息数据采集步骤：

1. 程序先是航空公司ICAO码爬取线程循环从https://zh.flightaware.com/live/fleet/获取新出现的航空公司ICAO码，
2. 然后是航班号爬取线程去根据航空公司ICAO码去https://zh.flightaware.com/live/fleet/（此处加上航空公司ICAO码）循环爬取新出现的航班号，
3. 然后再是具体航班信息爬取线程根据航班号去https://zh.flightaware.com/live/flight/（此处是航班号）/history/2000，爬取一个包含历史日期的具体航班信息的那一页的网页地址集合，然后筛选出符合设定的要爬取的时间范围的数据，该页需要登录的cookie身份凭证，
4. 然后程序根据先前那个网页地址集合爬取具体航班信息并调用hibernate持久化数据，

**注：**

* 1. 最初程序中有100条身份凭证cookie，若中途凭证时效，程序将会调用注册用户的方法去flightaware网站上去注册一个帐号并替换掉原来的失效的条身份凭证cookie。
  2. 航空公司ICAO码爬取，航班号爬取，具体航班信息爬取三个模块都会用到代理IP爬取线程爬取到的代理ip,由于学校电信宽带登录后容易断网，所以整个程序也会运行校园网断网重登模块来保证网络连通。

数据爬完后，调用com.drsn.flights.other包中的机场所在地信息更新模块，根据经纬度从百度经纬度查询api获取所在地的国家，省份，市区信息，并更新至数据库；调用机场名更新模块，根据机场ICAO码从安能全球机场代码http://airport.anseo.cn/网站上爬取机场名信息，并更新至数据库;

**新增天气数据模块：**

1. 先在http://www.tianqihoubao.com/lishi/网页爬取国内地名与其拼音的对照，
2. 然后再根据拼音到http://www.tianqihoubao.com/lishi/{拼音}/month/｛日期，如201802｝.html爬取一个月的天气数据
3. **程序设计时遇到的部分问题及其对应的解决方案**

|  |  |
| --- | --- |
| 问题 | 解决方法 |
| 校园网断网问题 | 编写程序实现周期性检测网络是否连通，是否需要重登录宽带; |
| 爬取数据时ip被封杀问题 | 添加代理ip爬取线程进行实时爬取网络上的可用代理IP并周期性进行验证其有效性; |
| 爬取数据时flightaware帐号被封杀问题 | 添加帐号随机注册模块，在账号失效时自动调用注册新账号; |
| 程序运行时出现线程会意外中止问题 | 编写线程管理器进行线程维护，重启意外中止的线程并恢复其进度; |

1. **源码包结构及其功能**

* /src
* com.drsn.flights
  + - * FlightsMain 数据采集程序启动主入口
* com.drsn.flights.bean
  + - * AirlineBean 航空公司信息实体类
      * AirportBean 机场信息实体类
      * CfgBean 存放该程序相关配置数据的Bean
      * FleetBean 航班号实体类
      * FlightBean 具体航班信息实体类
      * PrefixBean 航空公司ICAO码实体类
      * ProxyBean 代理ip实体类
      * SpringConfig 用来存放spring的ApplicationContext的类
      * AirlineBean.hbm.xml hibernate 映射文件，实体类：AirlineBean，数据库中的表：airline
      * AirportBean.hbm.xml hibernate 映射文件，实体类：AirportBean，数据库中的表：airport
      * FleetBean.hbm.xml hibernate 映射文件，实体类：FleetBean，数据库中的表：fleet
      * FlightBean.hbm.xml hibernate 映射文件，实体类：FlightBean，数据库中的表：flight
      * PrefixBean.hbm.xml hibernate 映射文件，实体类：PrefixBean，数据库中的表：prefix
      * ProxyBean.hbm.xml hibernate 映射文件，实体类：ProxyBean，数据库中的表：proxy
* com.drsn.flights.dao
  + - * AirlineDao 航空公司信息实体类的DAO操作
      * AirportDao 机场信息实体类的DAO操作
      * Daobasics 该程序中DAO类的基础类，包含SessionFactory和Session
      * FleetDao 航班号实体类的DAO操作
      * FlightDao 具体航班信息实体类的DAO操作
      * PrefixDao 航空公司ICAO码实体类的DAO操作
      * ProxyDao 代理ip实体类的DAO操作
* com.drsn.flights.task
  + - * FleetCollectTask 航班号爬取线程
      * FlightsCollectTask 具体航班信息爬取线程
      * PrefixCollectTask 航空公司ICAO码爬取线程
      * ProxyTask 代理IP爬取线程（从西刺代理获取）
      * ProxyFromSqlTask 代理IP获取线程（从数据库中获取）
      * TaskBasics 该程序的线程基础类，内含一些基本定义
      * TaskManager 该程序的线程管理器
* com.drsn.flights.other
  + - * AirportLocationCrawl 机场所在地数据更新模块（数据来源于百度api），有main方法，
      * AirportNameCrawl 机场名数据更新模块（数据来源于安能全球机场代码），有main方法，
      * CloseCmdSend 用于关闭该程序，
      * FlightawareRegister flightaware用户注册模块
      * HeavyLogin 校园网断网重登模块，有main方法，
* /config
  + - applicationContext.xml spring，数据库，程序初始化相关配置
    - hibernate.cfg.xml hibernate相关配置
* /lib/\* 该程序所用到的jar包

**新增天气数据模块：**

* /src
  + - com.test
      * DaoBases.java DAO父类
      * TestA.java 程序入口，处理该程序主要逻辑
      * TianqiBean.java 与数据库中tianqi表对应的实体类
      * TianqiBean.hbm.xml 映射文件，实体类：TianqiBean，数据库中的表：tianqi
    - hibernate.cfg.xml hibernate相关配置