

# RobotFramework-Selenium2Library

## 中文版

Library Version : V1.8

翻译：代兵，张哲

校对：张哲

日期：2017-02-28

Selenium2Library.....	10
介绍.....	10
测试前提条件.....	10
定位或指定元素.....	11
使用 <b>Locators</b> .....	11
使用 <b>webelements</b> .....	13
自定义定位器.....	14
超时等待.....	14
<b>Import</b> 引入策略.....	15
关键字.....	16
关键字 : <b>Add Cookies</b> .....	16
关键字 : <b>Add Location Strategy</b> .....	16
关键字 : <b>Alert Should Be Present</b> .....	17
关键字 : <b>Assign Id To Element</b> .....	17
关键字 : <b>Capture Page Screenshot</b> .....	18
关键字 : <b>Checkbox Should Be Selected</b> .....	18
关键字 : <b>Checkbox Should Not Be Selected</b> .....	19
关键字 : <b>Choose Cancel On Next Confirmation</b> .....	19
关键字 : <b>Choose File</b> .....	19
关键字 : <b>Choose Ok On Next Confirmation</b> .....	20
关键字 : <b>Clear Element Text</b> .....	20
关键字 : <b>Click Button</b> .....	20
关键字 : <b>Click Element</b> .....	21

关键字 : <b>Click Element At Coordinates</b> .....	21
关键字 : <b>Click Image</b> .....	22
关键字 : <b>Click Link</b> .....	22
关键字 : <b>Close All Browsers</b> .....	22
关键字 : <b>Close Browsers</b> .....	23
关键字 : <b>Close Window</b> .....	23
关键字 : <b>Confirm Action</b> .....	23
关键字 : <b>Current Frame Contains</b> .....	24
关键字 : <b>Current Frame Should Not Contain</b> .....	24
关键字 : <b>Create Webdriver</b> .....	25
关键字 : <b>Delete All Cookies</b> .....	26
关键字 : <b>Delete Cookie</b> .....	26
关键字 : <b>Dismiss Alert</b> .....	26
关键字 : <b>Double Click Element</b> .....	27
关键字 : <b>Drag And Drop</b> .....	27
关键字 : <b>Drag And Drop By Offset</b> .....	27
关键字 : <b>Element Should Be Disabled</b> .....	28
关键字 : <b>Element Should Be Enabled</b> .....	28
关键字 : <b>Element Should (Not) Be Enable</b> .....	29
关键字 : <b>Element Should Be Visible</b> .....	29
关键字 : <b>Element Should Contain</b> .....	29
关键字 : <b>Element Should Not Be Visible</b> .....	30

关键字 : <b>Element Should Not Contain</b> .....	30
关键字 : <b>Element Text Should Be</b> .....	31
关键字 : <b>Execute Async Javascript</b> .....	31
关键字 : <b>Execute Javascript</b> .....	31
关键字 : <b>Focus</b> .....	32
关键字 : <b>Frame Should Contain</b> .....	32
关键字 : <b>Get Alert Message</b> .....	32
关键字 : <b>Get All Links</b> .....	33
关键字 : <b>Get Cookie Value</b> .....	33
关键字 : <b>Get Cookies</b> .....	33
关键字 : <b>Get Element Attribute</b> .....	33
关键字 : <b>Get Horizontal Position</b> .....	34
关键字 : <b>Get List Items</b> .....	34
关键字 : <b>Get location</b> .....	34
关键字 : <b>Get Matching Xpath Count</b> .....	34
关键字 : <b>Get Selected List Lable</b> .....	35
关键字 : <b>Get Selected List Value</b> .....	35
关键字 : <b>Get Selected List Values</b> .....	35
关键字 : <b>Get Selenium Implicit Wait</b> .....	35
关键字 : <b>Get Selenium Speed</b> .....	36
关键字 : <b>Get Selenium Timeout</b> .....	36
关键字 : <b>Get Source</b> .....	36

关键字 : <b>Get Table Cell</b> .....	36
关键字 : <b>Get Text</b> .....	37
关键字 : <b>Get Title</b> .....	37
关键字 : <b>Get Value</b> .....	37
关键字 : <b>Get Vertical Position</b> .....	38
关键字 : <b>Get Webelement</b> .....	38
关键字 : <b>Get Window Identifiers</b> .....	38
关键字 : <b>Get Window Names</b> .....	38
关键字 : <b>Get Window Position</b> .....	38
关键字 : <b>Get Window Size</b> .....	39
关键字 : <b>Get Window Title</b> .....	39
关键字 : <b>Go Back</b> .....	39
关键字 : <b>Go To</b> .....	40
关键字 : <b>Input Password</b> .....	40
关键字 : <b>Input Text</b> .....	40
关键字 : <b>Input Text Into Prompt</b> .....	40
关键字 : <b>List Windows</b> .....	40
关键字 : <b>List Selection Should Be</b> .....	41
关键字 : <b>List Should Hava No Selections</b> .....	41
关键字 : <b>Location Should Be</b> .....	41
关键字 : <b>Location Should Contain</b> .....	41
关键字 : <b>Log Location</b> .....	42

关键字 : <b>Locator Should Match X Times</b> .....	42
关键字 : <b>Log Location</b> .....	42
关键字 : <b>Log Source</b> .....	42
关键字 : <b>Log Title</b> .....	43
关键字 : <b>Maximize Browser Window</b> .....	43
关键字 : <b>Mouse Down</b> .....	43
关键字 : <b>Mouse Down On Image</b> .....	43
关键字 : <b>Mouse Down On Link</b> .....	44
关键字 : <b>Mouse Out</b> .....	44
关键字 : <b>Mouse Over</b> .....	44
关键字 : <b>Mouse Up</b> .....	45
关键字 : <b>Open Browser</b> .....	45
关键字 : <b>Open Context Menu</b> .....	46
关键字 : <b>Page Should Contain</b> .....	46
关键字 : <b>Page Should Contain Button</b> .....	47
关键字 : <b>Page Should Contain Checkbox</b> .....	47
关键字 : <b>Page Should Contain Element</b> .....	47
关键字 : <b>Page Should Contain Image</b> .....	48
关键字 : <b>Page Should Contain Link</b> .....	48
关键字 : <b>Page Should Contain List</b> .....	49
关键字 : <b>Page Should Contain Radio Button</b> .....	49
关键字 : <b>Page Should Contain TextField</b> .....	49

关键字 :	Page Should (Not) Contain.....	50
关键字 :	Page Should (Not) Contain Button.....	50
关键字 :	Page Should (Not) Contain Checkbox.....	51
关键字 :	Page Should (Not) Contain Element.....	51
关键字 :	Page Should (Not) Contain Image.....	51
关键字 :	Page Should (Not) Contain Link.....	52
关键字 :	Page Should (Not) Contain List.....	52
关键字 :	Page Should (Not) Contain Radio Button.....	52
关键字 :	Page Should (Not) Contain TextField.....	53
关键字 :	Press Key.....	53
关键字 :	Press Key Native.....	56
关键字 :	Press Key Native Radio Button Should (Not) Be Set to.....	56
关键字 :	Register Keyword To Run On Failure.....	57
关键字 :	Reload Page.....	57
关键字 :	Select All From List.....	57
关键字 :	Select Checkbox.....	58
关键字 :	Select All Frame.....	58
关键字 :	Select From List.....	58
关键字 :	Select From List List By Index.....	58
关键字 :	Select From List List By Label.....	59
关键字 :	Select From List List By Value.....	59
关键字 :	Select Radio Button.....	59

关键字 : <b>Select Window</b> .....	60
关键字 : <b>Set browser implicit wait</b> .....	60
关键字 : <b>Set Selenium implicit wait</b> .....	61
关键字 : <b>Set Selenium speed</b> .....	61
关键字 : <b>Set Selenium Timeout</b> .....	61
关键字 : <b>Set Window Size</b> .....	62
关键字 : <b>Simulate</b> .....	62
关键字 : <b>Switch Browser</b> .....	62
关键字 : <b>Table cell should contain</b> .....	63
关键字 : <b>Table Column should contain</b> .....	64
关键字 : <b>Table Footer should contain</b> .....	64
关键字 : <b>Table Header should contain</b> .....	64
关键字 : <b>Table Row should contain</b> .....	65
关键字 : <b>Table should Contain</b> .....	65
关键字 : <b>Textarea Should Contain</b> .....	66
关键字 : <b>Textarea value should be</b> .....	66
关键字 : <b>Textfield should contain</b> .....	66
关键字 : <b>Textfield Value Should be</b> .....	67
关键字 : <b>Title should be</b> .....	67
关键字 : <b>Unselect checkbox</b> .....	67
关键字 : <b>Unselect Frame</b> .....	67
关键字 : <b>Unselect From List</b> .....	68



