



ThoughtWorks

---

# Robot Framework

---

Jingjing Mao

Email: [jjmao@thoughtworks.com](mailto:jjmao@thoughtworks.com)

Skype: [maomao112724@hotmail.com](https://www.hotmail.com/maomao112724)

# 为什么学习使用Robot Framework

---

1				
2				
3				
4				
5				

# 为什么学习使用Robot Framework

Prj Test Report

Generated  
20150524 01:38:03 GMT -07:00  
18 seconds ago

LOG

Summary Information

Status: 1 critical test failed

Start Time: 20150524 01:37:49.021

End Time: 20150524 01:38:03.378

Elapsed Time: 00:00:14.357

Log File: [log.html](#)

Test Statistics

Total Statistics	Total	Pass	Fail	Elapsed	Pass / Fail
Critical Tests	2	1	1	00:00:14	<div><div></div></div>
All Tests	2	1	1	00:00:14	<div><div></div></div>

Statistics by Tag	Total	Pass	Fail	Elapsed	Pass / Fail
important	1	1	0	00:00:08	<div><div></div></div>

Statistics by Suite	Total	Pass	Fail	Elapsed	Pass / Fail
Prj	2	1	1	00:00:14	<div><div></div></div>
Prj.TestSuite	2	1	1	00:00:14	<div><div></div></div>

Test Details

Totals

Tags

Suites

Search

Type:  
☒ Critical Tests  
☐ All Tests

Status: 2 total, 1 passed, 1 failed

Total Time: 00:00:14.027

Name	Documentation	Tags	Crit.	Status	Message	Elapsed	Start / End
Prj.TestSuite.Search			yes	FAIL	'robotframework_百度搜索' does not contain 'cucumber'	00:00:05.754	20150524 01:37:49.351 20150524 01:37:55.105
Prj.TestSuite.Search_Use_Keywords		important	yes	PASS		00:00:08.273	20150524 01:37:55.105 20150524 01:38:03.378

# 为什么学习使用Robot Framework

C:\Users\Administrator\AppData\Local\Temp\RIDEowg9td.d\log.html Prj Test Log

Status: 2 critical test, 1 passed, 1 failed  
2 test total, 1 passed, 1 failed

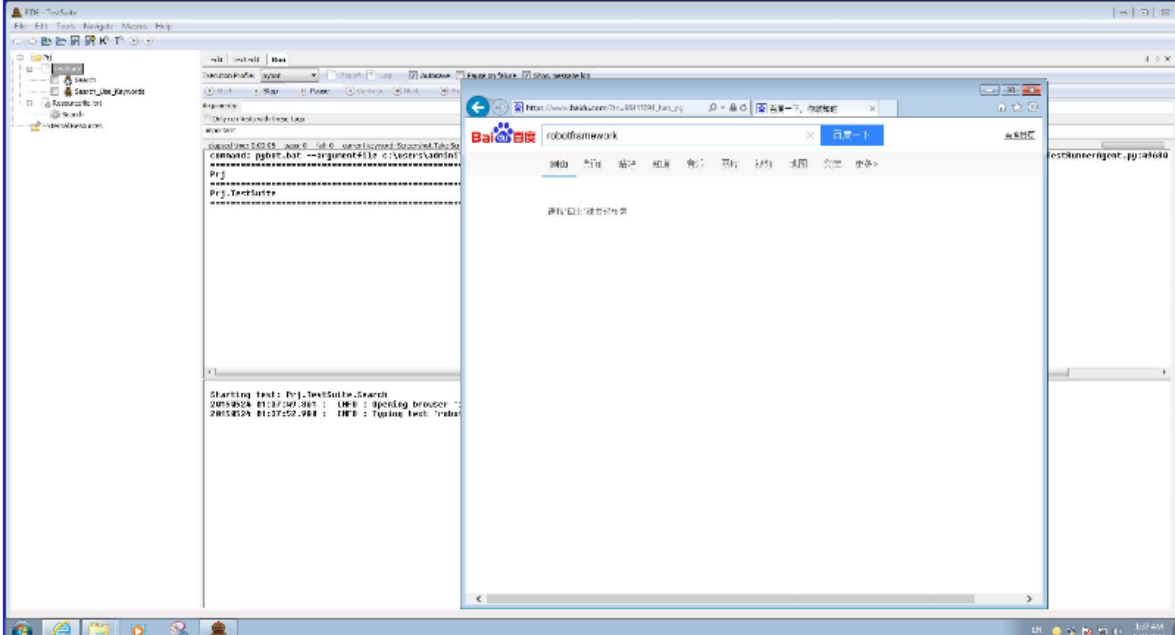
REPORT

**TEST CASE: Search** 00:00:05.754  
Full Name: Prj.TestSuite.Search  
Start / End / Elapsed: 20150524 01:37:49.351 / 20150524 01:37:55.105 / 00:00:05.754  
Status: **FAIL** (critical)  
Message: 'robotframework\_百度搜索' does not contain 'cucumber'

**SETUP: Selenium2Library.Open Browser** <http://www.baidu.com>, ie 00:00:03.619  
Documentation: Opens a new browser instance to given URL.  
Start / End / Elapsed: 20150524 01:37:49.361 / 20150524 01:37:52.980 / 00:00:03.619  
01:37:49.361 INFO Opening browser 'ie' to base url 'http://www.baidu.com'

**KEYWORD: Selenium2Library.Input Text** id=kw, robotframework 00:00:00.711  
Documentation: Types the given 'text' into text field identified by 'locator'.  
Start / End / Elapsed: 20150524 01:37:52.980 / 20150524 01:37:53.691 / 00:00:00.711  
01:37:52.980 INFO Typing text 'robotframework' into text field 'id=kw'

**KEYWORD: Screenshot.Take Screenshot** 00:00:00.140  
Documentation: Takes a screenshot in JPEG format and embeds it into the log file.  
Start / End / Elapsed: 20150524 01:37:53.691 / 20150524 01:37:53.831 / 00:00:00.140  
01:37:53.831 INFO



**KEYWORD: Selenium2Library.Click Button** id=su 00:00:00.442  
**KEYWORD: Selenium2Library.Go To** [https://www.baidu.com/s?ie=utf-8&f=8&rsrv\\_bp=1&rsrv\\_idx=1&tn=baidu&wd=robotframework&rsrv\\_pq=ac148f4c000326ef&rsrv\\_t=db1aQU6%2FFyOpB8L0O6kCoCoFIVH%2F6U3G9kUpNFURlyivLWm0pB6vuMBvEQ&rsrv\\_enter=1&inputT=3606&rsrv\\_sug1=8&rsrv\\_n=2&rsrv\\_sug3=9&bs=cucumber](https://www.baidu.com/s?ie=utf-8&f=8&rsrv_bp=1&rsrv_idx=1&tn=baidu&wd=robotframework&rsrv_pq=ac148f4c000326ef&rsrv_t=db1aQU6%2FFyOpB8L0O6kCoCoFIVH%2F6U3G9kUpNFURlyivLWm0pB6vuMBvEQ&rsrv_enter=1&inputT=3606&rsrv_sug1=8&rsrv_n=2&rsrv_sug3=9&bs=cucumber) 00:00:00.822  
**KEYWORD: \${title} = Selenium2Library.Get Title** 00:00:00.010  
**KEYWORD: BuiltIn.Should Contain** \${title}, cucumber 00:00:00.000

