# 概要说明

## Excel使用原因与作用

由于本项目框架为基于关键字驱动的selenium自动化测试框架，又名为基于表格测试，或者名为基于行为词测试（Action word based），是通过解析相应的excel以实现测试的逻辑与步骤，以及测试数据的获取与读入。因此，excel在此框架中扮演着非常重要的角色，起着至关重要的作用。

## Excel的总体介绍和说明

本项目总共采用了2个excel表，一个是存储测试步骤，逻辑与数据的测试案例excel，一个是用于存储页面元素位置的excel。Excel的版本无论是2003，还是2007年版本均可以适用，本项目推荐使用07版本的excel。图1展示了存储页面元素位置的excel，比如其中LoginPage.xlsx为一个存储登录页面里所有元素位置信息的excel，以下把这种存储页面元素位置信息的excel简称为Page。

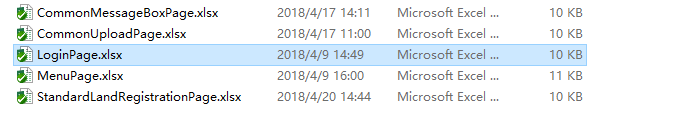


图1 页面元素存储excel示例

图2展示了存储测试步骤，逻辑与数据的测试案例excel。比如login\_test.xlsx就是为一个登录模块的测试案例excel，以下叙述均把这种excel简称为Testcase。

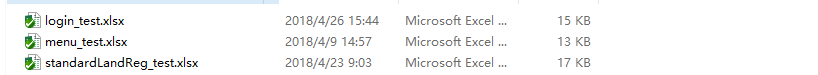
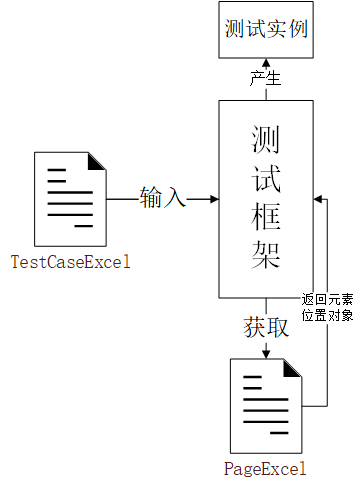


图2 测试案例excel示例

图3展示了两个excel，以及框架之间的相互关系。

 图3 两个excel和框架之间的关系

# 详细说明

## Page

### Page说明

Page，也就是存储页面元素位置信息的excel，用来表示页面的元素定位，框架将会解析这个excel，自动生成元素定位信息，通过一个定位元素对象封装并提供给测试案例。这个excel里面包含了七列，从左到右依次为别名，一级节点定位方式，一级定位值，二级节点定位方式，二级定位值，三级节点定位方式，三级定位值，它的每一行为一个元素位置对象。接下来详细介绍这些列的含义，作用与规约。

1. 别名： 自定义的元素位置名称，用来体现和标明元素，别名只要与测试案例里相应的元素位置列里字符“.”后面的一致即可，无其他任何约束。图示例见。
2. 一级节点定位方式：定位元素时采用的一级定位方式，目前支持的定位方式的书写形式有id, name, className, tagName, linkText, xpath, css。这里大小写随意，但推荐使用驼峰式写法。同时这里的css书写形式即为定位中的cssSelector方式，其他的书写形式与定位方式一致。
3. 一级定位置：一级定位元素时具体的定位值，与相应的通过代码书写的定位值一致即可，定位置的书写在excel中没有任何约束。

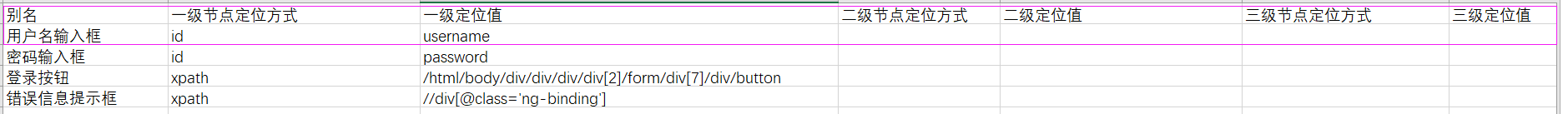
这里的一级元素意思是如果元素采用间接定位时，所最先定位的元素对象，即通过一级定位方式+一级定位值确定一级定位元素对象，如果元素是采用直接定位方式，那么一级元素对象即为自己本身。

