rebotframework的架构是一个通用的验收测试和验收测试驱动开发的自动化测试框架,它具有易于使用的表格来组织测试过程和测试数据

他使用关键字驱动的测试方法

对于上面的例子来说,open broser input text click button 和close browser,都是关键字,这些关键字由rebotframework-selenium2library

安装环境

1. 首先安装python环境(setup tools 和pip)
2. 安装robotframework(可以通过pip命令安装)
3. 安装wxpython(安装wxpython和ride要匹配
4. 安装ride

Rebotframework框架的结构

1. 建立项目
2. 建立测试套件
3. 建立测试用例

学习关键字

1. log(具有输出和打印的功能)
2. Set variable(定义变量)
3. Catenate 连接
4. SEPARATOR= 可以将多个连接的信息进行分割,等号后面接的是连接符
5. Create list 定义列表(即定义多个变量):注意:打印出来的时候,每个字符串前面加u,是为了统一编码的问题,将字符串转化成unicode编码
6. In range:表示范围in range范围定义为10,他表示的范围是0~9
7. :for:表示循环
8. Exit For Loop If:执行for循环的判断,Exit For Loop If条件成立,通过log打印当前的字符
9. Evaluate:通过他可以使用python语言中所提供的方法
10. Import library:导入python中的文件
11. Evaluate:调用python中的方法
12. Comment:注释(注释必须和关键字不在同一行的单元格里)
13. #:注释:注释和’#’建可以在同一个单元格里

二.其他的库

使用其他的库时,需要先导入

1.screen库

关键字

1. take screenshot:实现当前屏幕的截图
2. collection库

关键字

1. 创建字典

Create Dictionary

1. 操作字典的关键字

Get dictionary items:用于读取字典的key和value

Get dictionary keys:用于读取字典的key

Get dicionary value:获取字典的value

Get from dictionary:获取字典中对应的value

三:认识ride

1. edit标签页中主要分为:加载外部文件,定义内部变量,定义元数据等三个部分

(1):加载外部文件

Add library:加载测试库,主要是python目录中的测试库

1. 如果添加的库库名不存在或者库名错误,将会红色显示,黑库正常表示正常’
2. 如果你是在”测试套件”中添加的库,那么这个库中所提供的关键字可以被当前测试套件下的用例使用
3. 如果你是在”测试项目”中添加的库,当前项目下的测试用例不能使用库中的关键字,需要在用例中相应的”测试套件”中再次添加库

Add Resource:加载资源,主要是你工程相关的资源文件

1. 添加资源的作用域与库一样,我们这里添加到的测试刀剑中,那么它的作用域就是当前测试套件下的所有用例;添加的资源可以在external Resource注重看到
2. Add variables:加载变量文件
3. :定义内部变量

Add Scalar:定义变量(即定义公共变量)

1)Name:用于定义变量名:${hi}

2)Value:用于给变量赋值

Add List:定义列表型数据

1)

1. :定义元数据

Add Metedata:定义元数据,

4.2text edit标签

4.3run标签

(1)run标签提供了丰富的操作和日志

1)execution profile:选择运行方式,默认使用pybot来运行案例,pybot的运行是通过python编译器来完成

2)start和stop:用例的运行和停止

3)report和log:报告和日志:要运行之后才能点击,他们之间的区别:报告更多的是结果上的显示,日志更多的是过程的记录,在测试用例调试的过程中更多使用日志来查看执行的错误,当只想知道测试用例的最终执行情况时用报告

4)autosave:自动保存

5)argument:pybot的参数(或者是jybot等),可以在这里输入pybot的命令完成相应的操作

6)only run test with these tags:只运行这些标记的测试案例

7)skip test with these tags:跳过这些标记的测试案例

4.3.2

运行和停止

4.33:筛选执行测试用例

1. select all test:选择当前测试套件的所有测试用例
2. Select only failed test:选择当前套件下运行失败的用例
3. Select only passed test:选择当前测试套件运行成功的用例

‘

筛选标记:

Settings:

Setup用于设置初始化工作,在每一个测试用例先被执行

Teardown:方法在每一个测试方法执行后调用,这个方法用于完成测试用例执行后的清理工作,如执行:”close browser”

Time out:用于设置用例的超时时间

Template:指定模版的关键字

Tags:用于给用例添加标记

注意:测试套件中的setup和teardown用于当前测试套件的开始和结束要做的的事情;他们会作用于测试套件下每一个测试用例开始和结束要作的事情

4.5用户关键字

1. 系统关键字和用户关键字

创建关键字的流程

1. 在弹出的菜单栏里输入关键字的名称

注意:这个关键字和测试用例平级,所以在测试用例中可以直接使用

1. 设置入参;如果需要多个参数的话,那么可以用”|”隔开;
2. 添加用例
3. 接着可以在用例中直接使用用户关键字

4.52:创建资源

1. 目的:创建资源存放用户关键字

创建资源的流程

1. 输入资源名称
2. 创建关键字
3. 导入资源

5.2:安装selenium2library

关键字:

open browser:打开浏览器;后面的一个单元格填写url地址

Input text:输入内容

Click button:点击按钮

Close browser:关闭浏览器

5.3元素定位

对于web自动化测试来说;就是操作元素之间需要先找到元素,换句话说就是

定义元素

Selenium2library提供了非常丰富的定位器

例如:

Id

Name

Xpath

Link

Link text

Tag

Input text:用于输入框的关键字;

Click button:是按钮点击的关键字;

Xpath定义

1. xpath的绝对路径
2. Xpath的相对路径
3. Xpath同样可以利用元素自身的属性

关闭浏览器:

Close Browser:关闭浏览器

Close all Browser:关闭当前的浏览器;关闭所有打开的浏览器和缓存重置

备注:要想通过不同的浏览器打开url地址,一定要安装浏览器对应的驱动

Chrome的驱动为:chromedriver.exe

IE的驱动为:IEDriverServer.exe

浏览器默认为空时启动:Firefox

Maximize Browser window:浏览器最大化;关键字使当前打开的浏览器全屏

Get window size:用于获取当前浏览器的宽度和高度,获得在浏览器窗口的宽,高