
Zowe Client Python SDK

Release 0.2.0

Contributors to the Zowe Project

May 31, 2023

CONTENTS:

1	Getting started	3
1.1	Installation	3
1.2	Basic usage	3
2	About the project	5
3	Zowe SDK packages	7
3.1	Core	7
3.2	z/OS Console	10
3.3	z/OS Files	11
3.4	z/OS Jobs	12
3.5	z/OS TSO	14
3.6	z/OSMF	16
4	Classes	17
5	Contributing	19
5.1	Git branching model	19
5.2	Pull requests	19
5.3	Code standards	19
	Index	21

The Zowe Client Python SDK, is a set of Python packages designed to allow programmatic interactions with z/OS REST API interfaces.

Python developers can leverage the Zowe SDK in order to create powerful scripts/applications that can interact with z/OS components.

GETTING STARTED

1.1 Installation

When installing the Zowe Client Python SDK you have two options.

- Install all the Zowe packages
- Install a single sub-package

The choice depends on your intentions. If you chose to install all Zowe SDK packages this means that you will install everything under the *zowe* namespace in PyPi.

Alternatively you can chose to install only a single subpackage for a smaller installation.

To install all Zowe SDK packages using pip:

```
pip install zowe
```

To install only a subpackage using pip:

```
pip install zowe.<subpackage>_for_zowe_sdk
```

To see all available sub-packages check the [Zowe SDK packages](#) section.

1.2 Basic usage

After you install the package in your project, import the class for the required sub-package (i.e *Console* class for z/OS Console commands). Create a dictionary to handle communication with the plug-in:

```
from zowe.zos_console_for_zowe_sdk import Console
connection = {
    "host_url": "'<host address>' ",
    "user": "<user>",
    "password": "<password>",
}

my_console = Console(connection)
```

Alternatively you can use an existing Zowe CLI profile instead:

```
from zowe.zos_console_for_zowe_sdk import Console

connection = {
    "plugin_profile": "<profile name>"
}

my_console = Console(connection)
```


ABOUT THE PROJECT

The Zowe Client Python SDK, is a set of Python packages designed to allow programmatic interactions with z/OS REST API interfaces.

Python developers can leverage the Zowe SDK in order to create powerful scripts/applications that can interact with z/OS components.

ZOWE SDK PACKAGES

The Zowe Client Python SDK is divided into multiple packages. Each one responsible for a single aspect of the SDK.

3.1 Core

Contains the core functionalities for the Zowe Client Python SDK.

3.1.1 Installing

To install this package using pip issue the following command:

```
pip install zowe.zowe_core_for_zowe_sdk
```

3.1.2 Reference

Core Classes

Classes belonging to the Zowe Client Python SDK core package.

SdkApi

class `core.zowe.core_for_zowe_sdk.SdkApi`(*connection, default_url*)

Abstract class used to represent the base SDK API.

connection

A dictionary containing the connection arguments

Type

dict

default_url

The default endpoint for the API

Type

str

create_custom_request_arguments()

Create a copy of the default request arguments dictionary.

This method is required because the way that Python handles dictionary creation

ZosmfProfile

class `core.zowe.core_for_zowe_sdk.ZosmfProfile(profile_name)`

Class used to represent a Zowe z/OSMF profile.

Description

This class is only used when there is already a Zowe z/OSMF profile created and the user opted to use the profile instead of passing the credentials directly in the object constructor.

profile_name

Zowe z/OSMF profile name

Type

str

load()

Load z/OSMF connection details from a z/OSMF profile.

Returns

z/OSMF connection object

Return type

zosmf_connection

property profiles_dir

Return the os path for the Zowe z/OSMF profiles.

ApiConnection

class `core.zowe.core_for_zowe_sdk.ApiConnection(host_url, user, password, ssl_verification=True)`

Class used to represent a connection with a REST API.

host_url

The base url of the rest api host

Type

str

user

The user of the rest api

Type

str

password

The password for the user

Type

str

ssl_verification

Type

bool

RequestHandler

class `core.zowe.core_for_zowe_sdk.RequestHandler`(*session_arguments*)

Class used to handle HTTP/HTTPS requests.

session_arguments

Zowe SDK session arguments

Type

dict

valid_methods

List of supported request methods

Type

list

handle_ssl_warnings()

Turn off warnings if the SSL verification argument is off.

normalize_response()

Normalize the response object to a JSON format.