# 议程

---

- 概念
- 创建测试数据
- 执行测试用例
- 参考链接
- 总结

# 概念

---

- **Robot Framework:**

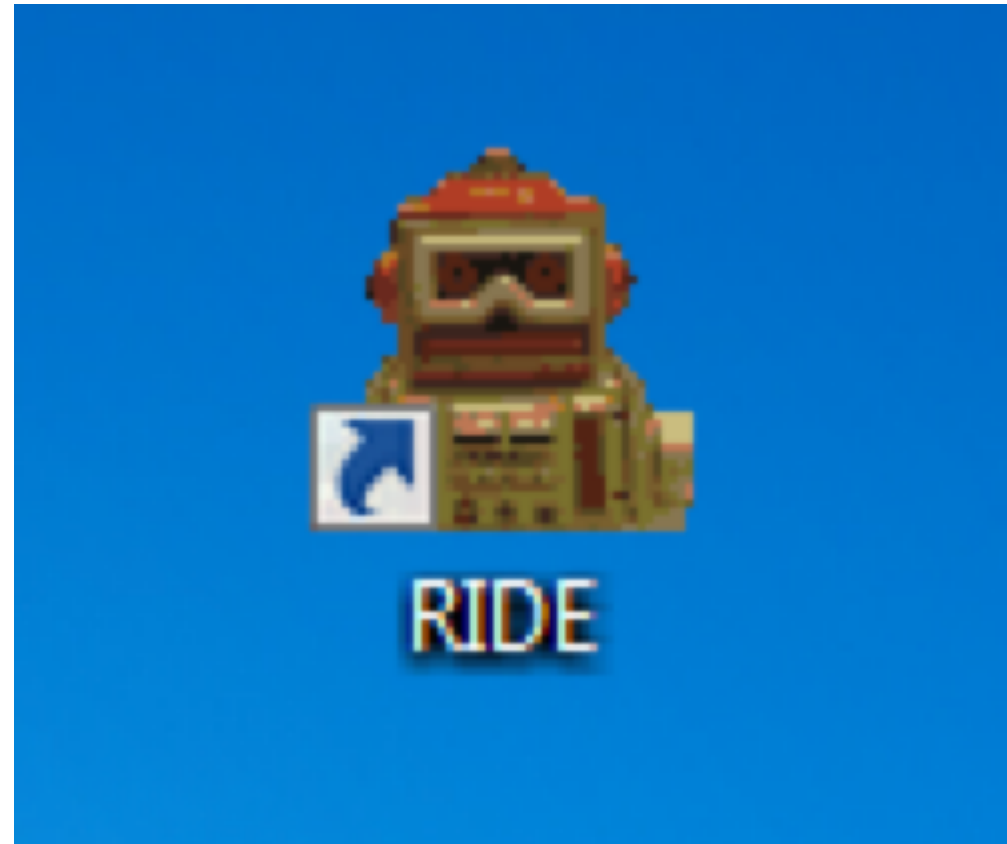
一种基于 Python 的，可扩展的关键字驱动的自动化测试框架。

# 概念

---

- **RIDE:**

Robot Framework提供的一个图形用户界面，它能够很好地编辑和管理用例、保存关键字的资源文件等。



# 创建测试数据

## 测试集

The screenshot displays the RIDE IDE interface for a project named 'Prj'. The left sidebar shows the project structure with a red box highlighting the 'TestSuite' folder, which contains 'Search' and 'Search\_Use\_Keywords' files. The main area shows the 'Prj' source code at 'C:\prj'. The 'Settings <<' button is visible. The 'Documentation' section is empty. The 'Suite Setup', 'Suite Teardown', 'Test Setup', and 'Test Teardown' sections each have a text input field and 'Edit' and 'Clear' buttons. The 'Force Tags' section has a '<Add New>' button and 'Edit' and 'Clear' buttons. The 'Import' table has columns for 'Name / Path', 'Arguments', and 'Comment'. The 'Variable' table has columns for 'Value' and 'Comment'. The right sidebar contains buttons for 'Add Import', 'Library', 'Resource', 'Variables', 'Import Failed Help', 'Add Scalar', and 'Add List'.

RIDE - Prj

File Edit Tools Navigate Macros Help

TestSuite

Search

Search\_Use\_Keywords

Resourcefile.txt

External Resources

Prj

Source C:\prj

Settings <<

Documentation

Suite Setup

Suite Teardown

Test Setup

Test Teardown

Force Tags

<Add New>

Import

Name / Path	Arguments	Comment
-------------	-----------	---------

Variable

Value	Comment
-------	---------

Edit Clear

Edit Clear

Edit Clear

Edit Clear

Edit Clear

Add Import

Library

Resource

Variables

Import Failed Help

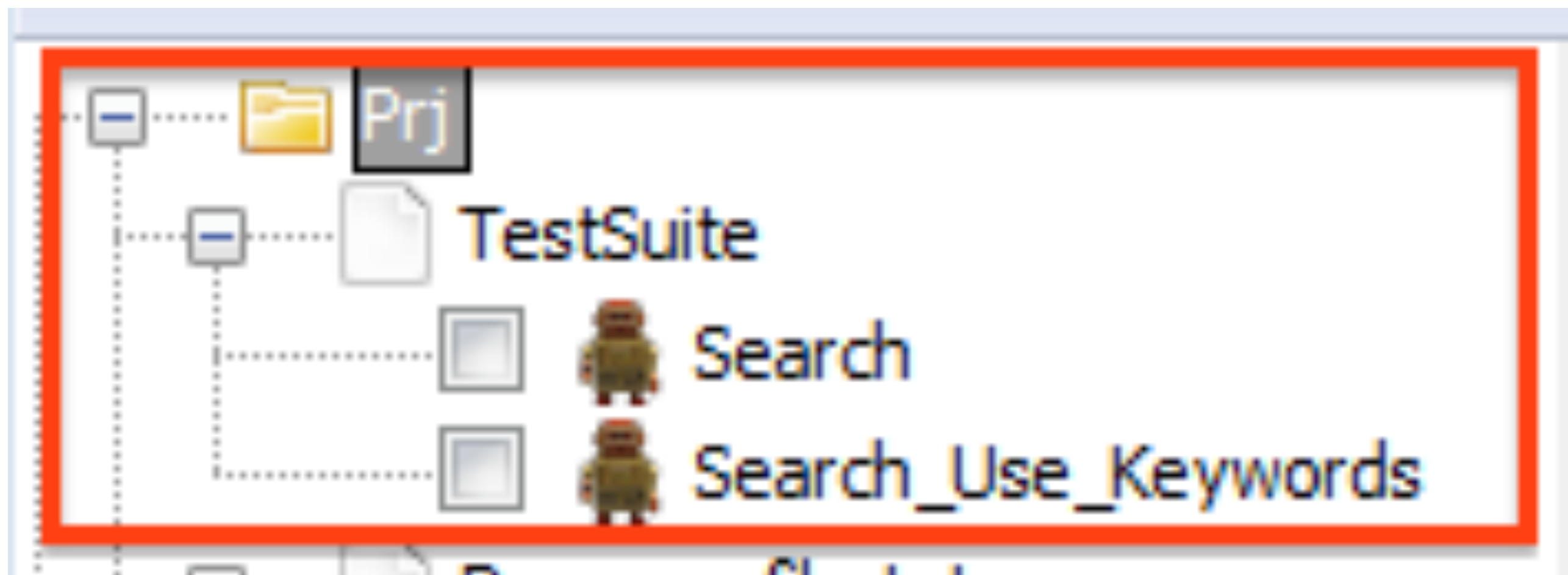
Add Scalar

Add List



# 创建测试数据

## 测试集



# 创建测试数据

## 测试用例

RIDE - TestSuite

File Edit Tools Navigate Macros Help

Prj TestSuite Search Search\_Use\_Keywords Resourcefile.txt External Resources

Edit Text Edit Run

Search

Settings >>

1	input text	id=kw	robotframework	
2	take screenshot			
3	click button	id=su		
4	goto	https://www.baidu.com/s?ie=utf-8		
5	\${title}	get title		
6	should contain	\${title}	cucumber	
7	take screenshot			
8				
9				
10				
11				
12				

# 创建测试数据

## 测试用例

1	input text	id=kw	robotframework				
2	take screenshot						
3	click button	id=su					
4	goto	https://www.baidu.com/s?ie=utf-8					
5	\${title}	get title					
6	should contain	\${title}	cucumber				
7	take screenshot						
8							
9							
10							

# 创建测试数据

## 标准库

The screenshot shows the RIDE - TestSuite application interface. The main window displays the 'TestSuite' source file at 'C:\prj\TestSuite.txt'. A 'Search Keywords' dialog box is open, showing a search term 'Search' and a source dropdown set to '<all keywords>'. The dialog contains a table of keywords and their descriptions.

Name	Source	Description
Set Global Variable	BuiltIn	Makes a variable available globally in all tests and suites.
Set Library Search Order	BuiltIn	Sets the resolution order to use when a name matches multiple keywords.
Set Log Level	BuiltIn	Sets the log threshold to the specified level and returns the old level.
Set Suite Documentation	BuiltIn	Sets documentation for the current test suite.
Set Suite Metadata	BuiltIn	Sets metadata for the current test suite.
Set Suite Variable	BuiltIn	Makes a variable available everywhere within the scope of the current suite.
Set Tags	BuiltIn	Adds given `tags` for the current test or all tests in a suite.
Set Test Documentation	BuiltIn	Sets documentation for the current test case.
Set Test Message	BuiltIn	Sets message for the current test case.
Set Test Variable	BuiltIn	Makes a variable available everywhere within the scope of the current test.
Set Variable	BuiltIn	Returns the given values which can then be assigned to a variables.
Set Variable If	BuiltIn	Sets variable based on the given condition.
Should Be Empty	BuiltIn	Verifies that the given item is empty.

