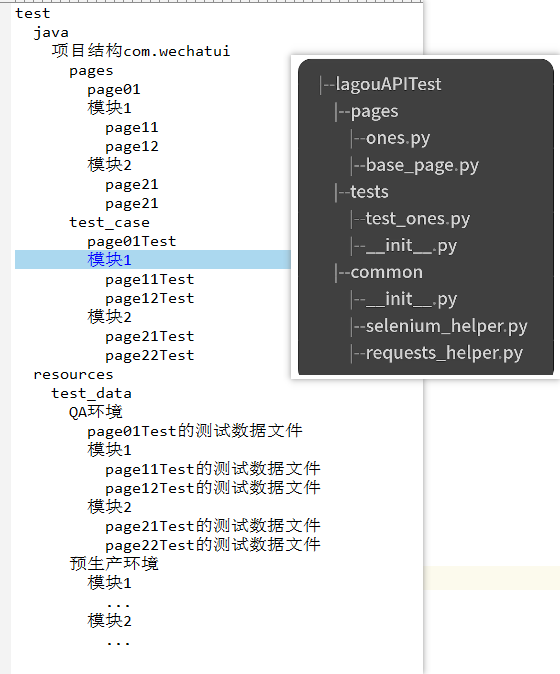
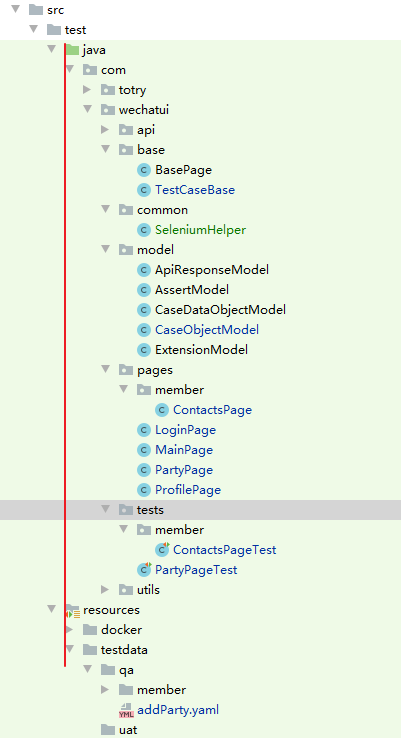
# 项目结构优化、测试环境切换

