****

**软件测试技术第二次实验报告**



**学 院 智能与计算学部**

**专 业 软件工程**

**年 级 16**

**姓 名 李沛霖**

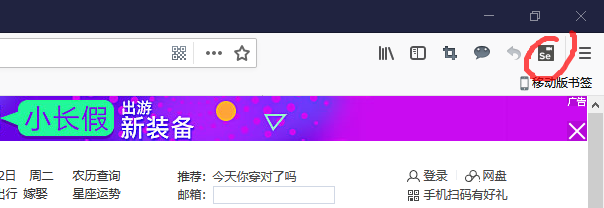
**2019年 3 月 30 日**

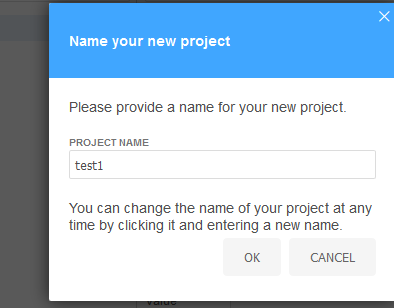
# 软件测试技术第二次实验报告

1. 需求分析（描述具体需求）

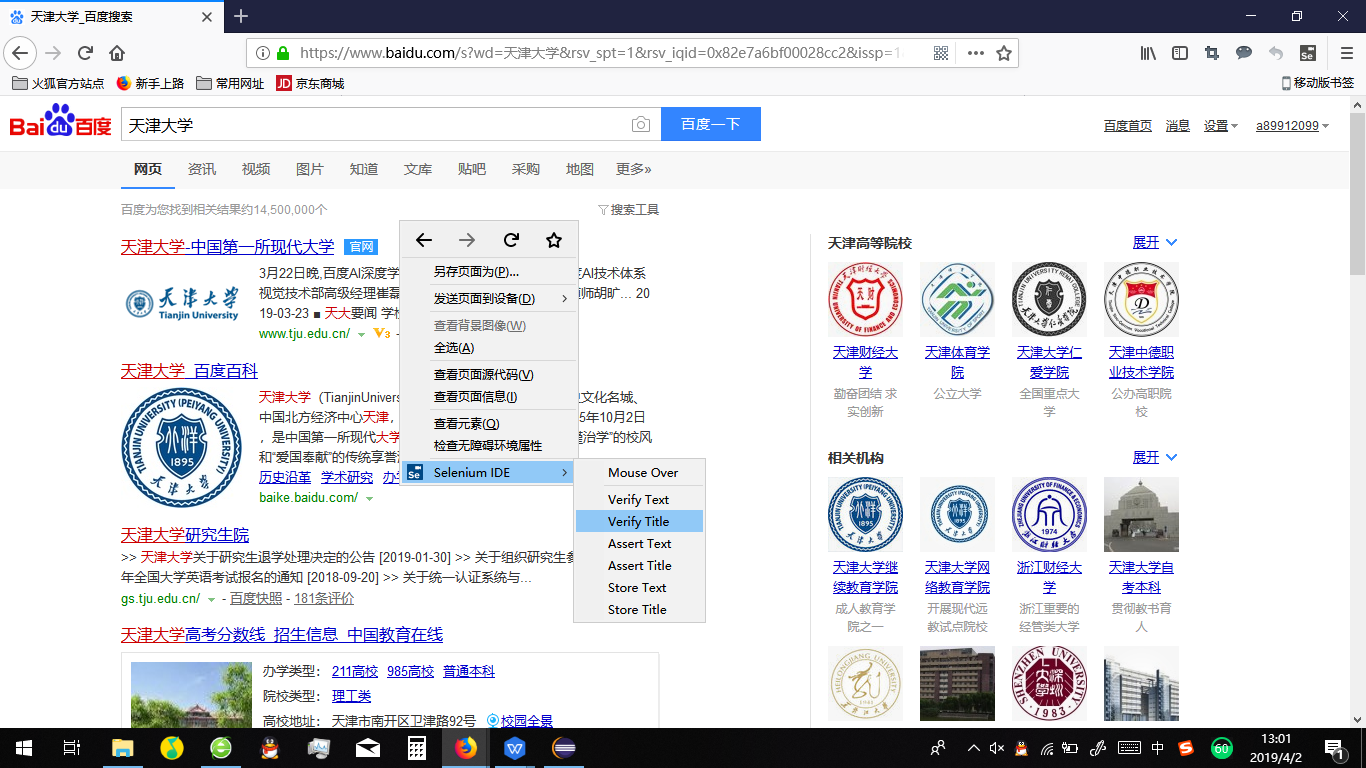
1.使用seleniumIDE进行录制并导出脚本。

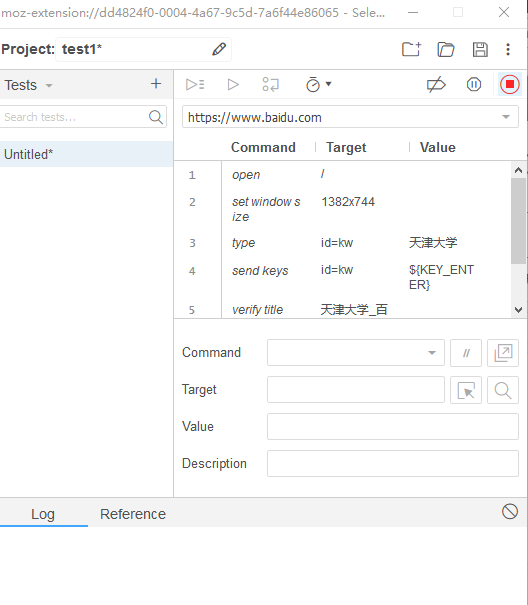
2.通过seleniumIDE，读取文件软件测试名单.xlsx中的内容，编写程序在网站<http://121.193.130.195:8800> 验证每个人的姓名，学号和URL

1. 概要设计（简单描述设计思路，配合UML图）
2. 打开firefox的selenium工具。
3. 录制一个名为test1的脚本



开始录制，并在录制完成后选择查看录制结果





按照内容编写程序，实现自动化测试

2.测试程序：

第一步，导入excel文件中的所有学生信息，存储在一个链表上

第二步，输入期望输出结果，得到输出。

1. 详细设计（详细描述具体如何实现，附代码及说明）

实验一：

**package** test;

**import** java.util.regex.Pattern;

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

**import** org.junit.\*;

**import** **static** org.junit.Assert.\*;

**import** **static** org.hamcrest.CoreMatchers.\*;

**import** org.openqa.selenium.\*;

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

**import** org.openqa.selenium.support.ui.Select;

**public** **class** TestBaidu {

**private** WebDriver driver;

**private** String baseUrl;

**private** **boolean** acceptNextAlert = **true**;

**private** StringBuffer verificationErrors = **new** StringBuffer();

@Before