Below the table, the details for the selected keyword 'Set Variable' are shown:

**Name:** Set Variable  
**Source:** BuiltIn <test library>  
**Arguments:** [ \*values ]

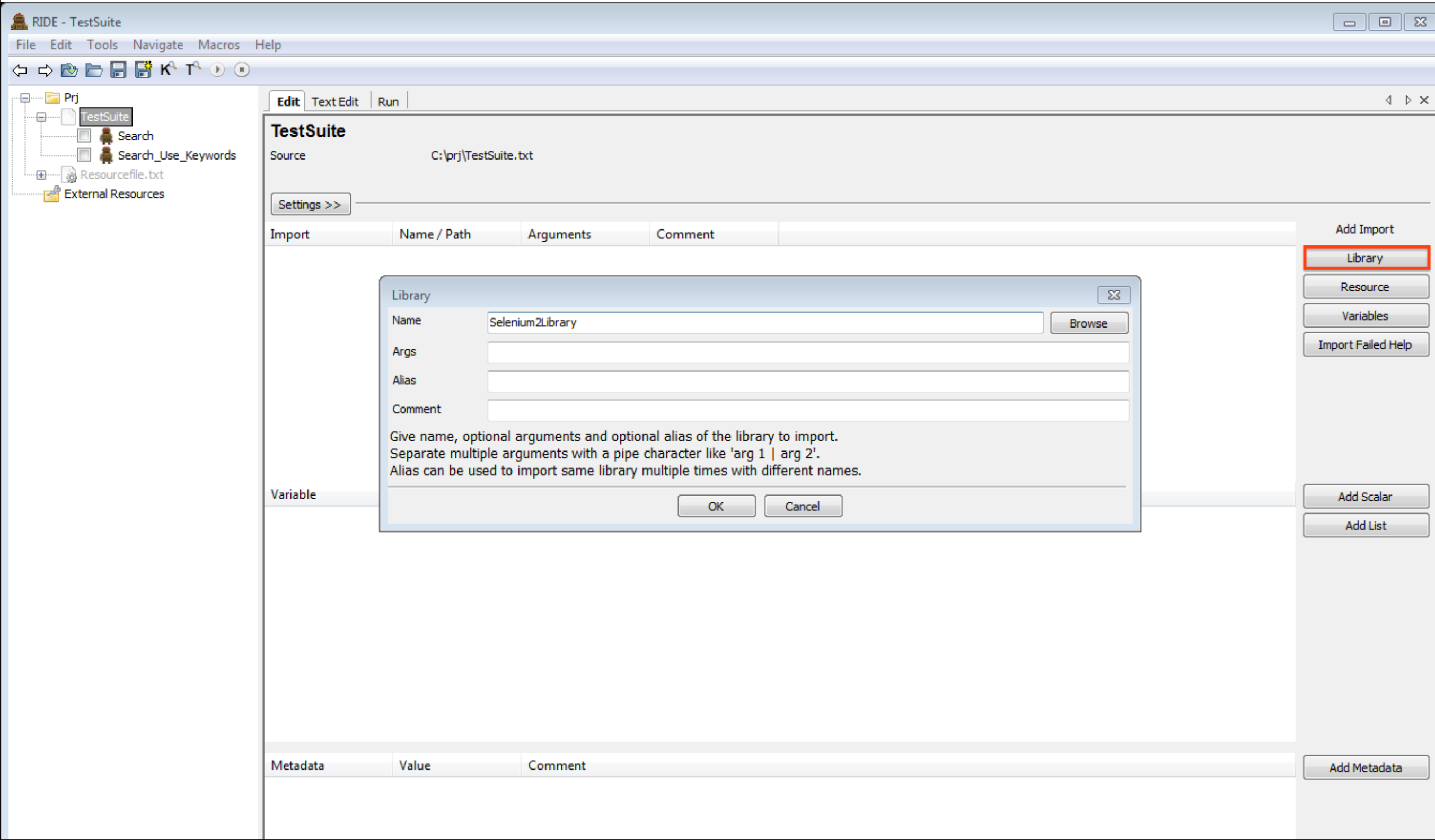
Returns the given values which can then be assigned to a variables.

This keyword is mainly used for setting scalar variables. Additionally it can be used for converting a scalar variable containing a list to a list variable or to multiple scalar variables. It is recommended to use `Create List` when creating new lists.

On the right side of the application, there are buttons for 'Add Import' (Library, Resource, Variables, Import Failed Help), 'Add Scalar', 'Add List', and 'Add Metadata'.

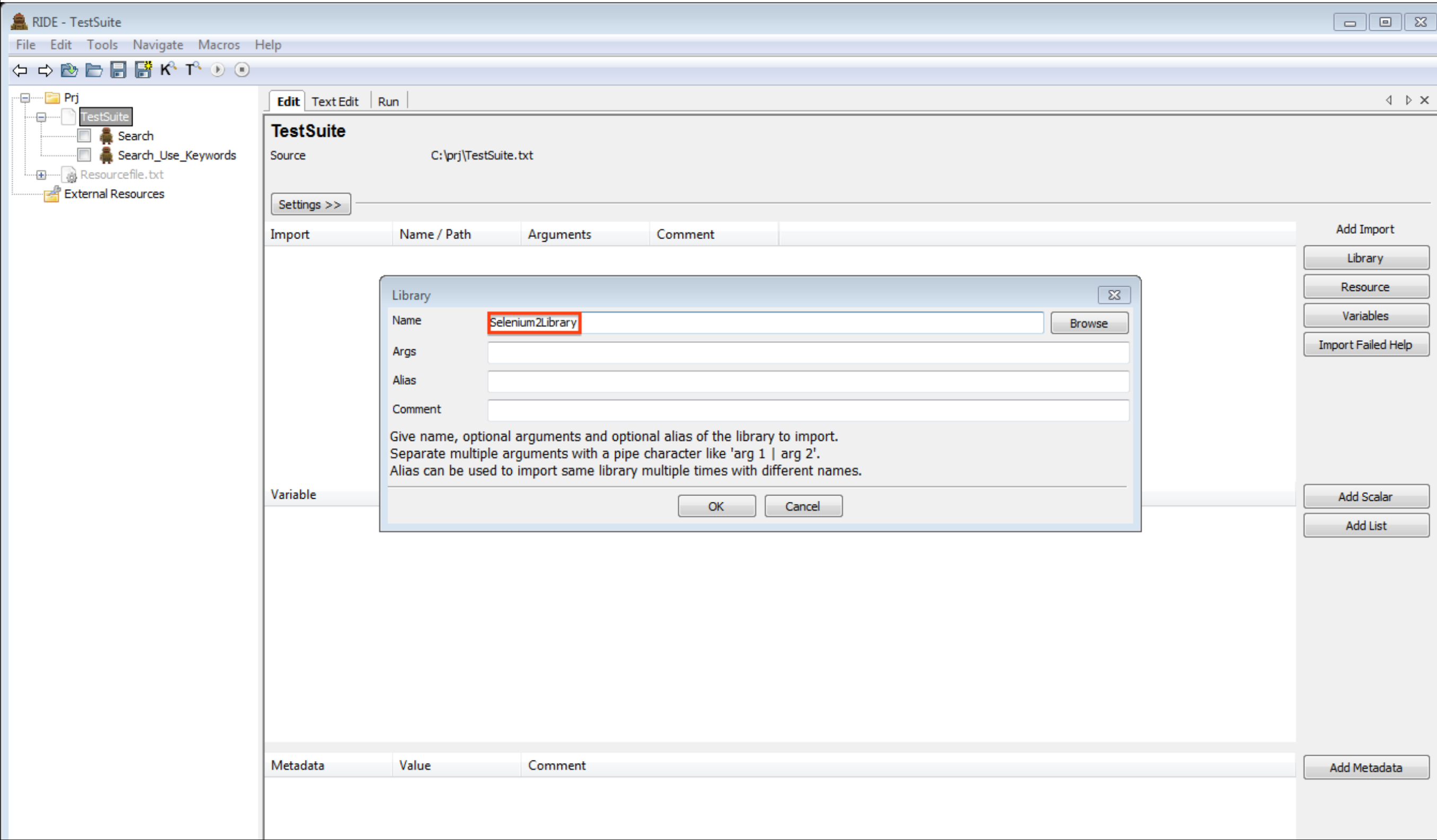
# 创建测试数据

## 外部扩展库



# 创建测试数据

## 外部扩展库



# 创建测试数据

## 外部扩展库

The screenshot displays the RIDE - TestSuite application. The main window shows the 'TestSuite' editor with the source file 'C:\prj\TestSuite.txt'. The 'Library' tab is selected, showing 'Selenium2Library' imported. A 'Search Keywords' dialog is open, displaying a list of keywords from the Selenium2Library. The 'Open Browser' keyword is highlighted. The dialog also shows the source of the keyword and its arguments.