## 项目结构优化

依据下面的结构优化原先的项目结构



优化后如下：



测试数据管理：测试数据存放的文件是否跟测试用例强绑定，测试数据是否容易替换，测试数据和测试框架的耦合度如何。

包括两种情况，如下：

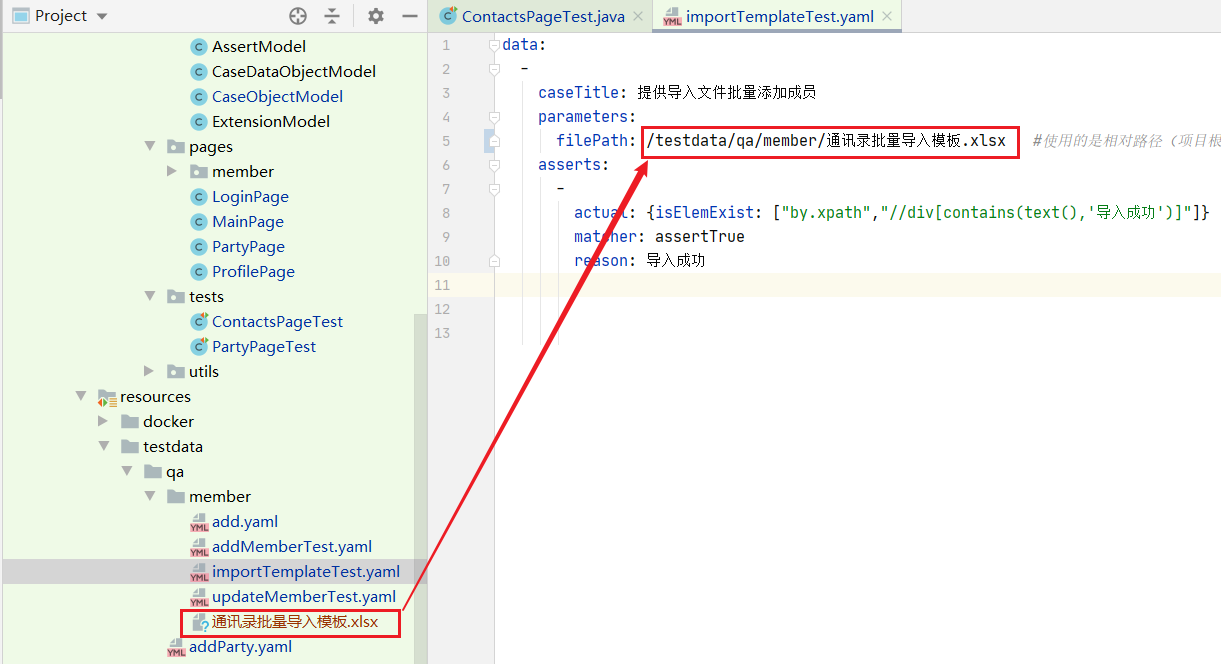
1）



当前测试数据文件的命名已按测试方法名称来重新命名。

把上面硬编码指定的文件名/testdata/qa/member/addMemberTest.yaml改成根据根据测试方法名自动生成，生成规则如下：/ 加 test\_data 加 命令行参数传入的测试环境字符串 加 测试类所在模块名 加 测试方法名 加 .yaml

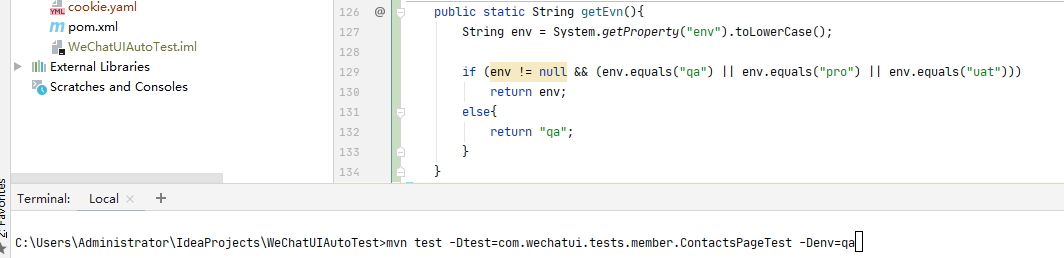
2）



按上面的例子说明规则和实现逻辑，importTemplateTest.yaml中filePath只需指定文件名称即可，无需指定相对路径，但该文件的存放路径需和测试文件的路径一致，即把importTemplateTest.yaml和通讯录批量导入模板.xlsx 放在同一个路径下。

实现参考代码逻辑

## 测试环境秒切换



mvn test执行时通过 –Denv=qa指定运行环境，通过System.getProperty("env")读取指定的环境。



因为获取测试文件路径都是通过getTestDataFilePath()方法来获取的，所以实现测试环境切换只需在该方法中统一处理即可。如下：



# 根据PO模式优化框架代码、日志输出优化

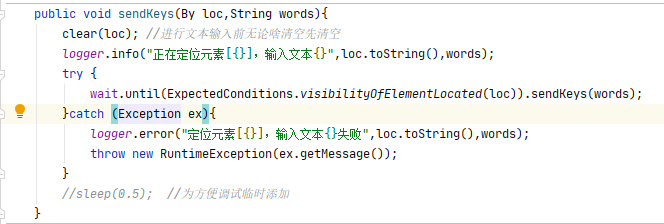
## 根据PO模式优化框架代码

在common包下新增seleniumHelper.java类，seleniumHelper里包括了所有针对selenium的操作，以后针对浏览器的各种的操作，全部放在这个文件。