关键字：	Unselect From List By Index.....	68
关键字：	Unselect From List By Label.....	68
关键字：	Unselect From List By Value.....	69
关键字：	Wait for condition.....	69
关键字：	Wait Until Element Is Visible.....	69
关键字：	Wait Until Page Contains.....	70
关键字：	Wait Until Page Contains Element.....	70
关键字：	Xpath Should Match X Times.....	70
附录：	.....	72
更新内容.....		72
Locators 定位方式.....		72
新增关键字.....		72
变更关键字.....		73
移出关键字.....		73

# Selenium2Library

类库版本： 1.8.0

类库范围： 全局

命名参数： 支持

## 介绍

Selenium2Library 是 RobotFramework 针对 web 测试所定义的一套类库文件。

本文档是关于使用 Selenium2Library [用户文档](#)。有关安装，支持，以及更多信息请访问[项目页面](#)。

它内部是通过使用 Selenium2（webdriver）来控制 Web 浏览器。关于

Selenium2 和 webdriver 的更多信息，详见 [Selenium2 官方页面](#)。

Selenium2Library 是在一个真实的浏览器实例执行测试的。它可以支持在当下主流浏览器上进行测试工作，测试脚本可以供 Python 和 Jython 的解释器执行。

## 测试前提条件

在使用 Selenium2Library 测试用例执行前，Selenium2Library 必须导入到你的 Robot 测试集中（Test Suite,引入操作参见 [Imports](#) 部分），之后中“*Open Browser*”关键字必须指定浏览器打开你预期 URL 路径。。

---首先注意的是 1.7.0 版本发布后重要的变化---

## 定位或指定元素

Selenium2Library 中需要在页面上查找元素的所有关键字都接受一个参数，无论是 Locator 还是当前的 webelement。Locator 是一个字符串，描述如何使用指定不同元素位置策略的语法来定位元素。webelement 元素的表示形式，是一个保存 WebElement 实例的变量。

## 使用 Locators

默认情况下，提供一个有效的 locator 值，它就可以匹配特定类型元素的某些关键属性。例如：Id 和 Name 是所有元素关键属性。id 和 Locator 属性可以适用于所有元素定位，例如：

```
Click Element my_element
```

当然，并不是说只能使用 id 和 name 属性来进行定位，在 Selenium2Library 中想去查找某个页面元素支持多种不同的方式，支持的策略如下：

Strategy	实例	Description 说明
----------	----	----------------

identifier	Click Element   identifier=my_element	Matches by @id or @name attribute
id	Click Element   id=my_element	Matches by @id attribute
name	Click Element   name=my_element	Matches by @name attribute
xpath	Click Element   xpath=//div[@id='my_element']	Matches with arbitrary XPath expression
dom	Click Element   dom=document.images[56]	Matches with arbitrary DOM express
link	Click Element   link=My Link	Matches anchor elements by their link text
partial link	Click Element   partial link=y Lin	Matches anchor elements by their partial link text
css	Click Element   css=div.my_class	Matches by CSS selector
jquery	Click Element   jquery=div.my_class	Matches by jQuery/sizzle selector
sizzle	Click Element   sizzle=div.my_class	Matches by jQuery/sizzle selector
tag	Click Element   tag=div	Matches by HTML tag name
default*	Click Link   default=page?a=b	Matches key attributes with

		value after first '='
--	--	-----------------------

default\*显式指定的默认策略，如果定位需要通过属性匹配元素，只需要属性和中值中间包含'='。以下会失败，因为它看起来好像页面？是指定的查找策略：

**Click Link**     **page?a=b**

这可以通过改变固定的定位器：

**Click Link**     **default=page?a=b**

## 使用 webelements

从 Selenium2Library1.7 版本开始，可以通过包含 WebElement 定位字符串参数进行定位。为了获取一个 WebElement 定位字符变量，可以使用 Get WebElement 关键字。例如：

<b>\${elem} =</b>	<b>Get WebElement</b>	<b>id=my_element</b>
<b>Click Element</b>	<b>\${elem}</b>	

查找表，表行，列，等等。

Table 相关的关键字，如 Table Should Contain,使用方法是不同的。默认情况下，当提供 Table 定位值时，它会搜索指定 Table ID 属性的 Table。例如：

**Table Should Contain**     **my\_table**     **text**

更复杂的 Table 查找定位策略支持方式：

Strategy	Example	Description
css	Table Should Contain /css=table.my_class /text	Matches by @id or @name attribute
xpath	Table Should Contain /xpath=//table[@name="my_table"] /text	Matches by @id or @name attribute

# 自定义定位器

如果需要比默认定位器进行更复杂的元素查找定位，则可以创建自定义查找策略。使用自定义定位器是一个两部分的过程。首先，创建一个可返回执行操作的WebElement 的关键字。

Custom Locator Strategy	[Arguments]	\$(browser)	\$(criteria)	\$(tag)	\$(constraints)
	\$(retVal)=	Execute Javascript	return window.document.getElementById('\$(criteria)');		
	[Return]	\$(retVal)			

该关键字的基本功能重新实现 ID 定位，其中\$(browser) 是对 webdriver 的实例的引用，\$(criteria) 定位器文本内容，格式是（所有定位内容在等号（=）之后）。如果要使用当前自定义定位器，必须要先进行注册，详见关键字 Add Location Strategy。

Add Location Strategy	custom	Custom Locator Strategy
-----------------------	--------	-------------------------

Add Location Strateg 的第一个参数是指定查找策略的名称（必须是唯一的）。自定义定位器在注册后，自定义的定位器用法与其他定位器用法相同。详见 Add Location Strategy 的更多细节。

## 超时等待

包含多种 wait ...关键字。所有的超时等待参数都是可选的。可以将这些 wait 关键字的值设置为全局变量，使 Selenium2Library 共用同一等待时间（参见 Set Selenium Timeout 关键字）。同样的超时等待可以支持 Execute Async Javascript 关键字。

超时等待时间可以给定一个明确的值来定义 (比如 0.5 或 42) , 或 or in Robot Framework's 的时间格式 (e.g. '1.5 seconds' (1.5 秒) or '1 min 30 s' (1 分 3 0 秒) )

更多的信息可以参照:

<http://robotframework.googlecode.com/svn/trunk/doc/userguide/RobotFrameworkUserGuide.html>

## Import 引入策略

参数:

timeout=5.0

implicit\_wait=0.0

run\_on\_failure=Capture Page Screenshot

描述:

Selenium2Library 可以导入可选参数。

默认的超时等待可以适用于所有的等待行为。它也可以通过 Set Selenium Timeout 来进行配置。

'implicit\_wait' 是隐式超时。可以通过 Set Selenium Implicit Wait 进行配置。更多的隐式等待信息请参照:

[http://seleniumhq.org/docs/04\\_webdriver\\_advanced.html#explicit-and-implicit-waits](http://seleniumhq.org/docs/04_webdriver_advanced.html#explicit-and-implicit-waits)

当 Selenium2Library 某个关键字执行失败时将指定运行关键字

run\_on\_failure。

默认可以使用 Capture Page Screenshot 来截取当前页面内容。使用值 'Nothing'可以完全禁用该功能。查看关键 Register Keyword To Run On Failure 来获得更多资料。

示例：

Library / Selenium2Library / 15	# Sets default timeout to 15 seconds
Library / Selenium2Library / 0 / 5	# Sets default timeout to 0 seconds and default implicit_wait to 5 seconds
Library / Selenium2Library / 5 / run_on_failure=Log Source	# Sets default timeout to 5 seconds and runs <a href="#">Log Source</a> on failure
Library / Selenium2Library / implicit_wait=5 / run_on_failure=Log Source	# Sets default implicit_wait to 5 seconds and runs <a href="#">Log Source</a> on failure
Library / Selenium2Library / timeout=10 / run_on_failure=Nothing	# Sets default timeout to 10 seconds and does nothing on failure

.

## 关键字

### 关键字：**Add Cookies**

参数：

Name,value,  
path=none,  
domain=none,  
secure=none,  
expiry=none

描述：

添加一个cookie到你当前的session中。Name和value变量为必输项，path、domain和secure是选输项。

### 关键字：**Add Location Strategy**



参数：

**Strategy\_name,Strategy\_keywork,**  
**Persist=False**

描述：

对用户自定义关键字添加元素定位变量及值信息。默认情况下，自定义关键字超出当前作用域的使用时，当前元素定位变量会失效。对元素定位变量设置为任意非空值的定位变量名后，会将当前用户自定义关键字的元素定位变量在整个项目的测试案例中生效。

如果尝试重复添加包含相同的元素定位变量的关键字，会导致当前关键字失败。

当前定位关键字实例：

Custom Locator Strategy	[Arguments]	\${browser}	\${criteria}		
	\${retVal}=	Execute Javascript	return window.document.getElementById('\${criteria}');		
	[Return]	\${retVal}			

使用实例：

Add Location Strategy	custom	Custom Locator Strategy
Page Should Contain Element	custom=my_id	

如果需要删除元素定位变量，详见 [Remove Location Strategy](#) .