**TestSuite**  
Source: C:\prj\TestSuite.txt

Settings >>

Import	Name / Path	Arguments	Comment
Library	Selenium2Library		

**Search Keywords**

Search term:  Search ☒ Search documentation

Source: <all keywords>

Name	Source	Description
Mouse Up	Selenium2Library	Simulates releasing the left mouse button on the element specified by 'locator'.
No Operation	BuiltIn	Does absolutely nothing.
Open Browser	Selenium2Library	Opens a new browser instance to given URL.
Open Context Menu	Selenium2Library	Opens context menu on element identified by 'locator'.
Page Should Contain	Selenium2Library	Verifies that current page contains 'text'.
Page Should Contain Button	Selenium2Library	Verifies button identified by 'locator' is found from current page.
Page Should Contain Checkbox	Selenium2Library	Verifies checkbox identified by 'locator' is found from current page.
Page Should Contain Element	Selenium2Library	Verifies element identified by 'locator' is found on the current page.
Page Should Contain Image	Selenium2Library	Verifies image identified by 'locator' is found from current page.
Page Should Contain Link	Selenium2Library	Verifies link identified by 'locator' is found from current page.
Page Should Contain List	Selenium2Library	Verifies select list identified by 'locator' is found from current page.
Page Should Contain Radio Button	Selenium2Library	Verifies radio button identified by 'locator' is found from current page.
Page Should Contain Textfield	Selenium2Library	Verifies text field identified by 'locator' is found from current page.

**Open Browser**  
Source: Selenium2Library <test library>  
Arguments: [ url | browser=firefox | alias=None | remote\_url=False | desired\_capabilities=None | ff\_profile\_dir=None ]

Opens a new browser instance to given URL.

Returns the index of this browser instance which can be used later to switch back to it. Index starts from 1 and is reset back to it when 'Close All Browsers' keyword is used. See 'Switch Browser' for example.

Buttons on the right: Add Import, Library, Resource, Variables, Import Failed Help, Add Scalar, Add List, Add Metadata.

# 创建测试数据

## 外部扩展库

*Name:* Open Browser

*Source:* Selenium2Library <test library>

*Arguments:* [ url | browser=firefox | alias=None | remote\_url=False | desired\_capabilities=None | ff\_profile\_dir=None ]

Opens a new browser instance to given URL.

Returns the index of this browser instance which can be used later to switch back to it. Index starts from 1 and is reset back to it when `Close All Browsers` keyword is used. See `Switch Browser` for example.

Optional alias is an alias for the browser instance and it can be used for switching between browsers (just as index can be used). See `Switch Browser` for more details.

Possible values for `browser` are as follows:

firefox	FireFox
ff	FireFox



# 创建测试数据

## 资源文件及用户关键字

RIDE - Resourcefile

File Edit Tools Navigate Macros Help

Prj

- TestSuite
- Search
- Search\_Use\_Keywords
- Resourcefile.txt
- Search
- External Resources

**Search**

Settings << Documentation

Arguments: `${search_content} | ${goto_url}`

Teardown:

Timeout:

Return Value:

Step	Step Name	ID	Data
1	input text	id=kw	<code>\${search_content}</code>
2	click button	id=su	
3	goto		<code>\${goto_url}</code>
4	<code>\${title}</code>	get title	
5	should contain		<code>\${title}</code> <code>\${search_content}</code>
6			
7			
8			
9			
10			

# 创建测试数据

## 资源文件及用户关键字

Arguments		<u><code>\${search_content}   \${goto_url}</code></u>			
Teardown					
Timeout					
Return Value					
1	input text	id=kw	<u><code>\${search_content}</code></u>		
2	click button	id=su			
3	goto	<u><code>\${goto_url}</code></u>			
4	<code>\${title}</code>	get title			
5	should contain	<code>\${title}</code>	<u><code>\${search_content}</code></u>		
6					
7					
8					
9					

# 创建测试数据

## 资源文件及用户关键字

The screenshot shows the RIDE - TestSuite application. The left sidebar displays a project tree with the following structure:

- Prj
  - TestSuite
    - Search
    - Search\_Use\_Keywords
  - Resourcefile.txt
  - Search
  - External Resources

The main editor area is titled "Search\_Use\_Keywords" and contains a table with 6 rows and 4 columns. The first row is highlighted with a red border and contains the following data:

	Search	cucumber	https://www.baidu.com/s?ie=utf-8&f=8&rsv_bp=0
1	Search	cucumber	https://www.baidu.com/s?ie=utf-8&f=8&rsv_bp=0
2			
3			
4			
5			
6			

# 创建测试数据

## 资源文件及用户关键字

1	Search	cucumber	<a href="https://www.baidu.com/s?ie=utf-8&amp;f=8&amp;rsv_bp=C">https://www.baidu.com/s?ie=utf-8&amp;f=8&amp;rsv_bp=C</a>
---	--------	----------	---

# 执行测试用例

## 测试结果

The screenshot displays the RIDE - TestSuite application interface. The left sidebar shows a project tree with 'Prj' containing 'TestSuite', 'Search', 'Search\_Use\_Keywords', 'Resourcefile.txt', and 'External Resources'. The main window is divided into a top toolbar and a large output area. The toolbar includes buttons for 'Start', 'Stop', 'Pause', 'Continue', 'Next', and 'Step over', along with checkboxes for 'Autosave', 'Pause on failure', and 'Show message log'. The output area shows the following test results:

```
elapsed time: 0:00:23 pass: 1 fail: 1
Prj
=====
Prj.TestSuite
=====
Search | FAIL |
'robotframework_百度搜索' does not contain 'cucumber'
-----
Search_Use_Keywords | PASS |
-----
Prj.TestSuite | FAIL |
2 critical tests, 1 passed, 1 failed
2 tests total, 1 passed, 1 failed
-----
Prj | FAIL |
2 critical tests, 1 passed, 1 failed
2 tests total, 1 passed, 1 failed
-----
Output: c:\users\admini~1\appdata\local\temp\RIDEkhdakt.d\output.xml
Log: c:\users\admini~1\appdata\local\temp\RIDEkhdakt.d\log.html
Report: c:\users\admini~1\appdata\local\temp\RIDEkhdakt.d\report.html

test finished 20150524 02:51:46

Starting test: Prj.TestSuite.Search
20150524 02:51:25.750 : INFO : Opening browser 'ie' to base url 'http://www.baidu.com'
20150524 02:51:32.351 : INFO : Typing text 'robotframework' into text field 'id=kw'
20150524 02:51:33.323 : INFO : <a href="screenshot_1.jpg"></a>
20150524 02:51:33.323 : INFO : Clicking button 'id=su'.
20150524 02:51:34.092 : INFO : Opening url 'https://www.baidu.com/s?ie=utf-8&f=8&rsv_bp=1&rsv_idx=1&tn=baidu&wd=robotframewor'
20150524 02:51:35.827 : INFO : ${title} = robotframework_百度搜索
20150524 02:51:35.947 : FAIL : 'robotframework_百度搜索' does not contain 'cucumber'
Ending test: Prj.TestSuite.Search

Starting test: Prj.TestSuite.Search_Use_Keywords
20150524 02:51:35.967 : INFO : Opening browser 'ie' to base url 'http://www.baidu.com'
```

# 执行测试用例

## 测试结果

elapsed time: 0:00:23 pass: 1 fail: 1

Prj

Prj.TestSuite

Search

| FAIL |

'robotframework\_百度搜索' does not contain 'cucumber'

Search\_Use\_Keywords

| PASS |

Prj.TestSuite

| FAIL |

2 critical tests, 1 passed, 1 failed

2 tests total, 1 passed, 1 failed

Prj

| FAIL |

2 critical tests, 1 passed, 1 failed

2 tests total, 1 passed, 1 failed

Output: c:\users\admini~1\appdata\local\temp\RIDEkhdakt.d\output.xml

Log: c:\users\admini~1\appdata\local\temp\RIDEkhdakt.d\log.html

Report: c:\users\admini~1\appdata\local\temp\RIDEkhdakt.d\report.html

test finished 20150524 02:51:46

Starting test: Prj.TestSuite.Search

20150524 02:51:25.750 : INFO : Opening browser 'ie' to base url 'http://www.baidu.com'

20150524 02:51:32.351 : INFO : Typing text 'robotframework' into text field 'id=kw'

20150524 02:51:33.323 : INFO : <a href="screenshot\_1.jpg"></a>

20150524 02:51:33.323 : INFO : Clicking button 'id=su'.

