链滴

Java 爬取美女图片妹子图（mzitu）

作者：724555508

原文链接：https://ld246.com/article/1597458193176

来源网站：链滴

1.准备图片网站 https://www.mzitu.com/

2.本篇是爬取妹子图网站全站图片的代码，采用 Java 语言编写，另外妹子图防爬措施，本篇采用极光 ip 客户端自动切换 ip 实现，考虑切换 ip 时会有访

异常，代码上相关下载处做了无限重试处理。

```
<!--相关依赖-->
<dependency>
        <groupId>org.jsoup</groupId>
        <artifactId>jsoup</artifactId>
        <version>1.11.3</version>
</dependency>
<dependency>
    <groupId>cn.hutool</groupId>
    <artifactId>hutool-all</artifactId>
    <version>5.1.0</version>
</dependency>
```

3.爬取全站图片时，妹子图分为两个地址

https://www.mzitu.com/all/

https://www.mzitu.com/old/

4.直接上代码了，文末有可直接下载的.java 文件

```
package com.Yang;

import java.io.File;
import java.util.UUID;
import java.util.concurrent.ExecutorService;
import java.util.concurrent.LinkedBlockingQueue;
import java.util.concurrent.ThreadPoolExecutor;
import java.util.concurrent.TimeUnit;

import org.jsoup.Jsoup;
import org.jsoup.nodes.Document;
import org.jsoup.nodes.Element;
import org.jsoup.select.Elements;
import org.junit.runner.RunWith;
import org.springframework.boot.test.context.SpringBootTest;
import org.springframework.test.context.junit4.SpringJUnit4ClassRunner;

import cn.hutool.core.thread.ThreadUtil;
import cn.hutool.http.HttpRequest;
import cn.hutool.http.HttpResponse;
import cn.hutool.http.HttpUtil;
import lombok.extern.slf4j.Slf4j;

@RunWith(SpringJUnit4ClassRunner.class)
@SpringBootTest
@Slf4j
public class Meizitu {

    static String FileDir = "D:\\mzitu\\";//图片保存位置
    static String UserAgent = "Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (K
TML, like Gecko) Chrome/53.0.2785.104 Safari/537.36 Core/1.53.2595.400 QQBrowser/9.6.108
```

```
2.400";
    static int queueMaxSize = 3;//同时处理最大队列数量
    static LinkedBlockingQueue<Runnable> queue = new LinkedBlockingQueue<Runnable>(q
eueMaxSize);
    static ExecutorService threadPool = new ThreadPoolExecutor(queueMaxSize, queueMaxSize
 0L, TimeUnit.MILLISECONDS,queue);

    // 目的：传递一个url（http的地址），返回对应地址下的HTML的文档
    public static String getHtml(String url){

        HttpRequest createGet = HttpUtil.createGet(url);
        createGet.header("User-Agent", UserAgent)
                .timeout(10000);

        HttpResponse response;
        while(true) {
            try {
                response = createGet.execute();
                break;
            } catch (Exception e) {

            }
        }
        int status = response.getStatus();
        log.info("[{}] , 返回状态码:{}" , url , status);
        if(status == 301 || status == 302) {
            HttpRequest createGet2 = HttpUtil.createGet(response.header("location"));
            createGet2.header("User-Agent", UserAgent)
                    .timeout(-1);
            HttpResponse response2 = createGet2.execute();
            log.info("[{}] , 返回状态码:{}" , response.header("location") , response2.getStatus());
            return response2.body();
        }
        if(status == 404) {
            return "";
        }
        return response.body();
    }

    public static void downLoadImg(String url,String dir){
        HttpRequest createGet = HttpUtil.createGet(url);
        createGet.header("User-Agent", UserAgent);
        createGet.header("Referer","https://www.mzitu.com/");
        createGet.timeout(3000);
        HttpResponse response = createGet.execute();
        log.info("[{}] , 返回状态码:{}" , url , response.getStatus());
        if(response.getStatus() == 429) {
            log.info("频率过快， 两秒后重试");
            ThreadUtil.sleep(2000);
            downLoadImg(url,dir);
        }
    if(response.getStatus() == 404) {
            return;
        }
```