**public** **void** setUp() **throws** Exception {

String driverPath = System.*getProperty*("user.dir") + "/src/test/geckodriver.exe";

System.*setProperty*("webdriver.gecko.driver", driverPath);

driver = **new** FirefoxDriver();

baseUrl = "https://www.baidu.com/";

driver.manage().timeouts().implicitlyWait(30, TimeUnit.***SECONDS***);

}

@Test

**public** **void** testBaidu() **throws** Exception {

driver.get(baseUrl + "/");

WebElement we = driver.findElement(By.*id*("kw"));

we.click();

// driver.findElement(By.id("kw")).click();

driver.findElement(By.*id*("kw")).clear();

driver.findElement(By.*id*("kw")).sendKeys("天津大学");

driver.findElement(By.*id*("su")).click();

*assertEquals*("天津大学\_百度搜索", driver.getTitle());

}

@After

**public** **void** tearDown() **throws** Exception {

// driver.quit();

// String verificationErrorString = verificationErrors.toString();

// if (!"".equals(verificationErrorString)) {

// fail(verificationErrorString);

// }

}

**private** **boolean** isElementPresent(By by) {

**try** {

driver.findElement(by);

**return** **true**;

} **catch** (NoSuchElementException e) {

**return** **false**;

}

}

**private** **boolean** isAlertPresent() {

**try** {

driver.switchTo().alert();