20150524 02:51:34.092 : INFO : Opening url 'https://www.baidu.com/s?ie=utf-8&f=8&rsu\_bp=1&rsu\_idx=1&tn=baidu&wd=robotframewor

20150524 02:51:35.827 : INFO : \${title} = robotframework\_百度搜索

20150524 02:51:35.947 : FAIL : 'robotframework\_百度搜索' does not contain 'cucumber'

Ending test: Prj.TestSuite.Search

Starting test: Prj.TestSuite.Search\_Use\_Keywords

20150524 02:51:35.967 : INFO : Opening browser 'ie' to base url 'http://www.baidu.com'

20150524 02:51:40.417 : INFO : Typing text 'cucumber' into text field 'id=kw'

20150524 02:51:41.158 : INFO : Clicking button 'id=su'.

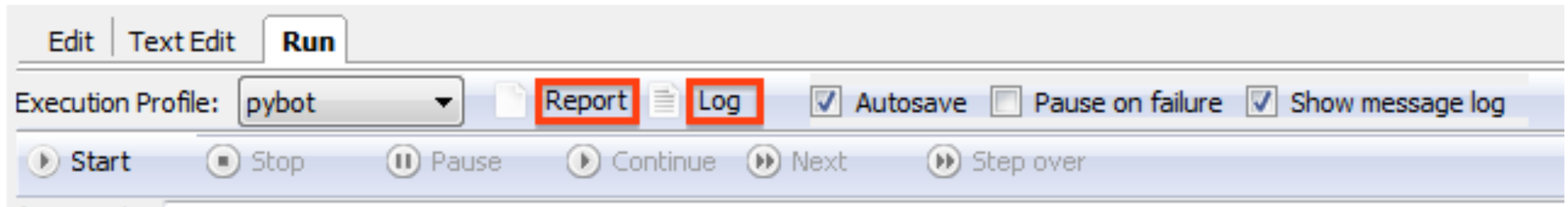
20150524 02:51:42.403 : INFO : Opening url 'https://www.baidu.com/s?ie=utf-8&f=8&rsu\_bp=0&rsu\_idx=1&tn=baidu&wd=cucumber&rsu\_

20150524 02:51:43.805 : INFO : \${title} = cucumber\_百度搜索

Ending test: Prj.TestSuite.Search\_Use\_Keywords

# 执行测试用例

## 测试报告及日志



# 执行测试用例

## 测试报告及日志

file:///C:/Users/Administrator/AppData/Local/Temp/RID... Prj Test Report

LOG  
Generated  
20150524 02:51:46 GMT -07:00  
4 minutes 12 seconds ago

### Prj Test Report

#### Summary Information

Status:

1 critical test failed

Start Time:

20150524 02:51:25.209

End Time:

20150524 02:51:45.968

Elapsed Time:

00:00:20.759

Log File:

log.html

#### Test Statistics

Total Statistics	Total	Pass	Fail	Elapsed	Pass / Fail
Critical Tests	2	1	1	00:00:20	<div></div>
All Tests	2	1	1	00:00:20	<div></div>

Statistics by Tag	Total	Pass	Fail	Elapsed	Pass / Fail
important	1	1	0	00:00:10	<div></div>

Statistics by Suite	Total	Pass	Fail	Elapsed	Pass / Fail
Prj	2	1	1	00:00:21	<div></div>
Prj.TestSuite	2	1	1	00:00:21	<div></div>

#### Test Details

Totals Tags Suites Search

Type:

Critical Tests

All Tests

Status:

2 total, 1 passed, 1 failed

Total Time:

00:00:20.228

Name	Documentation	Tags	Crit.	Status	Message	Elapsed	Start / End
Prj.TestSuite.Search			yes	FAIL	'robotframework_百度搜索' does not contain 'cucumber'	00:00:10.217	20150524 02:51:25.740 20150524 02:51:35.957
Prj.TestSuite.Search_Use_Keywords		important	yes	PASS		00:00:10.011	20150524 02:51:35.957 20150524 02:51:45.968

C:/Users/Administrator/AppData/Local/Temp/RIDEkhdal... Prj Test Log

REPORT

### Test Statistics

Total Statistics	Total	Pass	Fail	Elapsed	Pass / Fail
Critical Tests	2	1	1	00:00:20	<div></div>
All Tests	2	1	1	00:00:20	<div></div>

Statistics by Tag	Total	Pass	Fail	Elapsed	Pass / Fail
important	1	1	0	00:00:10	<div></div>

Statistics by Suite	Total	Pass	Fail	Elapsed	Pass / Fail
Prj	2	1	1	00:00:21	<div></div>
Prj.TestSuite	2	1	1	00:00:21	<div></div>

### Test Execution Log

TEST SUITE: Prj

00:00:20.759

Full Name: Prj

Source: C:\prj

Start / End / Elapsed: 20150524 02:51:25.209 / 20150524 02:51:45.968 / 00:00:20.759

Status: 2 critical test, 1 passed, 1 failed  
2 test total, 1 passed, 1 failed

TEST SUITE: TestSuite

00:00:20.648

Full Name: Prj.TestSuite

Source: C:\prj\TestSuite.txt

Start / End / Elapsed: 20150524 02:51:25.320 / 20150524 02:51:45.968 / 00:00:20.648

Status: 2 critical test, 1 passed, 1 failed  
2 test total, 1 passed, 1 failed

TEST CASE: Search

00:00:10.217

Full Name: Prj.TestSuite.Search

Start / End / Elapsed: 20150524 02:51:25.740 / 20150524 02:51:35.957 / 00:00:10.217

Status: FAIL (critical)

Message: 'robotframework\_百度搜索' does not contain 'cucumber'

SETUP: Selenium2Library.Open Browser http://www.baidu.com, ie

00:00:06.601

KEYWORD: Selenium2Library.Input Text id=kw, robotframework

00:00:00.862

KEYWORD: Screenshot.Take Screenshot

00:00:00.110

KEYWORD: Selenium2Library.Click Button id=su

00:00:00.769

KEYWORD: Selenium2Library.Go To https://www.baidu.com/s?ie=utf-8&f=8&rsv\_bp=1&rsv\_idx=1&tn=baidu&wd=robotframework&rsv\_pq=ac148f4c000326ef&rsv\_t=db1aQU6%2FFYdOpB8L0O6kCoCoFIVH%2F6U3G9kUpNFURiyivLWm0pB6vuMBvEQ&rsv\_enter=1&inputT=3606&rsv\_sug1=8&rsv\_n=2&rsv\_sug3=9&bs=cucumber

00:00:01.674

KEYWORD: \${title} = Selenium2Library.Get Title

00:00:00.030

KEYWORD: BuiltIn.Should Contain \${title}, cucumber

00:00:00.130

Documentation: Fails if 'item1' does not contain 'item2' one or more times.

Start / End / Elapsed: 20150524 02:51:35.827 / 20150524 02:51:35.957 / 00:00:00.130

02:51:35.947 FAIL 'robotframework\_百度搜索' does not contain 'cucumber'



# 执行测试用例

## 配置执行

The screenshot shows the RIDE - TestSuite application interface. The left sidebar displays a project tree with the following structure:

- Prj
  - TestSuite
    - Search (with a red 'X' icon)
    - Search\_Use\_Keywords (with a green icon)
  - Resourcefile.txt
  - Search (with a gear icon)
  - External Resources

The main window is titled 'Search\_Use\_Keywords' and contains a 'Settings <<' section. This section is highlighted with a red border and includes the following configuration fields:

- Documentation: A large text area for documentation, with 'Edit' and 'Clear' buttons.
- Setup: A text field for setup code, with 'Edit' and 'Clear' buttons.
- Teardown: A text field containing 'close browser', with 'Edit' and 'Clear' buttons.
- Timeout: A text field for timeout settings, with 'Edit' and 'Clear' buttons.
- Template: A text field for template settings, with 'Edit' and 'Clear' buttons.
- Tags: A text field containing 'important' and '<Add New>', with 'Edit' and 'Clear' buttons.