优化前BasePage和TestCaseBase中都存在针对selenium的操作，现在需要抽离出来。

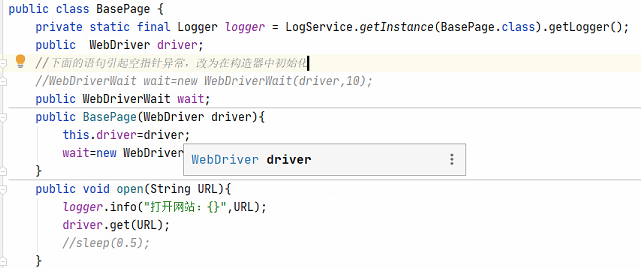
那seleniumHelper中操作浏览器的各种方法用静态方法还是实例方法？

因为优化前BasePage和TestCaseBase是被pages下的page类和tests下的测试类继承的，也就是每次创建page类或测试类对象时都会创建父类对象，对page类对象或测试类对象而已是有其对应的唯一父类对象的，而这些父类对象是存在针对selenium的操作的。现在把对这些针对selenium的操作抽离到seleniumHelper对象中，不使用继承，那就采用组合的方式实现，即page类和测试类有一个seleniumHelper类型的属性，且seleniumHelper中的操作方法都是实例方法。如果seleniumHelper中操作浏览器的方法是静态方法，在多线程运行时可能存在问题，如下面的代码



因为是UI操作，在多线程的情况下第一个线程在执行clear()操作时，另一个线程又调用sendKeys()方法传入了新的words参数，那第一个线程后面执行sendKeys(word)时输入的就是后一个线程的关键字。

所有浏览器操作封装在SeleniumHelper类中，之前所有需要用到Driver对象的类由组合Driver类型属性改为组合SeleniumHelper类型的属性。



如之前的BasePage有一个WebDriver driver属性，现改为SeleniumHelper seleniumHelper，而具体的page类通过继承BasePage类获得SeleniumHelper属性，page类中与浏览器的交互动作均通过调用SeleniumHelper对象完成。

调用关系如下（-->表示继承）：

SeleniumHelper{ WebDriver属性 ; WebDriverWait属性 }

ContactsPageTest --> TestCaseBase{有一个SeleniumHelper类型属性;有一个@BeforeAll方法}

调用

ContactsPage --> BasePage{有一个SeleniumHelper类型属性}

说明如下：当前并发模式采用的是测试类并发，即多个测试类并发运行，但一个测试类内部的测试方法是顺序运行。因为存在测试类并发的情况，所以SeleniumHelper中操作浏览器的方法不能使用静态方法，原因上面有解析，且因为需要并发，所以SeleniumHelper中的WebDriver属性也不能是类变量。程序调用过程如下：运行ContactsPageTest中的测试用例时TestCaseBase中的@BeforeAll方法先运行，在@BeforeAll方法中创建SeleniumHelper对象并赋给TestCaseBase的SeleniumHelper属性，接着运行ContactsPageTest中的测试用例，测试用例中调用ContactsPage中的方法时需先实例化ContactsPage对象，实例化ContactsPage对象时传入前面创建的SeleniumHelper对象。

可见，ContactsPageTest测试类执行过程中是通过同一个SeleniumHelper对象串联起来的，而SeleniumHelper对象中有一个WebDriver属性，所以操作的是同一个浏览器。

通过上面的调用关系可见，TestCaseBase和BasePage都组合了SeleniumHelper类，即耦合，根据面向接口编程的原则，将定义一个接口，SeleniumHelper类实现该接口，且TestCaseBase和BasePage类耦合该接口而不是SeleniumHelper类。因为SeleniumHelper是实现接口的，所以可以更方便的通过AOP进行统一日志处理。

## 日志输出优化

对处处理能力强的另一个特点就是系统日志清晰，方便调试。除了错误处理器之外，系统的操作日志也能帮你快速排查问题根源。所以平时的日志一定要清晰详细，最好具备上下文，这样我们才能根据日志进行有效调试。快速定位错误发生的原因，对于测试框架来说，系统日志除了要按等级，比如debug、info 、warn、error 等划分外，最好还应该包括如下的内容。第一个内容是记录进入测试用例的开始和结束时间。 第二个是最好记录测试人员的 关键操作，比如说写文件操作，连接D B操作，更改D B 操作等。第三个是关键方法的异常 信息一定要有，例如啊测试运行模块出错后，它的出错时的部分上下文信息最好能够保持。