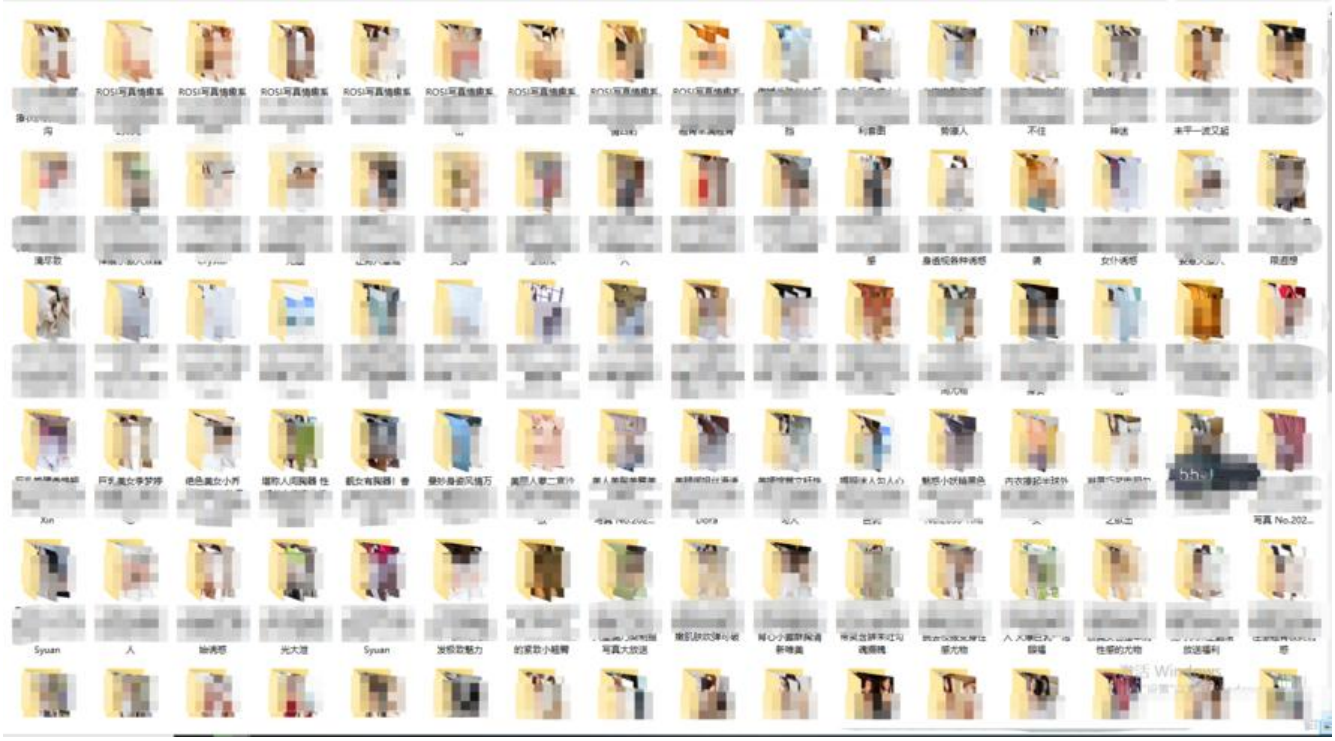
```
        String ext = url.substring(url.lastIndexOf("."));
        String img = "";
        img = UUID.randomUUID().toString()+ext;
     // 先建文件夹
        File file = new File(FileDir + dir.replaceAll("\\:", "").replaceAll("\\?", "").replaceAll("\"", "") +
"\\");
        synchronized (file) {
            if (!file.exists()){
                file.mkdirs();
            }
        }
        response.writeBody(new File(FileDir + dir.replaceAll("\\:", "").replaceAll("\\?", "").replaceAl
("\"", "") + "\\" + img));
    }



    public static void submit(int index ,int page, String url , String dir) {
        String url_cp = url;
        String dir_cp = dir;
        while(true) {
            ThreadUtil.sleep(20);
            if(queue.size() < queueMaxSize) {

                threadPool.execute(new Runnable() {

                    @Override
                    public void run() {
                        int count = 0 ;

                        while(true) {
                            try {
                                downLoadImg(url_cp,dir_cp);
                                return;
                            } catch (Exception e) {
                                count++;
                                log.error("第" + index + "条,page:" + page +"下载失败" + count + "次" , e);
                            }
                        }
                    }
                });
                return;
            }
        }
    }

    public static void main(String[] args) {

        // 初始化url
        String url = "https://www.mzitu.com/old/";
//        String url = "https://www.baidu.com";
        // 获取所有页面
        String html = getHtml(url);
        // 解析url
```

```
        Document document = Jsoup.parse(html);
        Elements elements = document.select("[target=_blank]");
        log.info("数据共{}条" , elements.size());
        int index = 0;
        for (Element element : elements) {
            index++;
            log.info("开始执行第{}条" , index);
            String albumUrl = element.attr("href");
            // 遍历解析每一个URL,得到每一个相册的html
            String eachAlblumHtml = getHtml(albumUrl);
            Document eachAlblumDocument = Jsoup.parse(eachAlblumHtml);
            Elements ele_a = eachAlblumDocument.getElementsByClass("pagenavi").get(0).getEl
mentsByTag("a");
            //找到最大页码
            int pageSize = Integer.valueOf(ele_a.get(ele_a.size() -2).text());
            log.info("最大页码:{}" , pageSize);
            //找图片url
            Elements imgElements = eachAlblumDocument.select(".main-image img");
            if (imgElements.size() > 0) {
                String imgSrc = imgElements.get(0).attr("src");
                String baseImg = imgSrc.substring(0, imgSrc.length() - 6);

                for(int page = 1 ; page<pageSize ; page++) {
                    int value = page;
                    try {
                        if(value < 10) {
                            submit(index, value, baseImg + "0" + value + ".jpg", element.text());
                        }else {
                            submit(index, value, baseImg + value + ".jpg", element.text());
                        }
                    } catch (Exception e) {
                        log.error("第" + index + "条,page:" + value +"下载失败######" , e);
                        page--;
                    }

                }
                while(queue.size() != 0) {
                    ThreadUtil.sleep(20);
                }
            }
        }
    }
}
```

5.源码文件下载

Meizitu.7z

6.最后图片下载出来的效果



原文链接：Java 爬取美女图片妹子图（mzitu）