## 关键字：**Alert Should Be Present**

参数：

**Text=**

描述：

校验是否有 **alert** 提示框出现。如果提示框信息不为空，则它将校验 **alert** 信息是否与 **text** 一致。

## 关键字：**Assign Id To Element**

参数：

**Locator , id**

描述：

分配一个临时属性给特殊标识元素。当遇到定位元素复杂/繁琐的 **Xpath** 表达式时该方法就很有必要了。

示例：

Assign ID to Element	xpath=//div[@id="first_div"]	my id
Page Should Contain Element	my id	

**关键字：** **Capture Page Screenshot**

参数：

**Filename=None**

描述：

截取当前页面并将其嵌入到日志。

**Filename** 参数指定写入日志的截图名称。

如果该参数为空，那么截图名称将以“selenium-screenshot-<counter>.png”格式保存到 **RF** 日志文件所在目录下。

**关键字：** **Checkbox Should Be Selected**

参数：

**Locator**

描述：

通过定位标识出的 **checkbox** 元素已被选择或者勾选。

**Checkboxes** 关键字属性为 **id** 和 **name**。

## 关键字：**Checkbox Should Not Be Selected**

参数：

**Locator**

描述：

通过定位标识出的 **checkbox** 元素已被选择或者勾选。

**Checkboxes** 关键字属性为 **id** 和 **name**。

## 关键字：**Choose Cancel On Next Confirmation**

参数：

无

描述：

系统弹出框后，点击“**cancel**”按钮。

## 关键字：**Choose File**

参数：

**Locator,**  
**file\_path**

描述：

在定位的输入框中输入 **file\_path** 信息。

该关键字常用在上传文件中。被指定的文件的 **file\_path** 必须可用。

示例：

Choose File	my_upload_field	/home/user/files/trades.csv
-------------	-----------------	-----------------------------

## 关键字：**Choose Ok On Next Confirmation**

参数：

无

描述：

系统弹出框后，点击“OK”按钮。

当前关键字与 "**Choose Cancel On Next Confirmation**" 相反。注意，Selenium 的覆盖 `window.confirm()` 函数通常会自动返回 `true`，就像用户手动单击 OK 一样，所以你不需要使用这个关键字，除非由于某种原因需要改变主意之前你在下一次确认。在任何确认后，继续使用默认行为，为今后确认，自动返回 `true` (OK)，除非/直到你明确地使用 **Choose Ok On Next Confirmation** 进行下一步确认。

## 关键字：**Clear Element Text**

参数：

Locator,

描述：

清除由定位器指定元素的文本内容。

请参阅有关本文介绍的元素定位内容。

## 关键字：**Click Button**

参数：

Locator,

file\_path

描述：

点击被定位按钮。

buttons 关键字属性为 id , name 和 value。

**关键字：** Click Element

参数：

Locator,

file\_path

描述：

点击被定位按钮。

Element 关键字属性为 id , name 和 value。

**关键字：** Click Element At Coordinates

参数：

Locator ,

Xoffset ,

Yoffset

描述：

通过 x , y 坐标点击页面元素。

## 关键字：**Click Image**

参数：

Locator

描述：

点击被定位图片。

Images 关键字属性为 id , src 和 alt。

## 关键字：**Click Link**

参数：

Locator

描述：

点击被定位链接。

Links 关键字属性为 id , name , href 和 link text。

## 关键字：**Close All Browsers**

参数：

无

描述：

关闭全部打开状态的浏览器并重置缓存。

执行完该关键字后，再次打开浏览器，浏览器对应 index 值从 1 开始计算。

关键字必须在 test 或者 suite 的 teardown 时使用，为了确保所有浏览器已被关闭。

## 关键字：**Close Browsers**

参数：

无

描述：

关闭当前浏览器。

## 关键字：**Close Window**

参数：

无

描述：

关闭当前打开的弹出框。

## 关键字：**Confirm Action**

参数：

Locator,  
file\_path

描述：

接触当前显示 confirm 对话框并返回它的 message 信息。

默认情况下，该关键字选择对话框中的“OK”项。如果需要选择“Cancel”项，请在该关键字执行前调用“**Choose Cancel On Next Confirmation**”，以免对话框已关闭。

示例：

Click Button	Send	# Shows a confirmation dialog
\${message}=	Confirm Action	# Chooses Ok
Should Be Equal	\${message}	Are your sure?
Choose Cancel On Next Confirmation		
Click Button	Send	# Shows a confirmation dialog
Confirm Action		# Chooses Cancel

## 关键字：**Current Frame Contains**

参数：

Text,

loglevel=info

描述：

指定当前 frame 下的 text。

可以查看 **"Page Should Contain"** 关于 loglevel 参数的解释。

## 关键字：**Current Frame Should Not Contain**

参数：

Text,

loglevel=info

描述：

指定当前 frame 下的 text。

可以查看 **"Page Should Contain"** 关于 loglevel 参数的解释。



## 关键字：**Create Webdriver**

参数：

```
driver_name,  
alias=None,  
kwargs={},  
**init_kwargs
```

描述：

webdriver 创建一个实例。

类似 Open Browser 关键字,但允许 WebDriver 的\_\_init\_\_传递参数。Open Browser 关键字是在可行的情况下首选创建 Webdriver。

执行关键字实例化后会返回这个浏览器索引的方便后续使用,方便使用索引值进行切换。索引从 1 开始,重置索引时需要关闭所有浏览器再使用该关键字。

DRIVER\_NAME 必须是 selenium.webdriver 调用 webdriver 正确的名称。 WebDriver 名称包括:Firefox、Chrome,Opera,Safari,PhantomJS 和 Remote。

使用该关键字参数来指定你想要的参数传给 WebDriver 的\_\_init\_\_。 未以任何方式处理的参数值传递之前。Robot Framework< 2.8,不支持关键字参数,创建一个关键字字典和作为参数传递 kwargs。 看到 Selenium API Documentation

示例：

# use proxy for Firefox			
\${proxy}=	Evaluate	sys.modules['selenium.webdriver'].Proxy()	sys, selenium.webdriver
\${proxy.http_proxy}=	Set Variable	localhost:8888	
Create Webdriver	Firefox	proxy=\${proxy}	
# use a proxy for PhantomJS			
\${service_args}=	Create List	--proxy=192.168.132.104:8888	
Create Webdriver	PhantomJS	service_args=\${service_args}	

Robot Framework< 2.8 的示例：

# debug IE driver					
\${kwargs}=	Create Dictionary	log_level	DEBUG	log_file	%{HOMEPATH}%{}/ie.log
Create Webdriver	Ie	kwargs=\${kwargs}			

## 关键字： Delete All Cookies

参数：

无

描述：

删除所有 cookies 信息。

## 关键字： Delete Cookie

参数：

Name

描述：

删除与参数 name 匹配的 cookies 信息。

当未匹配到对应 name 的 cookies 信息，则不执行。

## 关键字： Dismiss Alert

参数：

accept=True

描述：

如果 accept 为 True,alert 窗口被定位，点击 alert 窗口的 ' 确认 ' 按钮！

如果如果 accept 为 False,点击 alert 窗口的 ' 取消 ' 按钮。

注意 ,后续关键字将运行失败 ,除非通过此关键或者其他关键字 ,比如 **Get Alert Message**。

## 关键字：**Double Click Element**

参数：

Locator

描述：

双击被定位的 element 元素。

属性为 id 和 name。

## 关键字：**Drag And Drop**

参数：

source,

target

描述：

拖拽被定位 element 资源。

Element 可被移动到其他目标参数之上。

目标为一个元素定位器，可以将对象拖拽并放下。

示例：

Drag And Drop	elem1	elem2	# Move elem1 over elem2.
---------------	-------	-------	--------------------------

## 关键字：**Drag And Drop By Offset**

参数：

Source,

xoffset,

yoffset

描述：

按照坐标拖拽。

示例：

Drag And Drop	myElem	50	-35	# Move myElem 50px right and 35px down.
---------------	--------	----	-----	---