1. 二级节点定位方式：定位元素时采用的二级定位方式。其他的见一级定位。
2. 二级定位值：二级定位元素具体的定位值。其他的见一级定位。
3. 三级节点定位方式：定位元素时采用的三级定位方式。其他的见一级定位。
4. 三级定位值：三级定位元素具体的定位值。其他的见一级定位。

二级和三级定位方式均在间接定位元素时使用，如果直接定位元素，这四列均可以为空。

### 简单案例说明：

比如对于一个用户名的输入框，在测试案例excel元素位置列为，所以其别名为“用户名输入框”。Page里面为



由于其采用直接定位方式，因此其二级和三级定位信息均为空。这个转义成代码即为：By.id(“username”)，也由此更为清晰的可见，定位方式即为By后面的id，定位值就是双引号之间的**所有内容（字符）。**这里关于元素定位的一些学习资料和文档说明详见<https://www.cnblogs.com/puresoul/p/4269236.html> <http://www.cnblogs.com/qingchunjun/p/4208159.html> 等

### 维护指南：

1. 该excel列的具体名称可以任意改变，无需变动代码。
2. 该excel所有列所在的位置，以及列名所处的行不能改变，比如列名为“别名”的列，它必须处在excel中的第一列和第一行，不能将其放到其他列和其他行里，如果确实要改变其所在列，则需要修改位于项目框架下src—main—java—ExcelStaticValue—ElementLocationExcelPage类下的代码，通过改变列的索引值即可。一般来说，列名习惯默认放在第一行，这里不建议修改。
3. 如果要添加定位方式和修改定位方式的书写形式的话，则需要修改位于项目框架下src—main—java—Utils—PageParse类里的parsePageElementsByInfos方法，增加else if和修改双引号里的值即可。
4. 这里不建议再增加定位层级，一般来说三级的定位方式已经完全可以定位所有“有必要且存在”的页面元素，再增加层级定位只会降低元素定位信息的可读性和维护性。如果元素难以定位，推荐使用cssSelector（excel中的书写形式为css）和xpath的定位方式，尤其是前者，通过一定的css语法基本上可以直接一次定位。详细的定位语法和写法可以网上搜索相应的教程和文档。
5. 元素位置信息的excel文件的命名方式并不存在特别的标准，只需要与测试案例excel中的元素位置信息里字符”.”前面的信息一致即可，例如上节的案例里，对于测试案例里的元素位置，字符 ”.”前面的信息前面为LoginPage，同时相应的存储这个元素所处的excel文件名也为LoginPage即可。但这里仍建议使用标准命名规则。

**规则为： “测试页面（模块）名”+Page(用来表示这是个存储页面元素的excel)+xlsx（或xls）。**

例如：存储登录页面元素位置信息的excel文件命名为---LoginPage.xlsx.

## TestCase

### TestCase说明

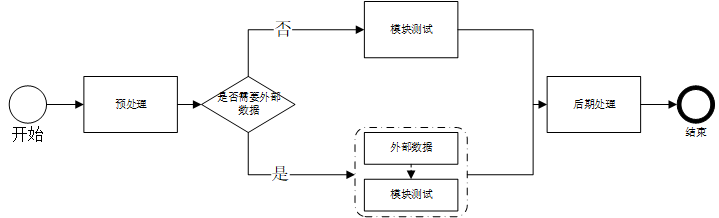
TestCase，是框架测试的核心excel，里面存储了测试的步骤，逻辑与测试所需的数据。框架将通过解析这个excel完成测试逻辑和测试任务。这个excel中目前含有预处理，模块测试，后期处理，以及外部数据四个sheet表。四个sheet表之间的关系与逻辑如图4所示。

图4 测试案例excel里各个sheet的关系与逻辑

如图2所示先是预处理，然后根据一定条件判断是否需要采用外部数据sheet，满足则采用外部数据，不满足则不采用，然后进行模块测试，最后进行后期处理，并最终结束测试案例。四个sheet的含义分别为

1. 预处理：含有模块测试前的准备工作的测试逻辑步骤，是一次性完成的工作，不需要进行反复的运行。
2. 模块测试：含有模块测试逻辑步骤的sheet，有可能需要反复运行测试，重复执行步骤逻辑。
3. 外部数据：大量的测试数据存在的sheet，有可能存在，有可能不存在。
4. 后期处理：含有模块测试结束时的后期清理的测试逻辑，包含数据的清理还原，以及浏览器的关闭等，也是一次性完成的工作，不需要反复的执行

这4个sheet除了外部数据sheet比较特殊以外，其他三个sheet仅具有逻辑意义上的差距，并无内容上的差距，即里面的内容和列是一样的。在这三个sheet里面都是每一行用来表示测试的每一个逻辑步骤，列分别有步骤id，描述，动作（关键字），元素位置，数据（值）（时间以ms为单位），接下来分别这几列进行说明。