**return** **true**;

} **catch** (NoAlertPresentException e) {

**return** **false**;

}

}

**private** String closeAlertAndGetItsText() {

**try** {

Alert alert = driver.switchTo().alert();

String alertText = alert.getText();

**if** (acceptNextAlert) {

alert.accept();

} **else** {

alert.dismiss();

}

**return** alertText;

} **finally** {

acceptNextAlert = **true**;

}

}

}

实验二：

**package** test;

**import** java.util.regex.Pattern;

**import** java.io.File;

**import** java.io.FileInputStream;

**import** java.io.InputStream;

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

**import** org.junit.\*;

**import** **static** org.junit.Assert.\*;

**import** **static** org.hamcrest.CoreMatchers.\*;

**import** org.openqa.selenium.\*;

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

**import** org.openqa.selenium.support.ui.Select;

**import** java.io.FileInputStream;

**import** java.io.FileNotFoundException;

**import** java.io.FileOutputStream;

**import** java.io.IOException;

**import** java.util.ArrayList;

**import** java.util.List;

**import** java.util.Map;

**import** org.apache.poi.hssf.usermodel.HSSFWorkbook;

**import** org.apache.poi.ss.usermodel.Cell;

**import** org.apache.poi.ss.usermodel.CellType;

**import** org.apache.poi.ss.usermodel.Row;

**import** org.apache.poi.ss.usermodel.Sheet;

**import** org.apache.poi.ss.usermodel.Workbook;

**import** org.apache.poi.xssf.usermodel.XSSFWorkbook;

**public** **class** TestBaidu {

**private** WebDriver driver;

**private** String baseUrl;

**private** **boolean** acceptNextAlert = **true**;

**private** StringBuffer verificationErrors = **new** StringBuffer();

@Before

**public** **void** setUp() **throws** Exception {

String driverPath = System.*getProperty*("user.dir") + "/src/test/geckodriver.exe";

System.*setProperty*("webdriver.gecko.driver", driverPath);

driver = **new** FirefoxDriver();

baseUrl = "http://121.193.130.195:8800";

driver.manage().timeouts().implicitlyWait(600, TimeUnit.***SECONDS***);

}

@Test

**public** **void** testBaidu() **throws** Exception {

driver.get(baseUrl+"/");

driver.findElement(By.*name*("id")).click();

driver.findElement(By.*name*("id")).sendKeys("3016218172");

driver.findElement(By.*name*("password")).sendKeys("218172");

driver.findElement(By.*id*("btn\_login")).click();

*assertEquals*("3016218172", driver.findElement(By.*id*("student-id")).getText());

*assertEquals*("李沛霖", driver.findElement(By.*id*("student-name")).getText());

*assertEquals*("https://github.com/SSFDLware/software-testing", driver.findElement(By.*id*("student-git")).getText());

driver.findElement(By.*id*("btn\_logout")).click();

driver.findElement(By.*id*("btn\_return")).click();

driver.findElement(By.*name*("id")).click();

driver.findElement(By.*name*("id")).sendKeys("3016218182");

driver.findElement(By.*name*("password")).sendKeys("218182");

driver.findElement(By.*id*("btn\_login")).click();

*assertEquals*("3016218182", driver.findElement(By.*id*("student-id")).getText());

*assertEquals*("王志航", driver.findElement(By.*id*("student-name")).getText());

*assertEquals*("https://github.com/YourBrotherTT/softwareTesting.git", driver.findElement(By.*id*("student-git")).getText());

}

@After

**public** **void** tearDown() **throws** Exception {

// driver.quit();

// String verificationErrorString = verificationErrors.toString();

// if (!"".equals(verificationErrorString)) {

// fail(verificationErrorString);

// }

}

**private** **boolean** isElementPresent(By by) {

**try** {

driver.findElement(by);

**return** **true**;

} **catch** (NoSuchElementException e) {

**return** **false**;

}

}

**private** **boolean** isAlertPresent() {

**try** {

driver.switchTo().alert();

**return** **true**;

} **catch** (NoAlertPresentException e) {

**return** **false**;

}

}

**private** String closeAlertAndGetItsText() {

**try** {

Alert alert = driver.switchTo().alert();

