

---

# InterfaceLibrary Documentation

*Release 0.0.9*

**RFUI**

August 12, 2016



<b>1</b>	<b>Interface testing Keywords methods</b>	<b>3</b>
<b>2</b>	<b>Indices and tables</b>	<b>7</b>



Contents:



---

## Interface testing Keywords methods

---

**class** InterfaceLibrary.HttpLibrary.**HttpLibrary**

Http Client

**create\_session** (*host, port, Alias*)

Create Session : host serverHTTP

*Alias* ,HttpSession

robotframework-requestscreate session

Examples: | Create Session | &{enviro}[host] | &{enviro}[port] | alias |

**delete\_request** (*uri, para, data, headers*)

Issues a HTTP DELETE request. *headers*

**encrypt\_post** (*para, body, headers*)

Issues a encrypt Http Post Request

*para* json

*body*

*headers* Id response

Examples: | \${enres} | encrypt post | {"c":""100"} | \${enbody} | {'Content-Type':'text/plain','time':\${strtime},'unique':\${unique},'sign':\${sign},'uid':'','ua':'IOS/1.0'}

**get\_request** (*uri, para=None*)

Issues a HTTP Get Request.

*para* URLJsonURL1024.

*uri* hosturl

Examples: | \${res} | Get Request | /cgi-bin/jssdk/ticket / { 'access\_token': '\${acces\_token}' }

!\${res} | post request | /weblogin/user/generate | {"mac":""\${mac}"} | None | | \${resdic} | Evaluate | \${res} | | Set Suite Variable | \${url} | \${resdic['result']['url']} | | \${res} | get request | \${url} | { 'uid': '\${uidencode}', 'timestamp': '\${strtime}', 'signature': '\${sign}', 'unique': '\${unique}' } |

**head\_request** (*uri, headers*)

Issues a HTTP HEAD request. *headers*

**post\_file\_request** (*uri, file, data=None, para=None*)

Issues a HTTP Post Request to upload file.

*file* file must be dictionary of { filename: fileobject } files to multipart upload.

*para* dictionary of URL parameters to append to the URL.

*data* the body to attach to the request must be a python dictionary.

Examples: | `Post File Request | /cgi-bin/file/upload | {'content':open('Resources/Material/logo.jpg','rb')} | {'type':'jpg'} | {'access_token':'${access_token}'}` |

**post\_request** (*uri, data=None, para=None*)

Issues a HTTP POST request.

*uri*

*para* URLJsonURL1024.

*data* POSTHTTPJson

robotframework-requestspost request

Examples: | `Post Request | /checkUser.do | {'user-Name': '&{account}[username]','password': '&{account}[password]'} | None | #cookie | ${res} | Post Request | /manualreply/addClassify.p | {'name': '${name}'} | None |`

**class** InterfaceLibrary.CheckResult.**CheckResult**

docstring for CheckResult

**check\_code** (*response, expectcode*)

Check HTTP response code.

Examples usage: | `Check | {"data": "null", "code": "200", "msg": ""} | 200 |`

**check\_if\_exists\_in\_database** (*sql*)

Check if no rows would be returned by given the input 'SQL' statement. If there are any results, then this will throw an AssertionError.

Example usage: | `Check If Not Exists In Database | SELECT name,company_id FROM elf_staff_tab WHERE id=337 |`

**check\_if\_not\_exists\_in\_database** (*sql*)

Check if no rows would be returned by given the input *selectStatement*. If there are any results, then this will throw an AssertionError.

Example usage: | `Check If Not Exists In Database | SELECT name,remark FROM elf_activity_tab WHERE id=337 |` If the id is not exist, the output is: No date return #pass

**connect\_to\_database** (*dbname, dbusername, dbpassword, dbhost, dbport=3306*)

Example usage: | `Connect To Database | dbname | dbusername | dbpassword | dbhost |`

**delete\_rows\_from\_table** (*condition, table*)

Delte all the rows within a given table and condition.

Example usage: | `Delete Rows FROM Table | id=337 | elf_activity_tab |`

**execute\_mysql\_script** (*sqlScriptFileName*)

Executes the sql sript file of the *sqlScriptFileName* as SQL commands. lines that starts with a number sign (#, -) are treated as a commented line. The sql script can include 'begin end' function.

Examples usage: | `Execute Mysql Script | ${CURDIR}test_scriptsqlfile.sql |`

**generate\_date** (*days=0*)

generate date based on current date Example usage: | `Generate Date | 5 | # Generate the date after 5 days from today | Generate Date | -5 | # Generate the date before 5 days from today`



**generate\_random\_int** (*n2, n1=1*)

generate random int between n1 and n2

Examples usage: | Generate Random Int | 100 |

Generate Random Int | 100 | 5

**mysql\_select** (*sql*)

Execute mysql SELECT statement,now it default handle the fist row of the select datas, if select one element,it return a string,if select multiple elements,it return a list.

Examples usage: | Mysql Select | SELECT name FROM elf\_activity\_tab WHERE ID = 3| this examples return a string

Mysql Select | SELECT name,remark FROM elf\_activity\_tab WHERE ID = 3|

this examples return a list

**random\_index** (*str, key*)

generate random int from http response keyword

Examples usage: | Random Index | \${res} | id | this examples return a int between 0 and id term length - 1



---

## Indices and tables

---

- `genindex`
- `modindex`
- `search`



**C**

check\_code() (InterfaceLibrary.CheckResult.CheckResult method), 4

check\_if\_exists\_in\_database() (InterfaceLibrary.CheckResult.CheckResult method), 4

check\_if\_not\_exists\_in\_database() (InterfaceLibrary.CheckResult.CheckResult method), 4

CheckResult (class in InterfaceLibrary.CheckResult), 4

connect\_to\_database() (InterfaceLibrary.CheckResult.CheckResult method), 4

create\_session() (InterfaceLibrary.HttpLibrary.HttpLibrary method), 3

**D**

delete\_request() (InterfaceLibrary.HttpLibrary.HttpLibrary method), 3

delete\_rows\_from\_table() (InterfaceLibrary.CheckResult.CheckResult method), 4

**E**

encrypt\_post() (InterfaceLibrary.HttpLibrary.HttpLibrary method), 3

execute\_mysql\_script() (InterfaceLibrary.CheckResult.CheckResult method), 4

**G**

generate\_date() (InterfaceLibrary.CheckResult.CheckResult method), 4

generate\_random\_int() (InterfaceLibrary.CheckResult.CheckResult method), 4

get\_request() (InterfaceLibrary.HttpLibrary.HttpLibrary method), 3

**H**

head\_request() (InterfaceLibrary.HttpLibrary.HttpLibrary method), 3

HttpLibrary (class in InterfaceLibrary.HttpLibrary), 3

**M**

mysql\_select() (InterfaceLibrary.CheckResult.CheckResult method), 5

**P**

post\_file\_request() (InterfaceLibrary.HttpLibrary.HttpLibrary method), 3

post\_request() (InterfaceLibrary.HttpLibrary.HttpLibrary method), 4

**R**

random\_index() (InterfaceLibrary.CheckResult.CheckResult method), 5