Atitit 雷速比赛数据抓取 爬虫抓取总结

经过测试headless webview不能抓取到数据，只有使用带头模式webview firefox webdriver可以获取到雷速数据

目录

[1. Htmlunit （经测试不行 1](#_Toc25232)

[1.1. phantomjs-2.1.1 经测试抓到部分，但无法抓取数据表格部分 2](#_Toc30000)

[1.2. Webdriver+ff （ok 3](#_Toc22754)

# Htmlunit （经测试不行

Htmlunit2.4老保内存溢出错误，ht2.8无法下载到远程网站html源码

// always mem over gc

**private** **static** **void** htmlunitFail(String url) **throws** IOException {

WebClient webClient = **new** WebClient(BrowserVersion.***CHROME***);// 新建一个模拟谷歌Chrome浏览器的浏览器客户端对象

//

webClient.getOptions().setThrowExceptionOnScriptError(**false**);// 当JS执行出错的时候是否抛出异常, 这里选择不需要

webClient.getOptions().setThrowExceptionOnFailingStatusCode(**false**);// 当HTTP的状态非200时是否抛出异常, 这里选择不需要

webClient.getOptions().setActiveXNative(**false**);

webClient.getOptions().setCssEnabled(**true**);// 是否启用CSS, 因为不需要展现页面, 所以不需要启用

webClient.getOptions().setJavaScriptEnabled(**true**); // 很重要，启用JS

webClient.setAjaxController(**new** NicelyResynchronizingAjaxController());// 很重要，设置支持AJAX

HtmlPage page = **null**;

**try** {

page = webClient.getPage(url);// 尝试加载上面图片例子给出的网页

} **catch** (Exception e) {

e.printStackTrace();

} **finally** {

webClient.close();

}

webClient.waitForBackgroundJavaScript(15000);// 异步JS执行需要耗时,所以这里线程要阻塞30秒,等待异步JS执行结束

String pageXml = page.asXml();// 直接将加载完成的页面转换成xml格式的字符串

// page.asText()

FileUtils.~~write~~(**new** File("d:\\cache\\leisu.htm"), pageXml);

//

////**TODO** 下面的代码就是对字符串的操作了,常规的爬虫操作,用到了比较好用的Jsoup库

//

Document document = Jsoup.*parse*(pageXml);// 获取html文档

document.getElementsByTag("tr");

//List<Element> infoListEle = document.getElementById("feedCardContent").getElementsByAttributeValue("class", "feed-card-item");//获取元素节点等

//infoListEle.forEach(element -> {

// System.out.println(element.getElementsByTag("h2").first().getElementsByTag("a").text());

// System.out.println(element.getElementsByTag("h2").first().getElementsByTag("a").attr("href"));

//});

//}

//

}

## phantomjs-2.1.1 经测试抓到部分，但无法抓取数据表格部分

调用截图api看到数据部分为空

system = require('system')

address = system.args[1];//获得命令行第二个参数 接下来会用到

//console.log('Loading a web page');

**var** page = require('webpage').create();

**var** url = address;

//console.log(url);

page.open(url, **function** (status) {

//Page is loaded!

/\*

if (status !== 'success') {

console.log('Unable to post!');

} else {

//此处的打印，是将结果一流的形式output到java中，java通过InputStream可以获取该输出内容

console.log(page.content);

}

// phantom.exit();

\*/

window.setTimeout(**function** () {

page.render( "webscreenshot.png"); //网页截图

console.log(page.content); //网页html文本

// console.log(page.content);

// page.render(output);

phantom.exit();

}, 30000); // Change timeout as required to allow sufficient time

});

// D:\phantomjs-2.1.1-windows\bin\phantomjs.exe D:\prj\sport-service\kok-sport-service\js\leisu.js <https://live.leisu.com/>

## Webdriver+ff （ok

**package** com.kok.sport.utils.mockdata;

**import** java.io.File;

**import** java.io.IOException;

**import** java.util.Date;

**import** java.util.concurrent.TimeUnit;

**import** org.apache.commons.io.FileUtils;

**import** org.openqa.selenium.WebDriver;

**import** org.openqa.selenium.firefox.FirefoxDriver;

**import** org.openqa.selenium.htmlunit.HtmlUnitDriver;

**import** org.openqa.selenium.ie.InternetExplorerDriver;

//import org.openqa.selenium.remote.DesiredCapabilities;

**import** com.gargoylesoftware.htmlunit.BrowserVersion;

**public** **class** WebdriverFirefox {

**public** **static** **void** main(String[] args) **throws** Exception {

// declaration and instantiation of objects/variables

System.*setProperty*("webdriver.gecko.driver", "D:\\prj\\spdjs\\node\_modules\\geckodriver\\geckodriver.exe");

WebDriver driver = **new** FirefoxDriver();

// org.openqa.selenium.remote.s

// DesiredCapabilities IEcaps = DesiredCapabilities.htmlUnit();

//INTRODUCE\_FLAKINESS\_BY\_IGNORING\_SECURITY\_DOMAINS

// IEcaps .setCapability(HtmlUnitDriver.INVALIDSELECTIONERROR,true);

// WebDriver driver = new HtmlUnitDriver(BrowserVersion.FIREFOX);

// Set implicit wait

driver.manage().timeouts().implicitlyWait(25, TimeUnit.***SECONDS***);

// comment the above 2 lines and uncomment below 2 lines to use Chrome

System.*setProperty*("webdriver.chrome.driver",

"D:\\prj\\spdjs\\node\_modules\\chromedriver\\lib\\chromedriver\\chromedriver.exe");

// WebDriver driver =new ChromeDriver();

// InternetExplorerDriver lang.UnsatisfiedLinkError:

// C:\Users\Administrator\AppData\Local\Temp\webdriver3771754343790130835.tmp:

// Can't load IA 32-bit .dll on a AMD 64-bit platform

// org.openqa.selenium.ie.InternetExplorerDriver.class

// new ChromeDriver();

String baseUrl = "https://live.leisu.com/";

String expectedTitle = "Welcome: Mercury Tours";

String actualTitle = "";

// launch Fire fox and direct it to the Base URL

driver.get(baseUrl);

// get the actual value of the title

actualTitle = driver.getTitle();

/\*

\* compare the actual title of the page with the expected one and print the

\* result as "Passed" or "Failed"

\*/

**if** (actualTitle.contentEquals(expectedTitle)) {

System.***out***.println("Test Passed!");

} **else** {

System.***out***.println("Test Failed");

}

Thread.*sleep*(12000);

**while** (**true**) {

String pageSource = driver.getPageSource();

Leisu.*processTime*(pageSource);

FileUtils.write(**new** File("d:\\cache\\leisu.htm"), pageSource);

Thread.*sleep*(7000);

System.***out***.println(**new** Date());

}

// close Fire fox

// driver.close();

// System.out.println("f");

}

}