**关键字：** **Element Should Be Disabled**

参数：

Locator

描述：

指定 element 元素不可见。

指定属性为 id 和 name。

**关键字：** **Element Should Be Enabled**

参数：

Locator

描述：

指定 element 元素可用。

指定属性为 id 和 name。

## 关键字：**Element Should (Not) Be Enable**

参数：

Locator

描述：

指定 element 元素(不)可用。

指定属性为 id 和 name。

## 关键字：**Element Should Be Visible**

参数：

Locator,

Message=

描述：

指定 element 元素可见。

于此，可见的意思是逻辑可见，而非视觉可见。

比如：一个 element 元素的 display 属性为 none，这个是逻辑不可见，所以使用这个关

键字后，针对该元素的操作将失败。

## 关键字：**Element Should Contain**

参数：

Locator,

Expected,

Message=

描述：

指定的 **element** 元素必须包含期望 **text** 内容。

如果你希望校验一个准确匹配（非模糊匹配）的 **Text** 信息，可以使用该关键字。

**Message** 可以用来覆盖缺省的报错信息。

## 关键字：**Element Should Not Be Visible**

参数：

**Locator**

描述：

指定 **element** 元素(不)可见。

指定属性为 **id** 和 **name**。

## 关键字：**Element Should Not Contain**

参数：

**Locator,**

**Expected,**

**Message=**

描述：

指定的 **element** 元素必须不包含期望 **text** 内容。

**Message** 可以用来覆盖缺省的报错信息。

元素定位的 **Id** 和 **Name** 的信息，请参阅关键字“**Element Should Contain**”的内容

## 关键字：Element Text Should Be

参数：

Locator,  
Expected,  
Message=

描述：

通过定位器指定 element 必须准确包含预期 text 内容。

与 “element should contain” 比较，该关键字仅支持精确匹配，不支持模糊字符串匹配。

## 关键字：Execute Async Javascript

参数：

\*code

描述：

执行异步 java 脚本代码。

代码可以包含多行代码但是必须包含返回信息。

代码可以在测试数据中被分割为多个单元。即便如此，彼此关联的组件间不允许任意增加空间。

如果被执行代码的路径为绝对路径，那么 javascript 将从这个绝对路径文件中读取并运行。

示例：

Execute Async JavaScript	var callback = arguments[arguments.length - 1]; window.setTimeout(callback, 2000);	
Execute Async JavaScript	\${CURDIR}/async_js_to_execute.js	
\${retval}=	Execute Async JavaScript	
...	var callback = arguments[arguments.length - 1];	
...	function answer() {callback("text");};	
...	window.setTimeout(answer, 2000);	
Should Be Equal	\${retval}	text

## 关键字：Execute Javascript

参数：

\*code

描述：

执行 javascript 代码。

代码可以包含多行代码但是必须包含返回信息。

代码可以在测试数据中被分割为多个单元。即便如此，彼此关联的组件间不允许任意增加空间。

如果被执行代码的路径为绝对路径，那么 javascript 将从这个绝对路径文件中读取并运行。

示例：

Execute JavaScript	window.my_js_function(' arg1', ' arg2' )	
Execute JavaScript	\${CURDIR}/js_to_execute.js	
\${sum}=	Execute JavaScript	return 1 + 1;
Should Be Equal	\${sum}	\${2}

## 关键字：**Focus**

参数：

Locator

描述：

设置定位器定位元素为焦点目标。

## 关键字：**Frame Should Contain**

参数：

Locator,

Text,

Loglevel=info

描述：

指定的 frame 必须包含期望 text 内容。

Loglevel 参数的使用可以见 “**Page Should Contain**” 的解释。

## 关键字：**Get Alert Message**

参数：

无

描述：

返回当前 js alert 窗口的 text 信息。

如果不存在 alert 窗口则执行该关键字失败。

注意，后续关键字将运行失败，除非通过此关键或者其他关键字，比如 **Get Alert Message**，将 alert 窗口关闭。



## 关键字：**Get All Links**

参数：

无

描述：

返回当前页面的所有 link 类型的对象，并存入 list

## 关键字：**Get Cookie Value**

参数：

Name

描述：

通过 name 查找对应 cookie 的值，并返回。

如果 cookie 不存在，则关键字执行失败。

## 关键字：**Get Cookies**

参数：

Name

描述：

返回当前页面的所有 Cookie。

## 关键字：**Get Element Size**

参数：

locator

描述：

返回由定位到 element 的宽度和高度。

返回元素的宽度和高度。 如果在页面上匹配到该元素，则失败。

## 关键字：**Get Element Attribute**

参数：

Attribute\_locator

描述：

返回 element 属性值。

属性定位器 attribute\_locator 由标记@和属性名称组成，比如 “element\_id@class”

## 关键字：**Get Horizontal Position**

参数：

Locator

描述：

返回定位 element 元素的水平位置。

位置为整型类型，从屏幕左上角算起。

请参阅：[Get Vertical Position](#)

## 关键字：**Get List Items**

参数：

Locator

描述：

从已定位 list 对象中返回该 list 下的所有 item 项。

用于关键字所选中列表的 Lists 和 combo boxes 的元素。使用选择关键属性是 ID 和名称，定位列表中内容。

请参阅有关本文介绍的元素定位内容。

## 关键字：**Get location**

参数：

无

描述：

返回当前位置信息

## 关键字：**Get Matching Xpath Count**

参数：

无

描述：

返回匹配 xpath 格式的元素数量。

如果你想对元素匹配进行断言，请使用 `xpath Should Match X Times`。

## 关键字：**Get Selected List Lable**

参数：

Locator

描述：

返回定位器定位的 list 下的被选元素的标签信息。

Select list 关键字可以在 lists 和 combo boxes 二者使用。它的关键字属性为 id 和 name。

## 关键字：**Get Selected List Value**

参数：

Locator

描述：

返回定位器定位的 list 下的被选元素的 value 值。

Select list 关键字可以在 lists 和 combo boxes 二者使用。它的关键字属性为 id 和 name。

## 关键字：**Get Selected List Values**

参数：

Locator

描述：

返回定位器定位的 list 下的被选元素的 value 值(以 list 形式保存)。

如果没有选项则关键字执行失败。

elect list 关键字可以在 lists 和 combo boxes 二者使用。它的关键字属性为 id 和 name。

## 关键字：**Get Selenium Implicit Wait**

参数：

无

描述：

设置 selenium 隐式等待时间。

Selenium 默认等待 30 秒。

该关键字解释说明请看 [Set Selenium Implicit Wait](#) 关键字。

## 关键字：[Get Selenium Speed](#)

参数：

无

描述：

获取执行每一条 selenium 命令时的延迟时间。

该关键字解释说明请看 [Set Selenium Speed](#) 关键字。

## 关键字：[Get Selenium Timeout](#)

参数：

无

描述：

获取该关键执行之前使用过的各类关键字的 timeout 时间。

该关键字解释说明请看 [Set Selenium Timeout](#) 关键字。

## 关键字：[Get Source](#)

参数：

无

描述：

获取当前页面全部 html 源。

## 关键字：[Get Table Cell](#)

参数：

Table\_locator,

Row,

Column,

Loglevel=info

描述：

返回指定表格单元的内容。

行和列起始从 1 开始。

头尾行包含在 count 里。这就意味着表头和表尾也可以通过该关键字获取。

详见 **Page Should Contain** 关键字有关日志级别参数的解释。

## 关键字：**Get Text**

参数：

locator,

描述：

返回定位器定位元素的 text 值。

请参阅有关本文介绍的元素定位内容。

## 关键字：**Get Title**

参数：

locator,

描述：

返回当前页面 title 属性值。

## 关键字：**Get Value**

参数：

locator,

描述：

返回当前定位器定位元素 value 属性值。

请参阅有关本文介绍的元素定位内容。

## 关键字：**Get Vertical Position**

参数：

locator,

描述：

返回定位 element 元素的垂直位置。

请参阅 **Get Horizontal Position** 关键字

## 关键字：**Get Webelement**

参数：

locator

描述：

返回页面第一个根据 locator 匹配到的 WebElement 对象。

请参阅有关本文介绍的元素定位内容。

## 关键字：**Get Window Identifiers**

参数：

无

描述：

返回并记录将所有可识别 windows 的 Id 属性。

## 关键字：**Get Window Names**

参数：

无

描述：

返回并记录将所有可识别 windows 的名字。

## 关键字：**Get Window Position**

参数：

无

描述：

返回当前窗口位置的 X 轴和 Y 轴。

实例：

<code>\${x}</code>	<code>\${y}=</code>	Get Window Position
--------------------	---------------------	---------------------

## 关键字：**Get Window Size**

参数：

无

描述：

返回当前 window 界面的尺寸，先宽度后高度。

示例：

<code>\${width}</code>	<code>\${height}=</code>	Get Window Size
------------------------	--------------------------	-----------------