1. 步骤id：每一步的一个标识符。写法无强制要求，但推荐为预处理sheet里为“TestBase\_” + 步骤序号，模块测试里为“模块测试名\_”+“test\_”+序号，后期处理sheet里为“AfterTest\_”+步骤序号。
2. 描述：每一步的直观描述，用来体现步骤的含义。
3. 动作（关键字）：每一步所需要采用的动作（关键字），比如点击，输入，前进页面，后退页面，选择下拉等。目前框架支持的动作（关键字）已全部写在了本项目框架下config文件下的actionLibraries.properties文件里。该文件里的左侧即为excel中动作的强制性标准写法。如图5所示的文件的部分截图的左侧。

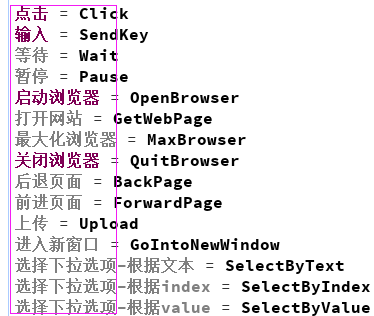


图5 actionLibraries.properties文件部分示例

1. 元素位置：用来表示测试步骤可能会存在元素对象信息。写法强制为：元素位置信息所在的excel文件名+“.”+元素别名。例如对于登录页面上的用户名输入框来说，元素位置信息所在的excel文件名为LoginPage，别名为用户名输入框，因此其写法为LoginPage.用户名输入框。
2. 数据（值）（时间以ms为单位）：可能需要的测试数据和值，当测试数据过多，需要外部数据sheet时，则此地方应强制写为“外部输入”即可。当数据不存在时，可以填写“无”，也可以直接空在那里。

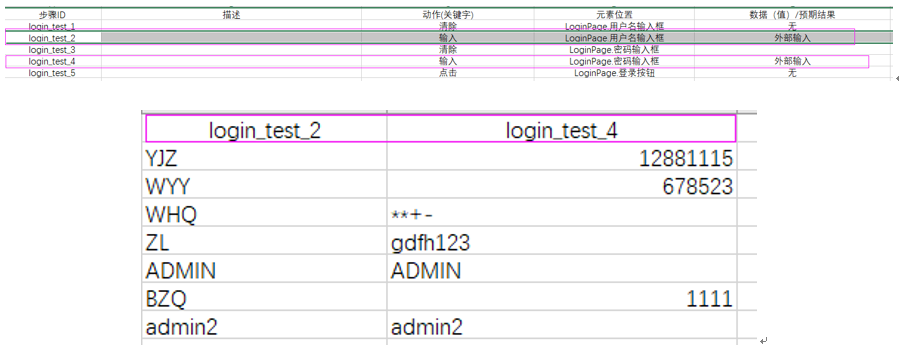
外部数据sheet即为外部数据所组成的sheet表。用来存放外部数据，这里需要注意的是目前框架里规定外部数据sheet里的外部数据列名必须为模块测试里相应的步骤id。如图6所示，当用户名和密码需要外部数据时，则需设定相应的步骤id为相应数据列名。