String alertText = alert.getText();

**if** (acceptNextAlert) {

alert.accept();

} **else** {

alert.dismiss();

}

**return** alertText;

} **finally** {

acceptNextAlert = **true**;

}

}

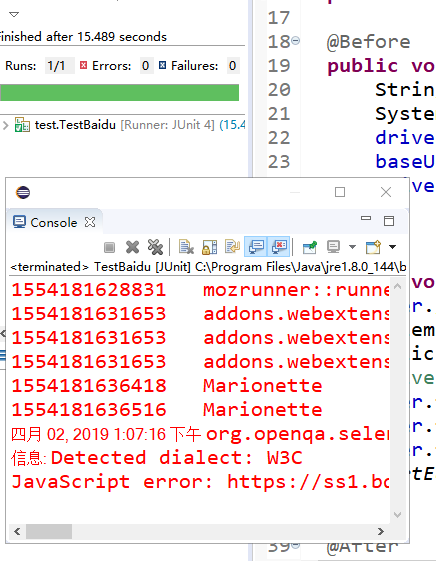
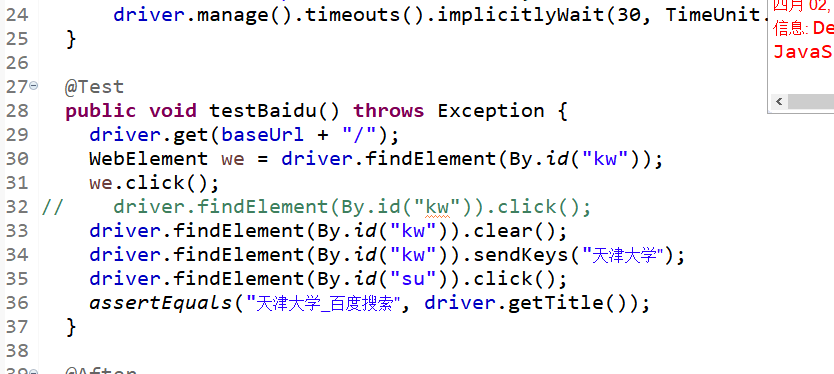
1. }调试分析（在实验过程中遇到的问题以及如何解决）

1在一开始编写测试程序时，我在Test中直接产生目标类，但是实际情况中这种做法会造成测试资源的浪费，所以将其单独列入before

2 源程序中我一开始是直接录入所有金钱种类，在程序直接调取数值，但是这种方法的适用性明显仅限于这个问题，因此我在后来将金钱录入方式改为数组，并在之后调用数组中的元素。