Below the settings section is a table with 6 rows and 5 columns. The first row is highlighted in blue and contains the following data:

	Search	cucumber	https://www.baidu.com/s?ie=utf-8	
1	Search	cucumber	https://www.baidu.com/s?ie=utf-8	
2				
3				
4				
5				
6				

# 执行测试用例

## 配置执行

EditText EditRun

Search\_Use\_Keywords

Settings <<

Documentation

EditClear

Setup

EditClear

Teardown

close browser

EditClear

Timeout

EditClear

Template

EditClear

Tags

important

<Add New>

EditClear

# 执行测试用例

## 配置执行

The screenshot displays the RIDE - TestSuite application window. The interface includes a menu bar (File, Edit, Tools, Navigate, Macros, Help) and a toolbar with various icons. On the left, a project tree shows the structure of the test suite, including a 'Prj' folder containing 'TestSuite', 'Search', 'Search\_Use\_Keywords', 'Resourcefile.txt', and 'External Resources'. The main area is titled 'TestSuite' and shows the source file 'C:\prj\TestSuite.txt'. A red box highlights the 'Settings <<' section, which contains the following configuration items:

Configuration Item	Value	Edit	Clear
Documentation			
Suite Setup			
Suite Teardown			
Test Setup	open browser   http://www.baidu.com   ie		
Test Teardown			
Test Template			
Test Timeout			
Force Tags	<Add New>		

Below the settings, there is a table for imports and a section for variables.

Import	Name / Path	Arguments	Comment
Library	Selenium2Library		
Resource	Resourcefile.txt		
Library	Screenshot		

On the right side of the interface, there are buttons for 'Add Import', 'Library', 'Resource', 'Variables', 'Import Failed Help', 'Add Scalar', and 'Add List'.

# 执行测试用例

## 配置执行

EditText Edit Run

TestSuite

SourceC:\prj\TestSuite.txt

Settings <<

Documentation

Suite Setup

Suite Teardown

Test Setup

Test Teardown

Test Template

Test Timeout

Force Tags

open browser | http://www.baidu.com | ie

<Add New>

EditClear

EditClear

EditClear

EditClear

EditClear

EditClear

EditClear

# 执行测试用例

## 配置执行

The screenshot displays the RIDE - TestSuite application window. The interface includes a menu bar (File, Edit, Tools, Navigate, Macros, Help) and a toolbar with various icons. On the left, a project tree shows the structure of the test suite, including 'Prj', 'TestSuite', 'Search', 'Search\_Use\_Keywords', 'Resourcefile.txt', and 'External Resources'. The main window is divided into two panes. The top pane, titled 'Run', contains execution controls: 'Execution Profile' set to 'pybot', buttons for 'Start', 'Stop', 'Pause', 'Continue', 'Next', and 'Step over', and checkboxes for 'Autosave', 'Pause on failure', and 'Show message log'. Below these are fields for 'Arguments' and 'Skip tests with these tags'. The 'Only run tests with these tags' checkbox is checked, and the tag 'important' is entered in the adjacent field. The bottom pane displays the execution output, showing the command used, the test results for 'Prj' and 'Prj.TestSuite', and the final status of the test run.

Execution Profile: **pybot** | Report | Log | ☒ Autosave | ☐ Pause on failure | ☒ Show message log

Arguments:  | ☐ Skip tests with these tags

☒ Only run tests with these tags  
important

elapsed time: 0:00:09 pass: 1 fail: 0

command: `pybot.bat --argumentfile c:\users\admini~1\appdata\local\temp\RIDEq5xf01.d\argfile.txt --listener C:\Python27\lib\sit`

Prj

Prj.TestSuite

Search\_Use\_Keywords | PASS |

Prj.TestSuite | PASS |

1 critical test, 1 passed, 0 failed

1 test total, 1 passed, 0 failed

Prj | PASS |

1 critical test, 1 passed, 0 failed

1 test total, 1 passed, 0 failed

Output: `c:\users\admini~1\appdata\local\temp\RIDEq5xf01.d\output.xml`

Log: `c:\users\admini~1\appdata\local\temp\RIDEq5xf01.d\log.html`

Report: `c:\users\admini~1\appdata\local\temp\RIDEq5xf01.d\report.html`

test finished 20150524 03:01:42

Starting test: Prj.TestSuite.Search\_Use\_Keywords

20150524 03:01:33.574 : INFO : Opening browser 'ie' to base url 'http://www.baidu.com'

20150524 03:01:37.383 : INFO : Typing text 'cucumber' into text field 'id=kq'

20150524 03:01:38.044 : INFO : Clicking button 'id=su'.

20150524 03:01:39.051 : INFO : Opening url 'https://www.baidu.com/s?ie=utf-8&f=8&rsv\_bp=0&rsv\_idx=1&tn=baidu&wd=cucumber&rsv\_'

20150524 03:01:39.930 : INFO : \${title} = cucumber\_百度搜索

Ending test: Prj.TestSuite.Search\_Use\_Keywords

# 执行测试用例

## 配置执行

```
elapsed time: 0:00:09  pass: 1  fail: 0
command: pybot.bat --argumentfile c:\users\admini~1\appdata\local\temp\RIDEq5xf01.d\argfile.txt --listener C:\Python27\lib\sit
=====
Prj
=====
Prj.TestSuite
=====
Search_Use_Keywords | PASS |
=====
Prj.TestSuite | PASS |
1 critical test, 1 passed, 0 failed
1 test total, 1 passed, 0 failed
=====
Prj | PASS |
1 critical test, 1 passed, 0 failed
1 test total, 1 passed, 0 failed
=====
Output: c:\users\admini~1\appdata\local\temp\RIDEq5xf01.d\output.xml
Log: c:\users\admini~1\appdata\local\temp\RIDEq5xf01.d\log.html
Report: c:\users\admini~1\appdata\local\temp\RIDEq5xf01.d\report.html

test finished 20150524 03:01:42
```

```
Starting test: Prj.TestSuite.Search_Use_Keywords
20150524 03:01:33.574 : INFO : Opening browser 'ie' to base url 'http://www.baidu.com'
20150524 03:01:37.383 : INFO : Typing text 'cucumber' into text field 'id=kw'
20150524 03:01:38.044 : INFO : Clicking button 'id=su'.
20150524 03:01:39.051 : INFO : Opening url 'https://www.baidu.com/s?ie=utf-8&f=8&rsv_bp=0&rsv_idx=1&tn=baidu&wd=cucumber&rsv_
20150524 03:01:39.930 : INFO : ${title} = cucumber_百度搜索
Ending test: Prj.TestSuite.Search_Use_Keywords
```

# 执行测试用例

## 配置执行

The screenshot displays the RIDE - TestSuite application interface. The left sidebar shows a project tree with 'Prj' containing 'TestSuite', 'Search', 'Search\_Use\_Keywords', 'Resourcefile.txt', and 'External Resources'. The main window is divided into several sections:

- Execution Profile:** Set to 'pybot'.
- Arguments:** '-t Search' (highlighted with a red box).
- Options:** 'Autosave' and 'Show message log' are checked.
- Execution Controls:** Buttons for Start, Stop, Pause, Continue, Next, and Step over.
- Execution Log:** Shows the command: `pybot.bat -t Search --argumentfile c:\users\admini~1\appdata\local\temp\RIDEq5xf01.d\argfile.txt --listener C:\Pyth`.
- Test Results:** A summary of the test execution:
  - Prj**: 1 critical test, 0 passed, 1 failed
  - Prj.TestSuite**: 1 test total, 0 passed, 1 failed
  - Search**: 1 critical test, 0 passed, 1 failed
- Output:** `c:\users\admini~1\appdata\local\temp\RIDEq5xf01.d\output.xml`
- Log:** `c:\users\admini~1\appdata\local\temp\RIDEq5xf01.d\log.html`
- Report:** `c:\users\admini~1\appdata\local\temp\RIDEq5xf01.d\report.html`
- Unexpected error:** Exception in thread Thread-1 (most likely raised during interpreter shutdown):
- Traceback:** (most recent call last):

The bottom section shows the detailed execution of the 'Search' test case:

```
Starting test: Prj.TestSuite.Search
20150524 03:03:45.221 : INFO : Opening browser 'ie' to base url 'http://www.baidu.com'
20150524 03:03:48.929 : INFO : Typing text 'robotframework' into text field 'id=kw'
20150524 03:03:49.881 : INFO : <a href="screenshot_1.jpg"></a>
20150524 03:03:49.881 : INFO : Clicking button 'id=su'.
20150524 03:03:50.860 : INFO : Opening url 'https://www.baidu.com/s?ie=utf-8&f=8&rsv_bp=1&rsv_idx=1&tn=baidu&wd=robotframewor
20150524 03:03:51.771 : INFO : ${title} = robotframework_百度搜索
20150524 03:03:51.781 : FAIL : 'robotframework_百度搜索' does not contain 'cucumber'
Ending test: Prj.TestSuite.Search
```