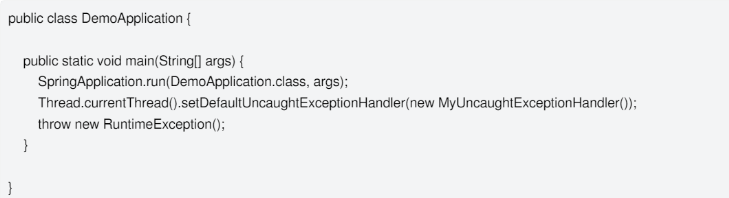
第一，测试报告，测试报告应该全面包括测试用例条数统计，测试用例成功失败的百分比，测试用例总执行时间等总体的信息，还应该包括对于单条用例来说是测试用例的ID、测试用例的运行结果、测试用例的运行时间，测试用例的所属模块测试失败时刻系统的截图，测试的日志信息等都应该包括在我们的测试报告里。

1. 如何统一处理未捕获的运行时异常以记录到日志？

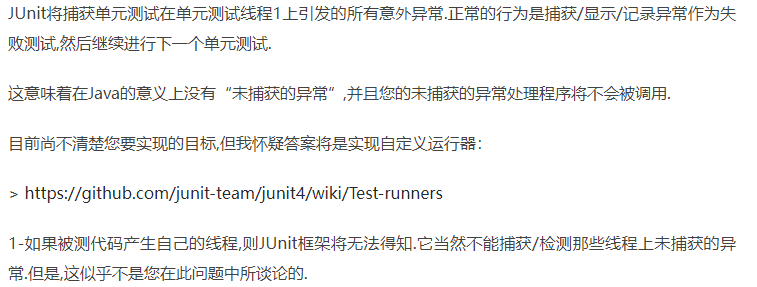
本想采用如下方法：

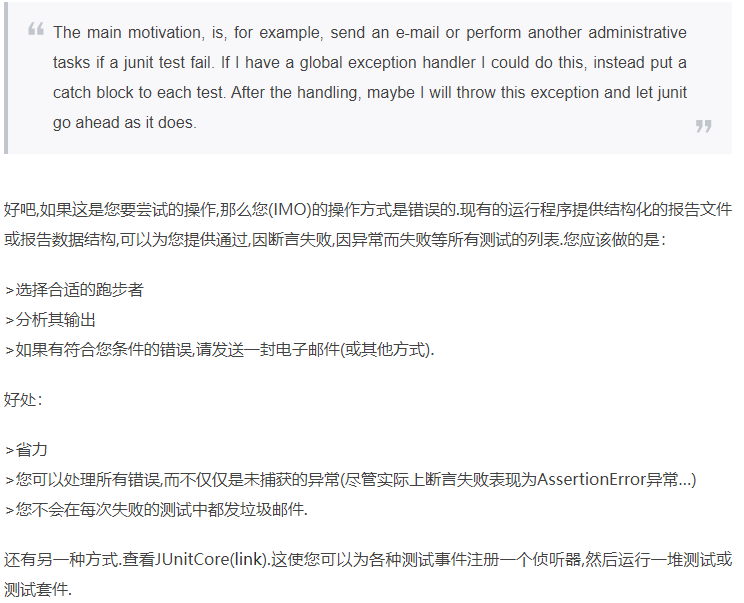






但还是无法捕获，具体原因可能是：<https://www.icode9.com/content-1-525698.html>







最终解决办法：



Junit测试运行时如果抛出未捕获运行异常，则该测试用例会fail or abort，那直接在TestWatcher中进行异常处理即可。

1. 如何往allure中添加用例执行的开始时间和结束时间？

实现思想如下：

