

Setup

Install library	<code>pip install robotframework-okw-api-rest</code>
Import in <i>Settings</i>	<code>Library okw-api-rest.library.O kw Api RestLibrary WITH NAME RESTAPI</code>
Minimum Python version	Python >= 3.10
Minimum RF version	robotframework >= 6.0

YAML locator file required: locators/ServiceName.yaml with `__self__.base_url` and `__self__.content_type`. See README for full YAML examples.

Start / Stop

RESTStart service	Start REST service (load YAML)
RESTStart service env	Start with environment file
RESTStop	Stop REST service

YAML config separates URLs and credentials from test code. Environment files are searched in: `~/okw/env/` → `$OKW_ENV_DIR` → `locators/` → OS env vars.

Scope

RESTSelectEndpoint /path	Select endpoint (resets body/headers)
RESTSetContext path	Navigate into nested JSON
RESTSetContext customer.address	Deeper nesting via dot notation
RESTSetContext items[0]	Select array element

RESTSelectEndpoint resets body, headers, and context. Each RESTSetContext replaces the previous one (flat, no stack).

Input

RESTSetValue field value	Set body field
RESTSetValueAsString field value	Always send the value as a string (no auto-conversion). Example: <code>RESTSetValueAsString zipcode 01234 # -> "01234"</code> <code>RESTSetValueAsString flag true # -> "true"</code>
RESTSetValue ?param value	Set query parameter (? prefix)
RESTSetHeader name value	Set request header



By **ZH** (openkeyword)

Published 21st June, 2026.
Last updated 21st June, 2026.
Page 2 of 8.

Sponsored by **Readable.com**
Measure your website readability!
<https://readable.com>

Input (cont)

RESTSetFile field filepath [mime_type] Set a file for multipart form-data upload. MIME type is auto-detected. Call multiple times for multiple files.

Example:

RESTSetFile avatar photo.jpg

RESTSetFile document report.pdf application/pdf

Multiple files (same field):

RESTSetFile attachments file1.jpg

RESTSetFile attachments file2.jpg

? prefix sends field as URL query parameter instead of body field. Query parameters are not affected by RESTSetContext. Password fields are masked (*) in logs.

Action

RESTSendRequest GET	Send GET request
RESTSendRequest POST	Send POST request
RESTSendRequest PUT	Send PUT request
RESTSendRequest PATCH	Send PATCH request
RESTSendRequest DELETE	Send DELETE request

Request and response bodies are logged as formatted JSON. After sending, all RESTVerify and RESTMemorize keywords operate on this response.

Auto Type Detection

RESTSetValue detects the JSON type automatically:

Zoltan -> "Zoltan" (string)

42 -> 42 (integer)

3.14 -> 3.14 (float)

true / false -> true / false (boolean)

null / \$NULL -> null

\$EMPTY -> "" (empty string)

Force string: Use RESTSetValueAsString

RESTSetValueAsString zipcode 01234 # -> "01234"



By ZH (openkeyword)

Published 21st June, 2026.

Last updated 21st June, 2026.

Page 3 of 8.

Sponsored by **Readable.com**

Measure your website readability!

<https://readable.com>

Verify

RESTVerifyStatus 200	Verify HTTP status code
RESTVerifyValue field expected	Verify field value (exact match)
RESTVerifyValueWCM field <i>text</i>	Wildcard match (* ?)
RESTVerifyValueREGX field <i>^[a-f]+\$</i>	Regex match
RESTVerifyHeader name expected	Verify response header
RESTVerifyResponseTime 500	Response time < 500ms
RESTVerifyListCount field expected	Verify the number of elements in a JSON array field. Context-aware. Example: RESTVerifyListCount todos 3 RESTVerifyListCount items 0

Dot notation for nested fields: data.user.name. With active context, fields resolve relative to context path. ResponseTime checks elapsed time, not timeout.

Memorize

RESTMemorizeValue field NAME	Store response field value
RESTMemorizeBody NAME	Store entire response body
\$MEM{NAME}	Use stored value in next request

Use \$MEM{NAME} in any keyword: RESTSetValue, RESTSetHeader, RESTSelectEndpoint. Missing keys cause an immediate error. Values persist across endpoints until RESTStop.