# 执行测试用例

## 配置执行

```

elapsed time: 0:00:08  pass: 0  fail: 1
command: pybot.bat -t Search --argumentfile c:\users\admini~1\appdata\local\temp\RIDEq5xf01.d\argfile.txt --listener C:\Pyth
=====
Prj
=====
Prj.TestSuite
=====
Search | FAIL |
'robotframework_百度搜索' does not contain 'cucumber'
-----
Prj.TestSuite | FAIL |
1 critical test, 0 passed, 1 failed
1 test total, 0 passed, 1 failed
=====
Prj | FAIL |
1 critical test, 0 passed, 1 failed
1 test total, 0 passed, 1 failed
=====
Output: c:\users\admini~1\appdata\local\temp\RIDEq5xf01.d\output.xml
Log: c:\users\admini~1\appdata\local\temp\RIDEq5xf01.d\log.html
Report: c:\users\admini~1\appdata\local\temp\RIDEq5xf01.d\report.html
unexpected error: Exception in thread Thread-1 (most likely raised during interpreter shutdown):
Traceback (most recent call last):

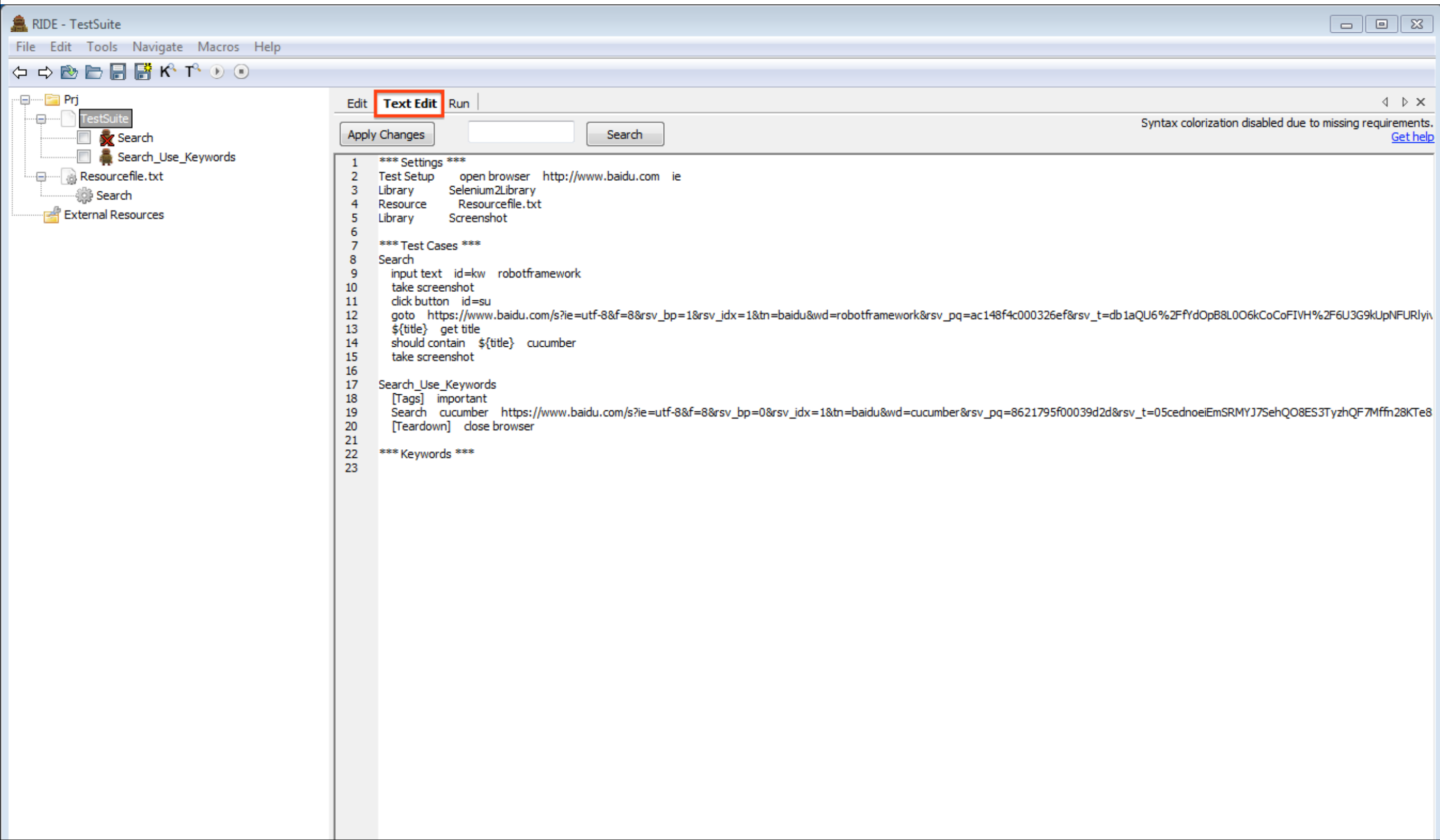
```

```
Starting test: Prj.TestSuite.Search
20150524 03:03:45.221 : INFO : Opening browser 'ie' to base url 'http://www.baidu.com'
20150524 03:03:48.929 : INFO : Typing text 'robotframework' into text field 'id=kw'
20150524 03:03:49.881 : INFO : <a href="screenshot_1.jpg"></a>
20150524 03:03:49.881 : INFO : Clicking button 'id=su'.
20150524 03:03:50.860 : INFO : Opening url 'https://www.baidu.com/s?ie=utf-8&f=8&rsu_bp=1&rsu_idx=1&tn=baidu&wd=robotframewor
20150524 03:03:51.771 : INFO : ${title} = robotframework_百度搜索
20150524 03:03:51.781 : FAIL : 'robotframework_百度搜索' does not contain 'cucumber'
Ending test: Prj.TestSuite.Search
```