图6 外部数据列名示例

### 简单案例说明

简单案例以一个登录页面的测试案例excel进行说明，图7展示了一个登录测试案例excel

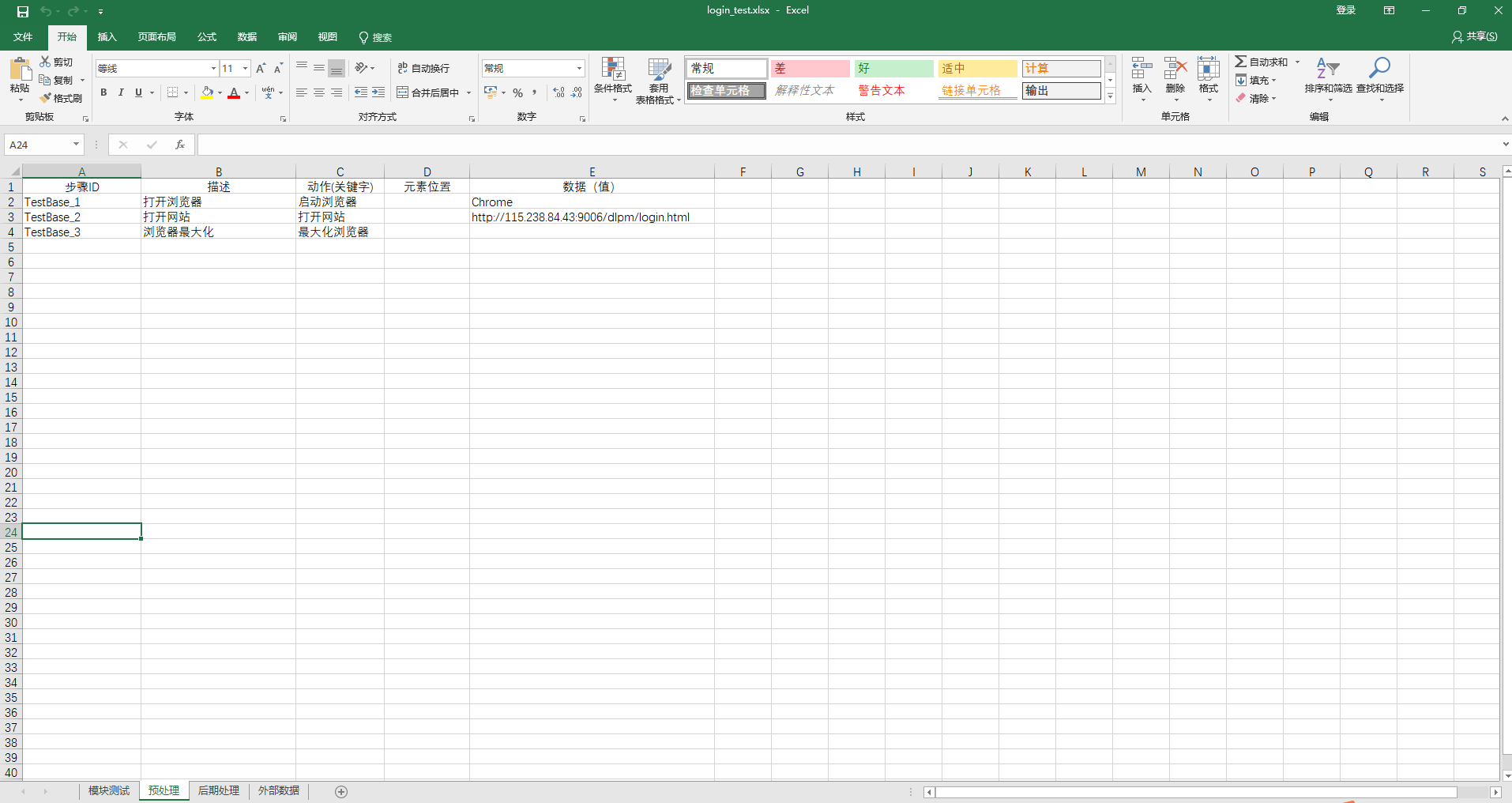


图7 登录测试案例excel示例

其中有模块测试，预处理，后期处理，外部数据sheet。图8展示了模块测试sheet里两个测试步骤。

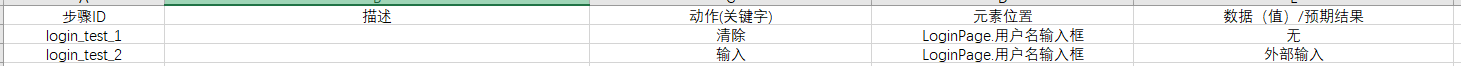


图8 测试步骤例子

### 维护指南

1. Excel中sheet名如果想改变，需要更改框架下的src—main—java—ExcelStaticValue—SheetName类里的属性值。各个sheet的顺序没有规定，可以任意修改。
2. Excel中各个列名可以任意更改，但列的所在位置如果需要修改，则需要更改框架下src—main—java—ExcelStaticValue—TestcaseExcelPage类里的相应索引值。列名所在的行习惯默认为第一行，不建议修改。
3. 如果增加动作（关键字），则需要第一步在actionLibraries.properties先增加相应的动作（关键字），以及相应的类名，写法参照已有动作（关键字）的写法。第二步在框架下src—main—java—KeyWordLibrary包下增加相应的类，这个类必须继承SuperAction类。
4. 如果在模块测试，预处理或者后期处理sheet里继续增加相关列，则需要在框架下src—main—java—Bean—TestStep类里增加相应的属性和方法，同时根据需要以及框架的特点在其他有关类和方法里增加相应的代码。
5. Excel文件的命名方式推荐命名方式为：“测试的模块名称”+“\_”+“test”。例如登录测试案例excel文件名为“login\_test.xlsx”。