Tokens

\$IGNORE	Skip keyword (no-op)
\$EMPTY	Empty string
\$MEM{KEY}	Stored value from RESTMemorizeValue
\$NULL	Set field to JSON null. Example: RESTSetValue comment \$NULL # -> "comment": null

RESTSetValue email \$IGNORE → field not sent. RESTSetValue email \$EMPTY → field sent as "". RESTSetHeader x-auth-token \$MEM{TOKEN} → inserts stored token value.



By ZH (openkeyword)

Published 21st June, 2026.
Last updated 21st June, 2026.
Page 4 of 8.

Sponsored by **Readable.com**
Measure your website readability!
<https://readable.com>

Phase Model

Start	RESTStart
Scope	RESTSelectEndpoint, RESTSetContext
Input	RESTSetValue, RESTSetValueAsString, RESTSetValueAsList, RESTSetFile, RESTSetHeader
Action	RESTSendRequest
Verify	RESTVerifyValue, RESTVerifyValueWCM, RESTVerifyValueREGX, RESTVerifyStatus, RESTVerifyResponseTime, RESTVerifyListCount, RESTVerifyHeader
Memorize	RESTMemorizeValue, RESTMemorizeBody
Save	RESTSaveResponseToFile
Stop	RESTStop
Config (YAML)	Retry, OAuth 2.0, Cookie Handling

Every REST test follows this sequence. Same pattern as OKW GUI keywords: StartApp → SelectWindow → SetValue → ClickOn → VerifyValue → StopApp.

Total: 20 keywords

vs. Playwright (Code-Based)

data: { title: 'Test' }	RESTSetValue title Test
expect(response.status()).toBe(201)	RESTVerifyStatus 201
const body = await response.json()	Not needed — automatic
expect(body.title).toBe('Test')	RESTVerifyValue title Test
if (email) payload.email = email	RESTSetValue email \$IGNORE
{ nested: { key: 'val' } }	RESTSetContext nested + RESTSetValue key val
File Upload	OKW: RESTSetFile PW: multipart option
Array Values	OKW: RESTSetValueAsList PW: JSON inline
Auto Typing	OKW: automatic PW: manual (JSON)
Cookie Handling	OKW: automatic PW: context.cookies()
Retry	OKW: YAML config PW: custom logic
OAuth 2.0	OKW: YAML config PW: custom logic
Response to File	OKW: RESTSaveResponseToFile PW: fs.writeFile()

No JSON syntax, no async/await, no variable assignments. Field name + value — readable by testers, BAs, and AI. Same keyword model for GUI and API testing.



By **ZH (openkeyword)**

Published 21st June, 2026.
Last updated 21st June, 2026.
Page 5 of 8.

Sponsored by **Readable.com**
Measure your website readability!
<https://readable.com>

Save

RESTSaveResponseToFile filepath Save response body (bytes) to a file. Binary-safe (PDF, ZIP, images).
Directories are created automatically. Existing file is overwritten (warning in log).
Supports: `\${MEM}`, ~, \$ENV_VAR, \$IGNORE'
Example:
RESTSaveResponseToFile C:/temp/response.json
RESTSaveResponseToFile ~/downloads/report.pdf

Arrays

RESTSetValueAsList field val1 val2 ... **Compact:**
RESTSetValueAsList tags a b c
-> "tags": ["a", "b", "c"]

RESTSetValue field[index] value **Index syntax:**
RESTSetValue scores[0] 42
RESTSetValue scores[1] 87
-> "scores": [42, 87]

RESTSetValueAsList field **Empty array:**
RESTSetValueAsList items
-> "items": []

RESTVerifyListCount field count **Verify length:**
RESTVerifyListCount tags 3

File Upload

RESTSetFile field filepath [mime] Set file for multipart upload. MIME is auto-detected.

RESTSetFile + RESTSetValue Text fields from RESTSetValue are sent as form fields alongside files.



By **ZH** (openkeyword)

Published 21st June, 2026.
Last updated 21st June, 2026.
Page 6 of 8.

Sponsored by **Readable.com**
Measure your website readability!
<https://readable.com>

cheatography.com/openkeyword/

File Upload (cont)

Multiple files all `RESTSetFile` multiple times — files accumulate until `RESTSelectEndpoint` resets them.