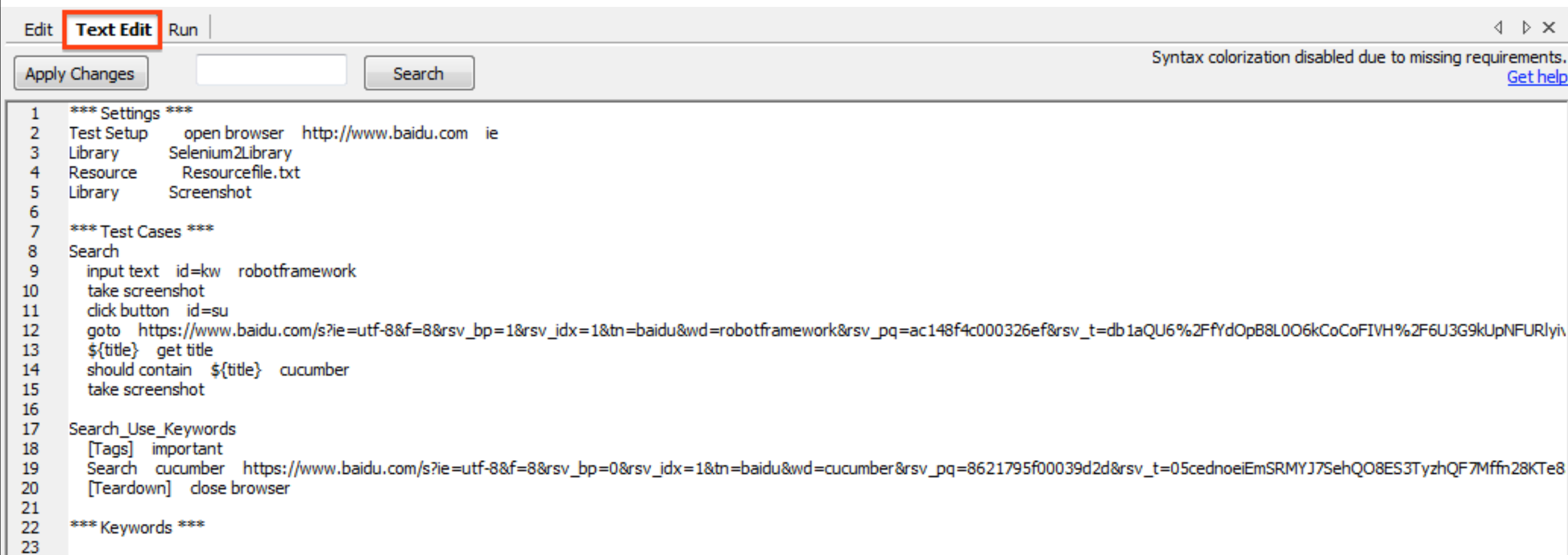
# 执行测试用例

## 测试用例文件



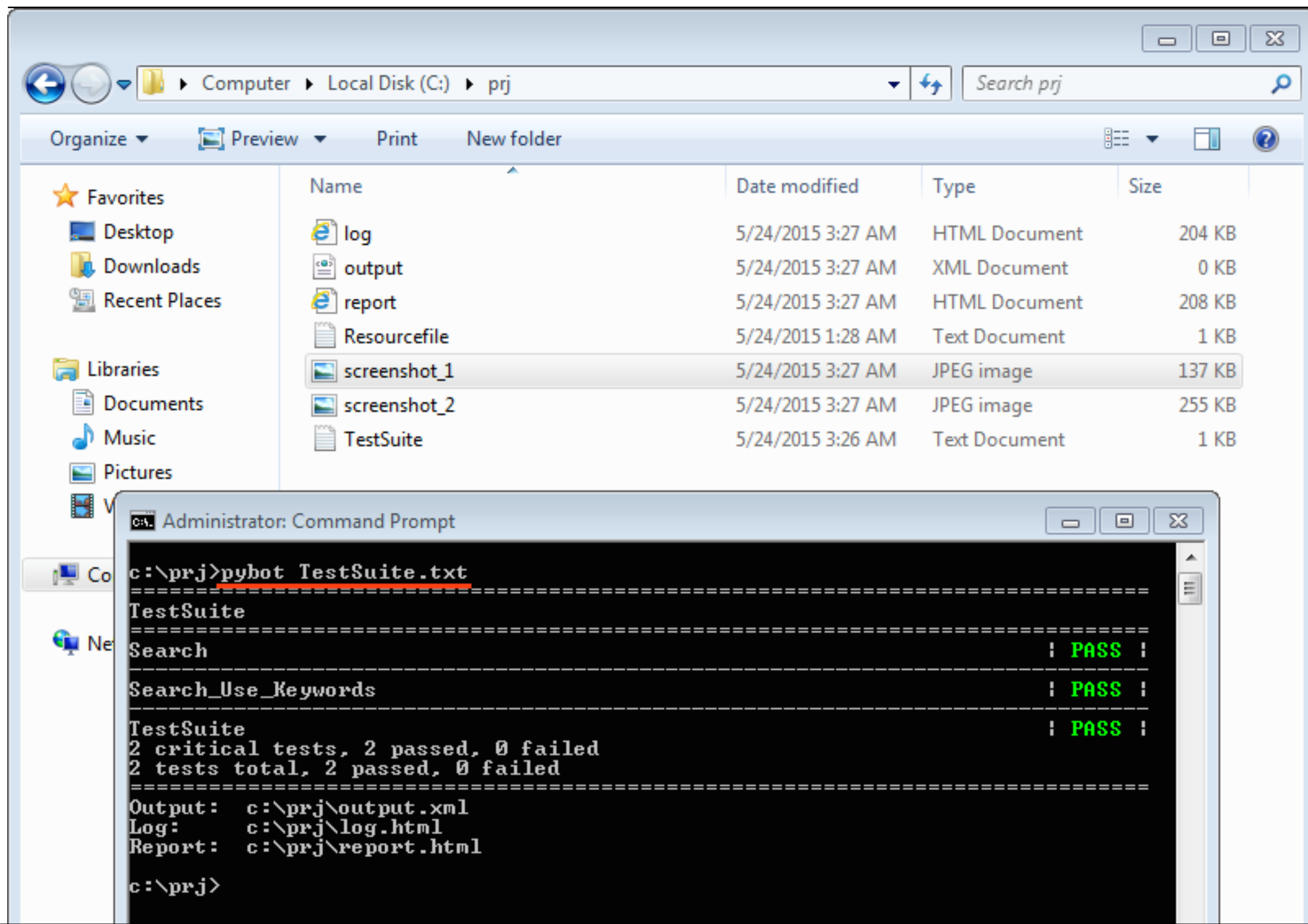
# 执行测试用例

## 测试用例文件



```
1 *** Settings ***
2 Test Setup      open browser    http://www.baidu.com    ie
3 Library         Selenium2Library
4 Resource        Resourcefile.txt
5 Library         Screenshot
6
7 *** Test Cases ***
8 Search
9     input text    id=kf    robotframework
10    take screenshot
11    click button   id=su
12    goto           https://www.baidu.com/s?ie=utf-8&f=8&rsv_bp=1&rsv_idx=1&tn=baidu&wd=robotframework&rsv_pq=ac148f4c000326ef&rsv_t=db1aQU6%2FfydOpB8L0O6kCoCoFIVH%2F6U3G9kUpNFURlyiv
13    ${title}      get title
14    should contain ${title}    cucumber
15    take screenshot
16
17 Search_Use_Keywords
18     [Tags]    important
19     Search    cucumber    https://www.baidu.com/s?ie=utf-8&f=8&rsv_bp=0&rsv_idx=1&tn=baidu&wd=cucumber&rsv_pq=8621795f00039d2d&rsv_t=05cednoeiEmSRMYJ7SehQO8ES3TyzhQF7Mffn28KTe8
20     [Teardown]    close browser
21
22 *** Keywords ***
23
```

# 执行测试用例



# 执行测试用例

pybot\_commands - Notepad

File Edit Format View Help

Robot Framework -- A generic test automation framework

Version: 2.8.7 (Python 2.7.9 on win32)

Usage: pybot|jybot|ipybot [options] data\_sources  
or: python|jython|ipy -m robot.run [options] data\_sources  
or: python|jython|ipy path/to/robot/run.py [options] data\_sources  
or: java -jar robotframework.jar run [options] data\_sources

Robot Framework is a Python-based keyword-driven test automation framework for acceptance level testing and acceptance test-driven development (ATDD). It has an easy-to-use tabular syntax for creating test cases and its testing capabilities can be extended by test libraries implemented either with Python or Java. Users can also create new keywords from existing ones using the same simple syntax that is used for creating test cases.

Depending is Robot Framework installed using Python, Jython, or IronPython interpreter, it has a start-up script, 'pybot', 'jybot' or 'ipybot', respectively. Alternatively, it is possible to directly execute 'robot.run' module (e.g. 'python -m robot.run') or 'robot/run.py' script using a selected interpreter. Finally, there is also a standalone JAR distribution.

Data sources given to Robot Framework are either test case files or directories containing them and/or other directories. Single test case file creates a test suite containing all the test cases in it and a directory containing test case files creates a higher level test suite with test case files or other directories as sub test suites. If multiple data sources are given, a virtual top level suite containing suites generated from given data sources is created.

By default Robot Framework creates an XML output file and a log and a report in HTML format, but this can be configured using various options listed below. Outputs in HTML format are for human consumption and XML output for integration with other systems. XML outputs can also be combined and otherwise further processed with 'rebot' tool. Run 'rebot --help' for more information.

Robot Framework is open source software released under Apache License 2.0. Its copyrights are owned and development supported by Nokia Solutions and Networks. For more information about the framework see <http://robotframework.org/>.

Options

=====

-N --name name	Set the name of the top level test suite. Underscores in the name are converted to spaces. Default name is created from the name of the executed data source.
-D --doc documentation	Set the documentation of the top level test suite. Underscores in the documentation are converted to spaces and it may also contain simple HTML formatting (e.g. *bold* and <a href="http://url/">http://url/</a> ).
-M --metadata name:value	* Set metadata of the top level suite. Underscores in the name and value are converted to spaces. Value can contain same HTML formatting as --doc. Example: --metadata version:1.2
-G --settag tag *	Sets given tag(s) to all executed test cases.
-t --test name *	Select test cases to run by name or long name. Name is case and space insensitive and it can also be a simple pattern where '*' matches anything and '?' matches any char. If using '*' and '?' in the console is problematic see --escape and --argumentfile.
-s --suite name *	Select test suites to run by name. When this option is used with --test, --include or --exclude, only test cases in matching suites and also matching other filtering criteria are selected. Name can be a simple pattern similarly as with --test and it can contain parent name separated with a dot. For example -s X.Y selects suite 'Y' only if its parent is 'X'. i. include tag *

Administrator: Command Prompt

c:\prj>pybot -h > pybot\_commands.txt

c:\prj>

pybot -h

# 参考链接

---

Cucumber:

[http://v.youku.com/v\\_show/id\\_XMjU2Njk4NjQw.html?  
from=s1.8-1-1.2](http://v.youku.com/v_show/id_XMjU2Njk4NjQw.html?from=s1.8-1-1.2)

自动化测试框架Cucumber和RobotFramework的实战对比:

[http://www.infoq.com/cn/articles/cucumber-robotframework-  
comparison](http://www.infoq.com/cn/articles/cucumber-robotframework-comparison)

# 总结

---

- 使用形式统一且容易使用的表格语法来创建测试用例。
- 能够从已有的关键字创建可重用的高层关键字。
- 提供易于阅读的HTML格式的结果报告和日志。
- 提供标签以分类和选择执行的测试用例。
- 提供测试用例和测试集级别的setup 和teardown。
- .....

Q & A

Thanks!