## 关键字：**Get Window Title**

参数：

无

描述：

返回并记录将所有可识别 windows 的标题。

## 关键字：**Go Back**

参数：

无

描述：

模拟用户点击浏览器界面的“后退”按钮。

## 关键字：**Go To**

参数：

Url

描述：

使当前激活浏览器页面跳转至用户所提供的 url 地址。

## 关键字：**Input Password**

参数：

Locator,

Text

描述：

将已给出密码项输入到定位文本域中。

与 “Input Text” 的区别再于该关键字无法记录所给的密码信息。

## 关键字：**Input Text**

参数：

Locator,

Text

描述：

将给出的文本内容输入到定位文本域中。请参阅有关本文介绍的元素定位内容。

## 关键字：**Input Text Into Prompt**

参数：

Text

描述：

将指定的 Text 输入到 Prompt 弹窗中。

## 关键字：**List Windows**



参数：

Text

描述：

将所有当前窗口 handles 以列表形式返回

## 关键字：**List Selection Should Be**

参数：

Locator,

\*items

描述：

验证被选 list 控件中的 selection 项是否与参数\*items 匹配。

如果你希望测试空选项的情况，通常情况下无需给出 items 值。

该关键字同时可以在 lists 和 combo boxes 中使用，常用属性为 id 和 name。

请参阅有关本文介绍的元素定位内容。

## 关键字：**List Should Hava No Selections**

参数：

Locator

描述：

验证被选 list 控件中的 selection 项是否未选择。

该关键字同时可以在 lists 和 combo boxes 中使用，常用属性为 id 和 name。

请参阅有关本文介绍的元素定位内容。

## 关键字：**Location Should Be**

参数：

Url

描述：

校验当前 URL 地址是否与所给 url 地址一致。

## 关键字：**Location Should Contain**

参数：

Expected

描述：

校验当前 URL 地址是否包含预期值。

**关键字：** Log Location

参数：

无

描述：

记录并返回当前位置。

**关键字：** Locator Should Match X Times

参数：

locator,

expected\_locator\_count,

message=,

loglevel=INFO

描述：

验证该页面包含位于由给出定位器元件的给定数量。

请参阅有关本文介绍的元素定位内容。

详见 **Page Should Contain Element** 关键字有关日志级别参数的解释。

**关键字：** Log Location

参数：

无

描述：

记录并返回当前位置。

**关键字：** Log Source

参数：

Loglevel=INFO

描述：

记录并返回当前页面或者 frame 框内的全部 html 资源。

该 loglevel 参数定义了使用的日志级别。有效的日志级别是 WARN, INFO (默认项), DEBUG, and NONE (无日志记录)。

**关键字：** **Log Title**

参数：

无

描述：

记录并返回当前页面 title 值。

**关键字：** **Maximize Browser Window**

参数：

无

描述：

将当前浏览器最大化。

**关键字：** **Mouse Down**

参数：

locator

描述：

模拟用户进行鼠标左键操作。

进行点击操作但不释放。

关键字属性为 id 和 name。

请参阅有关本文介绍的元素定位内容。

请参阅关键字 **Mouse Down On Image** 和 **Mouse Down On Link** 数的解释。

**关键字：** **Mouse Down On Image**

参数：

locator

描述：

模拟鼠标点击 image 事件。

定位使用 Image 元素的 id, src 和 alt 属性。

请参阅有关本文介绍的元素定位内容。

## 关键字：**Mouse Down On Link**

参数：

locator

描述：

模拟鼠标点击 Link 事件。

定位使用 Link 元素 id, name, href 和 link text 属性。

请参阅有关本文介绍的元素定位内容。

## 关键字：**Mouse Out**

参数：

locator

描述：

模拟将鼠标指针从定位元素上移开事件。

定位使用 Element 元素的 ID 和 Name 属性。

请参阅有关本文介绍的元素定位内容。

## 关键字：**Mouse Over**

参数：

locator

描述：

模拟鼠标悬停动作。

定位使用 Element 元素的 ID 和 Name 属性。

请参阅有关本文介绍的元素定位内容。

## 关键字：**Mouse Up**

参数：

locator

描述：

模拟释放鼠标左键动作。

定位使用 Element 元素的 ID 和 Name 属性。

请参阅有关本文介绍的元素定位内容。

## 关键字：**Open Browser**

参数：

Url,

Browser=firefox,

Alias=none,

Remote\_url=false,

Desired\_capabilities=none,

Ff\_profile\_dir=none

描述：

打开一个新的 browser，并转至所给 url 地址。

返回新打开 browser 的 index 值，该值可以在之后的操作进行 browser 切换操作。Index 值从 1 开始，当执行了 close all browsers 关键字后，资源释放。

可选参数 alias 是浏览器实例的别名，它能够在浏览器间切换时使用（类似于 index），可以查看关键字“Switch Browser”获得更多细节。

必填参数 Browser 有如下值可以使用：

firefox	FireFox
ff	FireFox
internetexplorer	Internet Explorer
ie	Internet Explorer
googlechrome	Google Chrome
gc	Google Chrome
chrome	Google Chrome
opera	Opera
phantomjs	PhantomJS
htmlunit	HTMLUnit
htmlunitwithjs	HTMLUnit with Javascript support
android	Android
iphone	Iphone
safari	Safari

可选的'remote\_url' "URL 是 selenium 远程服务器地址例如 "http://127.0.0.1/wd/hub" 。如果用于 remote 指定一个值 ,还可以指定'desired\_capabilities' ,这是一个字符串的形式 KEY1 :val1 中 ,KEY2 : val2 中 , 将用于指定 desired\_capabilities 到远程服务器。这是做这样的事情指定代理服务器的 Internet Explorer 或如果您使用 saucelabs.com 指定浏览器和操作系统是非常有用。 'desired\_capabilities'也可以是一个 Python 的 dictionary ( 字典 ) , 以允许更复杂的配置。

## 关键字： Open Context Menu

参数：

Locator

描述：

打开上下文菜单。

## 关键字： Page Should Contain

参数：

Text,

Loglevel=info

描述：

验证当前页面包含 "text" 信息。

如果该关键字失败，它将自动的使用 loglevel 记录这个页面资源。

不给出 loglevel 参数，将无法记录日志。

该 loglevel 参数定义了使用的日志级别。有效的日志级别是 WARN, INFO (默认项), DEBUG, and NONE (无日志记录).