`RESTSendRequest` automatically switches from JSON to multipart/form-data when files are present.

Example:

```
RESTSelectEndpoint /api/upload
RESTSetValue title Report
RESTSetFile file report.pdf
RESTSendRequest POST
```

Retry

<code>retry_count: 3</code>	Max retries (default: 0 = off)
<code>retry_delay: 1000</code>	Delay between retries in ms (default: 1000)
<code>retry_on: 429,502,503</code>	Status codes that trigger retry

Configure in YAML `__self__` — no keyword needed. After all retries exhausted: last status is kept, `RESTVerifyStatus` will fail.

Log:

```
<<< 429 — retry 1/3 (waiting 1000ms)
<<< 429 — retry 2/3 (waiting 1000ms)
<<< 200 OK
```

OAuth 2.0

<code>auth_type: oauth2_client_credentials</code>	Machine-to-machine API access
<code>token_url: https://auth.example.com/oauth/token</code>	OAuth token endpoint
<code>client_id / client_secret: \${...}</code>	From env file (<code>~/.okw/env/env-test.yaml</code>)
<code>scope: read write</code>	Space-separated scopes (optional)

`RESTStart` fetches the access token automatically. Authorization: Bearer <token> is set on every request.

Test:

```
RESTStart MyAPI env-test
RESTSelectEndpoint /api/data
RESTSendRequest GET
RESTVerifyStatus 200
RESTStop
```

Cookie Handling

Automatic (requests.Session)	Server sends Set-Cookie -> cookie is stored. Subsequent requests send the cookie automatically. No new keyword needed.
------------------------------	--

Example (login with session cookie):

```
RESTSelectEndpoint /auth/login
RESTSetValue username admin
RESTSendRequest POST # Set-Cookie stored
RESTSelectEndpoint /api/data # cookie sent automatically
RESTSendRequest GET
RESTVerifyStatus 200
```



Header Logging

Request headers Logged after >>> POST url in cleartext

Response headers Logged after <<< 200 OK in cleartext

All request and response headers are logged in cleartext for maximum observability. Body fields (password, token, secret) are masked with ***.

Log:

>>> POST https://api.example.com/login

Headers:

Content-Type: application/json

<<< 200 OK

Headers:

Set-Cookie: JSESSIONID=abc123; Path=/; HttpOnly

YAML Config

Basic YAML ServiceName.yaml in locators/ folder:

```
ServiceName:
  __self__:
    class: okw_api_rest.library.OkwApiRestLibrary
    base_url: https: //a pi.e xa mpl e.com
    conten t_type: applic ati on/json
```

With Environment File

YAML with \${VAR} placeholders — resolved from env file:

```
ServiceName:
  __self__:
    base_url: ${BASE_URL}
    conten t_type: ${CONTENT_TYPE}
```

env-test.yaml (in ~/.okw /env/):

```
BASE_URL: https: //a pi.e xa mpl e.com
CONTENT_TYPE: applic ati on/json
```



By ZH (openkeyword)

cheatography.com/openkeyword/

Published 21st June, 2026.

Last updated 21st June, 2026.

Page 8 of 8.

Sponsored by **Readable.com**

Measure your website readability!

<https://readable.com>

YAML Config (cont)

Authentication (YAML)

Basic Auth:

```
'auth_type: basic
auth_user: ${API_USER}
auth_password: ${API_PASSWORD}
```

API Key:

```
auth_type: api_key
auth_header: X-API-Key
auth_key: ${API_KEY}
```

Bearer Token:

```
auth_type: bearer
auth_token: ${AUTH_TOKEN}'
```

SSL / Certificates

```
verify_ssl: false # self-signed certs
client_cert: /.okw/certs/client.pem
client_key: /.okw/certs/client.key
ca_bundle: ~/.okw/certs/ca-bundle.pem
```

All settings go in `__self__`. Credentials use `${VAR}` placeholders resolved from env files — no secrets in YAML. OAuth 2.0 and Retry are also configured here (see separate blocks).



By **ZH** (openkeyword)

cheatography.com/openkeyword/

Published 21st June, 2026.

Last updated 21st June, 2026.

Page 9 of 8.

Sponsored by **Readable.com**

Measure your website readability!

<https://readable.com>