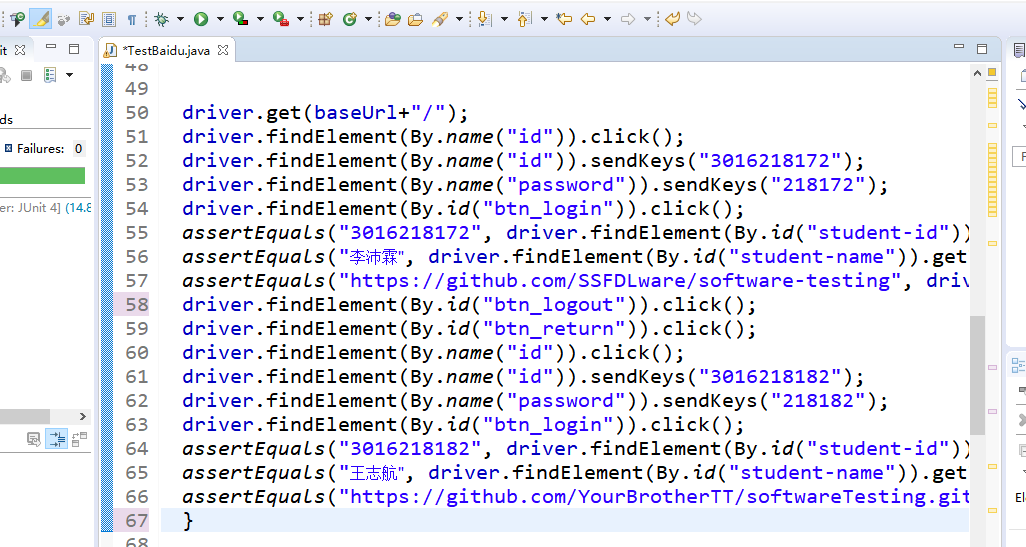
1. 测试结果（描述输入和输出）

任务一：测试样例与期望（输入）：

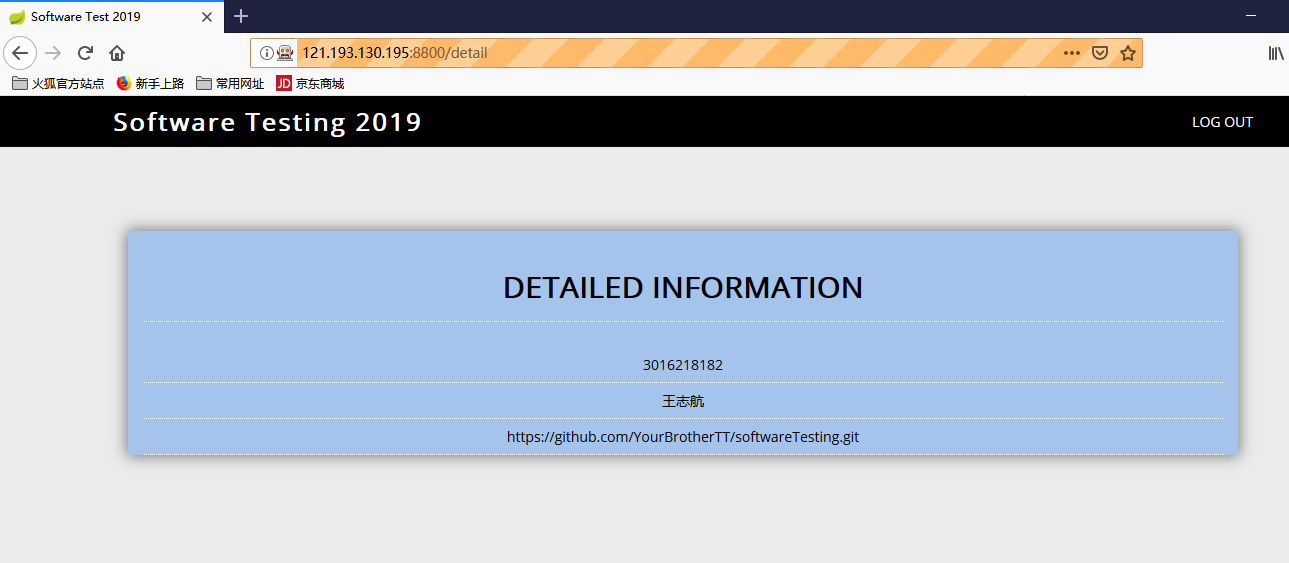


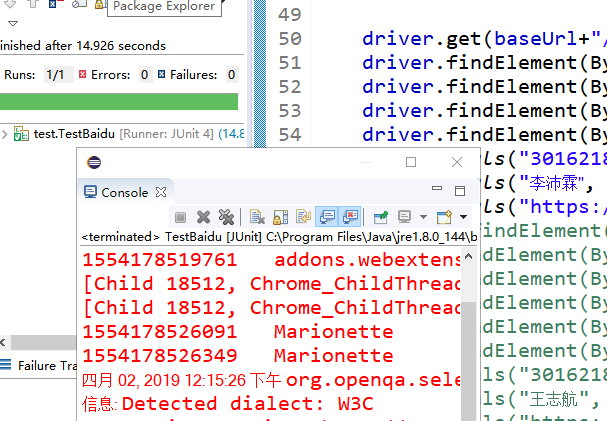


任务二：测试样例与期望（输入）：



结果：





1. 总结

完成任务一，任务二能够实现对每个人的个人信息的验证，但是由于jar包出现问题所以在xlsx文件导入时发生错误。

仓库url：https://github.com/SSFDLware/software-testing.git