## 关键字： Page Should Contain Button

参数：

Locator,  
Message=,  
Loglevel=info

描述：

验证当前页面包含某 button。

该关键字所查找的 buttons 一般情况下是由 input 或者 button tag 提供的。

定位使用 Buttons 元素的 ID,Name 和 Value 属性。

请参阅有关本文介绍的元素定位内容。

该 loglevel 参数定义了使用的日志级别。有效的日志级别是 WARN, INFO (默认项), DEBUG, and NONE (无日志记录)。

## 关键字： Page Should Contain Checkbox

参数：

Locator,  
Message=,  
Loglevel=info

描述：

验证当前页面包含某 Checkbox。

定位使用 Checkbox 元素 ID 和 Name 属性

请参阅有关本文介绍的元素定位内容。

该 loglevel 参数定义了使用的日志级别。有效的日志级别是 WARN, INFO (默认项), DEBUG, and NONE (无日志记录)。

## 关键字： Page Should Contain Element

参数：

Locator,  
Message=,  
Loglevel=info

描述：

验证当前页面包含某 Element。

定位使用 Element 元素 ID 和 Name 属性

请参阅有关本文介绍的元素定位内容。

该 loglevel 参数定义了使用的日志级别。有效的日志级别是 WARN, INFO (默认项), DEBUG, and NONE (无日志记录)。

## 关键字：**Page Should Contain Image**

参数：

Locator,

Message=,

Loglevel=info

描述：

验证当前页面包含某 Image。

定位使用 Image 元素 ID,alt 和 src 属性

请参阅有关本文介绍的元素定位内容。

该 loglevel 参数定义了使用的日志级别。有效的日志级别是 WARN, INFO (默认项), DEBUG, and NONE (无日志记录)。

## 关键字：**Page Should Contain Link**

参数：

Locator,

Message=,

Loglevel=info

描述：

验证当前页面包含某 Link。

定位使用 Link 元素 ID,href 和 link text 属性

请参阅有关本文介绍的元素定位内容。

该 loglevel 参数定义了使用的日志级别。有效的日志级别是 WARN, INFO (默认项), DEBUG, and NONE (无日志记录)。



## 关键字：**Page Should Contain List**

参数：

Locator,  
Message=,  
Loglevel=info

描述：

验证当前页面包含某 List。

定位使用 List 元素 ID 和 Name 属性

请参阅有关本文介绍的元素定位内容。

该 loglevel 参数定义了使用的日志级别。有效的日志级别是 WARN, INFO (默认项), DEBUG, and NONE (无日志记录)。

## 关键字：**Page Should Contain Radio Button**

参数：

Locator,  
Message=,  
Loglevel=info

描述：

验证当前页面包含某 Radio Button。

定位使用 Radio Button 元素 ID , Value 和 Name 属性

请参阅有关本文介绍的元素定位内容。

该 loglevel 参数定义了使用的日志级别。有效的日志级别是 WARN, INFO (默认项), DEBUG, and NONE (无日志记录)。

## 关键字：**Page Should Contain TextField**

参数：

Locator,  
Message=,  
Loglevel=info

描述：

验证当前页面包含某 TextField。

定位使用 Radio Button 元素 ID 和 Name 属性

请参阅有关本文介绍的元素定位内容。

该 loglevel 参数定义了使用的日志级别。有效的日志级别是 WARN, INFO (默认项), DEBUG, and NONE (无日志记录)。

## 关键字：**Page Should (Not) Contain**

参数：

Text,

Loglevel=info

描述：

验证当前页面(不)包含 “text” 信息。

如果该关键字失败，它将自动的使用 loglevel 记录这个页面资源。

不给出 loglevel 参数，将无法记录日志。

该 loglevel 参数定义了使用的日志级别。有效的日志级别是 WARN, INFO (默认项), DEBUG, and NONE (无日志记录)。

## 关键字：**Page Should (Not) Contain Button**

参数：

Locator,

Message=,

Loglevel=info

描述：

验证当前页面(不)包含某 button。

该关键字所查找的 buttons 一般情况下是由 input 或者 button tag 提供的。

定位使用 Buutons 元素的 ID,Name 和 Value 属性。

请参阅有关本文介绍的元素定位内容。

该 loglevel 参数定义了使用的日志级别。有效的日志级别是 WARN, INFO (默认项), DEBUG, and NONE (无日志记录)。

## 关键字：**Page Should (Not) Contain Checkbox**

参数：

Locator,  
Message=,  
Loglevel=info

描述：

验证当前页面(不)包含某 Checkbox。

定位使用 Checkbox 元素 ID 和 Name 属性

请参阅有关本文介绍的元素定位内容。

该 loglevel 参数定义了使用的日志级别。有效的日志级别是 WARN, INFO (默认项), DEBUG, and NONE (无日志记录)。

## 关键字：**Page Should (Not) Contain Element**

参数：

Locator,  
Message=,  
Loglevel=info

描述：

验证当前页面(不)包含某 Element。

定位使用 Element 元素 ID 和 Name 属性

请参阅有关本文介绍的元素定位内容。

该 loglevel 参数定义了使用的日志级别。有效的日志级别是 WARN, INFO (默认项), DEBUG, and NONE (无日志记录)。

## 关键字：**Page Should (Not) Contain Image**

参数：

Locator,  
Message=,  
Loglevel=info

描述：

验证当前页面(不)包含某 Image。

定位使用 Image 元素 ID,alt 和 src 属性

请参阅有关本文介绍的元素定位内容。

该 loglevel 参数定义了使用的日志级别。有效的日志级别是 WARN, INFO (默认项), DEBUG, and NONE (无日志记录)。

## 关键字：**Page Should (Not) Contain Link**

参数：

Locator,  
Message=,  
Loglevel=info

描述：

验证当前页面(不)包含某 Link。

定位使用 Link 元素 ID,href 和 link text 属性

请参阅有关本文介绍的元素定位内容。

该 loglevel 参数定义了使用的日志级别。有效的日志级别是 WARN, INFO (默认项), DEBUG, and NONE (无日志记录)。

## 关键字：**Page Should (Not) Contain List**

参数：

Locator,  
Message=,  
Loglevel=info

描述：

验证当前页面(不)包含某 List。

定位使用 List 元素 ID 和 Name 属性

请参阅有关本文介绍的元素定位内容。

该 loglevel 参数定义了使用的日志级别。有效的日志级别是 WARN, INFO (默认项), DEBUG, and NONE (无日志记录)。

## 关键字：**Page Should (Not) Contain Radio Button**

参数：

Locator,  
Message=,  
Loglevel=info

描述：

验证当前页面(不)包含某 Radio Button。

定位使用 Radio Button 元素 ID , Value 和 Name 属性

请参阅有关本文介绍的元素定位内容。

该 loglevel 参数定义了使用的日志级别。有效的日志级别是 WARN, INFO (默认项), DEBUG, and NONE (无日志记录)。

## 关键字：**Page Should (Not) Contain TextField**

参数：

Locator,  
Message=,  
Loglevel=info

描述：

验证当前页面(不)包含某 TextField。

定位使用 Radio Button 元素 ID 和 Name 属性

请参阅有关本文介绍的元素定位内容。

该 loglevel 参数定义了使用的日志级别。有效的日志级别是 WARN, INFO (默认项), DEBUG, and NONE (无日志记录)。

## 关键字：**Press Key**

参数：

Locator,  
Key ,  
Wait=

描述：

模拟用户键盘操作。

Key 既可以是单个字符，也可以是 ASCII 码，但是需要在前加入 ' \' 。

请参阅有关本文介绍的元素定位内容。

有时，这种关键字不会触发点击元素的正确 JavaScript 事件。在这种情况下，**Press Key Native** 可以作为一种解决方法。

示例：

Press Key	text_field	q	
Press Key	login_button	\13	# ASCII code for enter key

这里的 ' \13 ' 对应 ASCII 码的回车键。

ASSCII 码列表：

```
NULL      ='\ue000'
CANCEL    ='\ue001'# ^break
HELP      ='\ue002'
BACKSPACE ='\ue003'
BACK_SPACE =BACKSPACE
TAB       ='\ue004'
CLEAR     ='\ue005'
RETURN    ='\ue006'
ENTER     ='\ue007'
SHIFT     ='\ue008'
LEFT_SHIFT =SHIFT
CONTROL   ='\ue009'
LEFT_CONTROL =CONTROL
ALT       ='\ue00a'
LEFT_ALT  =ALT
PAUSE     ='\ue00b'
ESCAPE    ='\ue00c'
SPACE     ='\ue00d'
PAGE_UP   ='\ue00e'
PAGE_DOWN ='\ue00f'
END       ='\ue010'
HOME      ='\ue011'
LEFT      ='\ue012'
ARROW_LEFT =LEFT
UP        ='\ue013'
ARROW_UP  =UP
RIGHT     ='\ue014'
```

ARROW\_RIGHT =RIGHT  
DOWN ='\ue015'  
ARROW\_DOWN =DOWN  
INSERT ='\ue016'  
DELETE ='\ue017'  
SEMICOLON ='\ue018'  
EQUALS ='\ue019'  
NUMPAD0 ='\ue01a'# number pad keys  
NUMPAD1 ='\ue01b'  
NUMPAD2 ='\ue01c'  
NUMPAD3 ='\ue01d'  
NUMPAD4 ='\ue01e'  
NUMPAD5 ='\ue01f'  
NUMPAD6 ='\ue020'  
NUMPAD7 ='\ue021'  
NUMPAD8 ='\ue022'  
NUMPAD9 ='\ue023'  
MULTIPLY ='\ue024'  
ADD ='\ue025'  
SEPARATOR ='\ue026'  
SUBTRACT ='\ue027'  
DECIMAL ='\ue028'  
DIVIDE ='\ue029'  
F1 ='\ue031'# function keys  
F2 ='\ue032'  
F3 ='\ue033'  
F4 ='\ue034'  
F5 ='\ue035'  
F6 ='\ue036'  
F7 ='\ue037'  
F8 ='\ue038'  
F9 ='\ue039'  
F10 ='\ue03a'  
F11 ='\ue03b'  
F12 ='\ue03c'  
META ='\ue03d'

COMMAND    ='\ue03d'

## 关键字：**Press Key Native**

参数：

KeyCode

Wait=

描述：

用户通过发送模拟按钮操作系统按键。

KeyCode 对应 java.awt.event.keyevent 常量，可以到寻找：

[http://java.sun.com/javase/6/docs/api/constant-values.html#java.awt.event.KeyEvent.CHAR\\_UNDEFINED](http://java.sun.com/javase/6/docs/api/constant-values.html#java.awt.event.KeyEvent.CHAR_UNDEFINED)

按钮不针对特定元素，一个元素可以通过 Focus 关键字定位为第焦点

请参阅有关本文介绍的元素定位内容。

示例：

Press Key Native	517	# Exclamation mark
Focus	login_button	
Press Key Native	10	# Enter key

注意这个关键字是非常脆弱的，例如，使用键盘或鼠标在测试运行时经常引起问题。可以将窗口前再前执行 Javascript。

Execute Javascript	window.focus()	
Focus	login_button	
Press Key Native	10	and wait

## 关键字：**Press Key Native Radio Button Should (Not)**

### **Be Set to**

参数：

Group\_name,

Value

描述：

在 radiobutton 组里查证指定组名的选项已(未)被选定。

可以查看 “Select Radio Button” 获取细节。



## 关键字：**Register Keyword To Run On Failure**

参数：

Keyword

描述：

当 Selenium2Library 类库关键字执行失败时，自动执行事先设置好的其他的关键字。

Keyword\_name 就是关键字名称（从任何可用库中获得）。如果 Selenium2Library 关键字执行失败的情况下它是无法使用的该关键字需要添加参数。使用参数值“Noting”将其完全禁用此功能。

最初的关键字的使用设置 Importing，那就是默认使用的关键字是捕获页面截图。以某事时，未能屏幕画面是一种非常有用的功能，但是请注意，该关键字会降低脚本执行速度。

## 关键字：**Reload Page**

参数：

无

描述：

模拟页面重载。

## 关键字：**Select All From List**

参数：

Locator

描述：

选取所有复选列表的值。

定位元素使用 ID 和 Name 属性

请参阅有关本文介绍的元素定位内容。

## 关键字：**Select Checkbox**

参数：

Locator

描述：

选择对应 checkbox 对象。当对象已选则不触发该动作。

定位元素使用 ID 和 Name 属性

请参阅有关本文介绍的元素定位内容。

## 关键字：**Select All Frame**

参数：

Locator

描述：

设为某 frame 为当前 frame。

定位使用 Frames 元素 ID 和 Name 属性

请参阅有关本文介绍的元素定位内容。

## 关键字：**Select From List**

参数：

Locator ,

\*items

描述：

从指定 list 对象中选取\*items 项。

如果给定参数为多个选项，但是对应页面的为单选列表，那么该关键字默认选在最后一项 item。

如果目标 list 是多选列表，并且给定 items 参数为空，则所有选项均被选取。

\*items 会先使用 value 值定位选择目标，次之使用 label 值。

该关键字比使用' by index/value/label' 方法执行快。

当单选列表中最后一项不存在或者所有选项不存在时，该关键报错。

当复选列表中如果不存在选项值或者所有值均不存在时，也将报错。

## 关键字：**Select From List List By Index**

参数：

Locator ,  
\*indexes

描述：

通过 index 属性定位选择列表项。该关键字可对操作 Lists 和 Combo boxes 元素。定位元素使用 ID 和 Name 属性。请参阅有关本文介绍的元素定位内容。

## 关键字：**Select From List List By Label**

参数：

Locator ,  
\*Labels

描述：

通过 Labels 定位选择列表项。该关键字可对操作 Lists 和 Combo boxes 元素。定位元素使用 ID 和 Name 属性。请参阅有关本文介绍的元素定位内容。

## 关键字：**Select From List List By Value**

参数：

Locator ,  
\*Values

描述：

通过 Value 定位选择列表项。该关键字可对操作 Lists 和 Combo boxes 元素。定位元素使用 ID 和 Name 属性。请参阅有关本文介绍的元素定位内容。

## 关键字：**Select Radio Button**

参数：

Group\_name ,  
Value

描述：

被选 radio button 的定位方式通过以下两个参数：

1. group\_name 用来存放 radio input 的名称；
2. Value 用来存放 value 属性值或者 id 属性值；

Xpath 用来定位正确的 radio button ,大致如此 : `//input[@type=' radio' and @name=' group_name' and(@value=' value' or @id=' value' )]`

示例 :

Select Radio Button	size	XL	# Matches HTML like <code>&lt;input type="radio" name="size" value="XL"&gt;XL&lt;/input&gt;</code>
Select Radio Button	size	sizeXL	# Matches HTML like <code>&lt;input type="radio" name="size" value="XL" id="sizeXL"&gt;XL&lt;/input&gt;</code>

## 关键字 : **Select Window**

参数 :

Locator=none

描述 :

选取窗口。

如果窗口找到 ,那么所有之后使用的命令均作用于该窗口 ,知道该关键字再次使用。

如果未找到窗口 , 该关键字执行失败。

默认情况下 ,当提供 locator 的值 ,它将与 window 的 title 和 window 的 javascript 的名称进行匹配。

当打开多个 windows 窗口时 ,如果查询条件一致 ,那么默认选择第一个。

特殊一些的情况有 main(default) ,它将被使用在定位主窗口时。

当然 Selenium2Library 也提供一些特殊的窗口定位目录 :

目录	示例	描述
Title	<code>select window/title=a</code>	通过窗口 title 匹配
Name	<code>select window/name=\${b}</code>	通过窗口 java 脚本名称匹配
Url	<code>select window/url=c</code>	通过窗口当前 URL 匹配

示例 :

Click Link	popup_link	# opens new window	
Select Window	popupName		
Title Should Be	Popup Title		
Select Window			# Chooses the main window again

## 关键字 : **Set browser implicit wait**

参数 :

Seconds

描述 :

设置当前浏览器隐式等待时间。

Selenium2 中设置超时隐式等待元素 ,当元素发现后完成等待。这种方法只需要每次 Session 调用一次。

示例 :

<code>\${orig wait} =</code>	<code>Set Selenium Implicit Wait</code>	<code>10 seconds</code>
<code>Perform AJAX call that is slow</code>		
<code>Set Selenium Implicit Wait</code>	<code>\${orig wait}</code>	

## 关键字：**Set Selenium implicit wait**

参数：

Seconds

描述：

设置 seleniu2 的默认隐式等待时间，设置所有打开浏览器的隐式等待时间。

Selenium2 中设置超时隐式等待元素，当元素发现后完成等待。这种方法只需要每次 Session 调用一次。

示例：

<code>\${orig wait} =</code>	<code>Set Selenium Implicit Wait</code>	<code>10 seconds</code>
<code>Perform AJAX call that is slow</code>		
<code>Set Selenium Implicit Wait</code>	<code>\${orig wait}</code>	

## 关键字：**Set Selenium speed**

参数：

Seconds

描述：

设置每一条 selenium 命令执行的等待延迟时间。

该关键字很适合减慢测试脚本执行速度，这样做可以浏览执行过程。

Seconds 参数可以用 RF 的 time 格式。

返回值为上一个 speed 的值。

示例：

<code>Set Selenium Speed</code>	<code>.5 seconds</code>
---------------------------------	-------------------------

## 关键字：**Set Selenium Timeout**

参数：

Seconds

描述：

设置当前某步骤执行前进行的超时等待时间。

请阅读本文片首部分“超时等待”内容。

示例：

<code>\${orig timeout} =</code>	<code>Set Selenium Timeout</code>	<code>15 seconds</code>
<code>Open page that loads slowly</code>		
<code>Set Selenium Timeout</code>	<code>\${orig timeout}</code>	

## 关键字：Set Window Size

参数：

Width ,

height

描述：

设置当前窗口的高度和宽度。

示例：

<code>Set Window Size</code>	<code>\${800}</code>	<code>\${600}</code>
<code>\${width}</code>	<code>\${height}=</code>	<code>Get Window Size</code>
<code>Should Be Equal</code>	<code>\${width}</code>	<code>\${800}</code>
<code>Should Be Equal</code>	<code>\${height}</code>	<code>\${600}</code>

## 关键字：Simulate

参数：

Locator,

Event

描述：

模拟定位器定位元素事件。如果执行 OnEvent 处理程序操作元素，可以使用该关键字。

请参阅有关本文介绍的元素定位内容。

## 关键字：Switch Browser

参数：

Index\_or\_alias

描述：

通过 index 或 alias 在所有运行状态下的 browser 进行切换选择。

Index 可以通过 open browser 返回获取，alias 由用户提供。

示例：

Open Browser	<a href="http://google.com">http://google.com</a>	ff	
Location Should Be	<a href="http://google.com">http://google.com</a>		
Open Browser	<a href="http://yahoo.com">http://yahoo.com</a>	ie	2nd conn
Location Should Be	<a href="http://yahoo.com">http://yahoo.com</a>		
Switch Browser	1	# index	
Page Should Contain	I'm feeling lucky		
Switch Browser	2nd conn	# alias	
Page Should Contain	More Yahoo!		
Close All Browsers			

以上例子，希望没有其他浏览器打开时它第一使用 Index“1”切换时浏览器。如果你不确定，你可以将 Index 赋值给一个变量，如下：

#{id} =	Open Browser	<a href="http://google.com">http://google.com</a>	*firefox
# Do something ...			
Switch Browser	#{id}		

## 关键字：Table cell should contain

参数：

Table\_locator,

Row,

Column,

Expected,

LogLevel =INFO

描述：

校验某单元格中是否包含期望值信息。

行数和列数从 1 开始计算。

当指定单元格中包含的内容和给定内容一致时，该关键字执行通过。

当你希望测试某单元格内容准确匹配，或者比如通过某些文本内容开头，那么你可以使用 get table cell 关键字并结合一些其他内嵌关键字使用，例如 should be equal/should start with 等。

对于 Table 的定位，请参阅有关本文介绍的元素定位内容。

可以查看“Page Should Contain”关于 loglevel 参数的解释。

## 关键字：Table Column should contain

参数：

Table\_locator,  
Col,  
Expected,  
Loglevel=INFO

描述：

校验指定列是否包含预期值。

最左边列计数 1 起始，-1 指定从尾行最后一排起始；

如果表格的单元格存在跨列组合的情况，这些组合元素将按照单列进行计算。比如有这样一个表格，当某一行是由列 A 和列 B 合并而成的，它的物理 colspan 为 2，但是逻辑第三列包含“C”。

示例：

Table Column Should Contain	tableId	3	C
Table Column Should Contain	tableId	2	C

对于 Table 的定位，请参阅有关本文介绍的元素定位内容。

可以查看“[Page Should Contain](#)”关于 loglevel 参数的解释。

## 关键字：Table Footer should contain

参数：

Table\_locator,  
Expected,  
Loglevel=INFO

描述：

校验表格的 footer 包含预期值。

Footer 的意思可以解释为任何 <td> 元素，该元素是 <tfoot> 的子元素。

如果希望了解 tables 是如何识别的，请在 introduction 中查询。

对于 Table 的定位，请参阅有关本文介绍的元素定位内容。

可以查看“[Page Should Contain](#)”关于 loglevel 参数的解释。

## 关键字：Table Header should contain



参数：

Table\_locator,  
Expected,  
Loglevel=INFO

描述：

校验表格的 header，任何<th>...</th>元素包含预期值。

对于 Table 的定位，请参阅有关本文介绍的元素定位内容。

可以查看“[Page Should Contain](#)”关于 loglevel 参数的解释。

## 关键字：[Table Row should contain](#)

参数：

Table\_locator,  
Row  
Expected,  
Loglevel=INFO

描述：

校验指定行是否包含预期值。

从最上面一列计数 1 起始，-1 指定从尾部最后一列起始；

如果表格的结构包含 thead,tbody 和 tfoot 庞大，最好从 tbody 部分开始搜索；

请阅读 [Table Header Should Contain](#) 或 [Table Footer Should Contain](#) 来验证表格的 Header 或 Footer 内容。

如果表格的单元格存在跨列组合的情况，这些组合元素将按照单列进行计算。比如有这样一个表格，当某一行是由列 A 和列 B 合并而成的，它的物理 colspan 为 2，但是逻辑第三列包含“C”。

## 关键字：[Table should Contain](#)

参数：

Table\_locator,  
Expected,  
Loglevel=INFO

描述：

校验在表格某个位置中可以找到期望值。

对于 Table 的定位，请参阅有关本文介绍的元素定位内容。

可以查看“[Page Should Element](#)”关于 loglevel 参数的解释。

## 关键字：Textarea Should Contain

参数：

Locator,  
Expected,  
Message=

描述：

验证指定文本输入域中存在期望值。  
Message 参数可以覆盖默认报错信息。  
该关键字使用 id 和 name 属性定位文本域。

## 关键字：Textarea value should be

参数：

Locator,  
Expected,  
Message=

描述：

验证指定文本输入域的值与期望值准确匹配。  
Message 参数可以覆盖默认报错信息。  
该关键字使用 id 和 name 属性定位文本域。

## 关键字：Textfield should contain

参数：

Locator,  
Expected,  
Message=

描述：

校验文本输入框中存在期望值。  
Message 参数可以覆盖默认报错信息。  
该关键字使用 id 和 name 属性定位文本域。

## 关键字：Textfield Value Should be

参数：

Locator,  
Expected,  
Message=

描述：

验证指定文本输入框的值与期望值准确匹配。

Message 参数可以覆盖默认报错信息。

该关键字使用 id 和 name 属性定位文本域。

## 关键字：Title should be

参数：

Title

描述：

校验当前页面 title 值与期望值一致。

## 关键字：Unselect checkbox

参数：

Locator

描述：

移除 checkbox 选取状态。

当 checkbox 未被选取时，此关键字无效。

该关键字使用 id 和 name 属性定位 checkbox

请参阅有关本文介绍的元素定位内容。

## 关键字：Unselect Frame

参数：

Locator

描述：

将最前端 frame 设置为当前操作 frame。

## 关键字：**Unselect From List**

参数：

Locator,  
\*items

描述：

移除指定 select 下的给定值的被选状态。

特别，当 list 参数为空时，将默认移除所有被选项。

\*items 通过 value 和 label 移除选项。

当使用 'by index/value/label' 方法将更快执行。

该关键字同样适用于 lists 和 combo boxes。

使用 id 和 name 定位 select lists,请参阅有关本文介绍的元素定位内容。

## 关键字：**Unselect From List By Index**

参数：

Locator,  
\*indexs

描述：

用途同 “unselect from list” ,通过 index 值确定 list 项。

该关键字同样适用于 lists 和 combo boxes。

使用 id 和 name 定位 select lists,请参阅有关本文介绍的元素定位内容。

## 关键字：**Unselect From List By Label**

参数：

Locator,  
\*Labels

描述：

用途同 “unselect from list” ,通过 label 值确定 list 项。

该关键字同样适用于 lists 和 combo boxes。

使用 id 和 name 定位 select lists,请参阅有关本文介绍的元素定位内容。

## 关键字：**Unselect From List By Value**

参数：

Locator,  
\*Values

描述：

用途同 “unselect from list” ,通过 Value 值确定 list 项。

该关键字同样适用于 lists 和 combo boxes。

使用 id 和 name 定位 select lists,请参阅有关本文介绍的元素定位内容。

## 关键字：**Wait for condition**

参数：

Condition,  
Timeout=none,  
Error=none

描述：

等待给出条件为真或者超时。

代码可以包含多行，但是必须包含返回语句。

条件可以是各种 javascript 表达式但是必须包含返回语句。

Error 可以用来覆盖默认报错信息。

可查看[介绍](#)来了解更多关于 timeout 和其默认值的内容。

类似关键字还有 **Wait until page contains** , **Wait until page contains element** , **Wait until element is visible** 和 BuiltIn 库的 **Wait Until Keyword Succeeds**。

## 关键字：**Wait Until Element Is Visible**

参数：

Locator,  
Timeout=none,  
Error=none

描述：

等待页面中指定元素可见,等待超时执行失败。

Error 可以用来覆盖默认报错信息。

类似关键字还有 **Wait until page contains** , **Wait until page contains element** , **Wait until element is visible** 和 BuiltIn 库的 **Wait Until Keyword Succeeds**。

## 关键字：**Wait Until Page Contains**

参数：

Locator,

Timeout=none,

Error=none

描述：

等待当前页面中指定 text 文本内容出现。

等待超时执行失败。

Error 可以用来覆盖默认报错信息。

类似关键字还有 **Wait until page contains** , **Wait until page contains element** , **Wait until element is visible** 和 BuiltIn 库的 **Wait Until Keyword Succeeds**。

## 关键字：**Wait Until Page Contains Element**

参数：

Locator,

Timeout=none,

Error=none

描述：

等待当前页面中某指定元素出现。

等待超时执行失败。

Error 可以用来覆盖默认报错信息。

类似关键字还有 **Wait until page contains** , **Wait until page contains element** , **Wait until element is visible** 和 BuiltIn 库的 **Wait Until Keyword Succeeds**。

## 关键字：**Xpath Should Match X Times**

参数：

Xpath,

Expected\_xpath\_count,

Message=,

Loglevel=INFO

描述：

验证页面包含指定的 Xpath 元素与 Expected\_xpath\_count 给定的数量是否一致。

Message 可以设置输出信息。

使用 xpath 定位语法不应使用 " **xpath=** " 作为前缀。

正确写法:

```
Xpath Should Match X Times | //div[@id='sales-pop'] | 1
```

错误写法:

```
Xpath Should Match X Times | xpath=//div[@id='sales-pop'] | 1
```

可以查阅 "**Page Should Contains Element**" 关键字关于 loglevel 参数的内容。

# 附录：

## 更新内容

**Selenium2Library1.8** 是将 Selenium2 API 版本更新到 2.8，根据 Selenium API 更新内容，Library 对关键字及元素定位方式做出了相应的更新.

更新内容详见以下内容：

### Locators 定位方式

---

定位方式中没有通配符支持

新增 prefixes/strategies 的定位方式: link=, css=, tag=

### 新增关键字

Add Location Strategy

Dismiss Alert

Get Horizontal Position

Get Element Size

Get Webelements

Get Selenium Speed

Get Selenium Timeout

Log Title

Log Location



Capture Page Screenshot

## 变更关键字

Choose File - 添加远端文件支持

Click Button - 移出参数: dont\_wait

Click Element - 移出参数: dont\_wait, coordinates

Click Image - 移出参数: dont\_wait

Click Link - 移出参数: dont\_wait

Double Click Element - 移出参数: dont\_wai, coordinates

Go Back - 移出参数: dont\_wait

Open Context Menu - 移出参数: offse

Press Key - 移出参数: dont\_waitargument

Select All From List - 移出参数: wait

Select Radio Button - 移出参数: wait

Submit Form - 移出参数: dont\_wait

## 移出关键字

Flex-related keywords

Add Location Strategy

Call Selenium API

Capture Screenshot 替换为 Capture Page Screenshot

Drag And Drop

Press Key Native

Start Selenium Server

Stop Selenium Server

Wait Until Page Loaded