Returns

A normalized JSON for the request response

Return type

json

perform_request(*method, request_arguments, expected_code=[200]*)

Execute an HTTP/HTTPS requests from given arguments and return validated response (JSON).

Parameters

- **method** (*str*) – The request method that should be used
- **request_arguments** (*dict*) – The dictionary containing the required arguments for the execution of the request
- **expected_code** (*int*) – The list containing the acceptable response codes (default is [200])

Returns

normalized_response – normalized request response in json (dictionary)

Return type

json

send_request()

Build a custom session object, prepare it with a custom request and send it.

validate_method()

Check if the input request method for the request is supported.

Raises

InvalidRequestMethod – If the input request method is not supported

validate_response()

Validate if request response is acceptable based on expected code list.

Raises

- **UnexpectedStatus** – If the response status code is not in the expected code list
- **RequestFailed** – If the HTTP/HTTPS request fails

Exception classes

InvalidRequestMethod

class `core.zowe.core_for_zowe_sdk.InvalidRequestMethod(input_method)`

Class used to represent an invalid request method exception.

3.2 z/OS Console

Contains the z/OSMF Console REST API functionalities.

3.2.1 Installing

```
pip install zowe.zos_console_for_zowe_sdk
```

3.2.2 Reference

z/OS Console classes

Console

class `zos_console.zowe.zos_console_for_zowe_sdk.Console(connection)`

Class used to represent the base z/OSMF Console API.

...

connection

Connection object

issue_command(*command*, *console=None*)

Issues a command on z/OS Console.

Parameters

- **command** – The z/OS command to be executed
- **console** – The console that should be used to execute the command (default is None)

Returns

A JSON containing the response from the console command

Return type

json

3.3 z/OS Files

Contains the z/OSMF Files REST API functionalities.

3.3.1 Installing

```
pip install zowe.zos_files_for_zowe_sdk
```

3.3.2 Reference

z/OS Files classes

Files

class `zos_files.zowe.zos_files_for_zowe_sdk.Files(connection)`

Class used to represent the base z/OSMF Files API.

...

connection

connection object

download_dsn(*dataset_name*, *output_file*)

Retrieve the contents of a dataset and saves it to a given file.

get_dsn_content(*dataset_name*)

Retrieve the contents of a given dataset.

Returns

A JSON with the contents of a given dataset

Return type

json

list_dsn(*name_pattern*)

Retrieve a list of datasets based on a given pattern.

Returns

A JSON with a list of dataset names matching the given pattern

Return type

json

list_dsn_members(*dataset_name*)

Retrieve the list of members on a given PDS/PDSE.

Returns

A JSON with a list of members from a given PDS/PDSE

Return type

json

upload_file_to_dsn(*input_file*, *dataset_name*)

Upload contents of a given file and uploads it to a dataset.

write_to_dsn(*dataset_name, data*)

Write content to an existing dataset.

Returns

A JSON containing the result of the operation

Return type

json

3.4 z/OS Jobs

Contains the z/OSMF Jobs REST API functionalities.

3.4.1 Installing

```
pip install zowe.zos_jobs_for_zowe_sdk
```

3.4.2 Reference

z/OS Jobs classes

Jobs

class `zos_jobs.zowe.zos_jobs_for_zowe_sdk.Jobs`(*connection*)

Class used to represent the base z/OSMF Jobs API.

connection

Connection object

get_job_status(*jobname, jobid*)

Retrieve the status of a given job on JES.

Parameters

- **jobname** (*str*) – The name of the job
- **jobid** (*str*) – The job id on JES

Returns

A JSON object containing the status of the job on JES

Return type

response_json

list_jobs(*owner=None, prefix='*', max_jobs=1000, user_correlator=None*)

Retrieve list of jobs on JES based on the provided arguments.

Parameters

- **owner** (*str, optional*) – The job owner (default is None)
- **prefix** (*str, optional*) – The job name prefix (default is *)
- **max_jobs** (*int, optional*) – The maximum number of jobs in the output (default is 1000)

- **user_correlator** (*str*, *optional*) – The z/OSMF user correlator attribute (default is None)

Returns

A JSON containing a list of jobs on JES queue based on the given parameters

Return type

json

submit_from_local_file(*jcl_path*)

Submit a job from local file.

This function will internally call the *submit_plaintext* function in order to submit the contents of the given input file

Parameters

jcl_path (*str*) – Path to the local file where the JCL is located

Raises

FileNotFoundError – If the local file provided is not found

Returns

A JSON containing the result of the request execution

Return type

json

submit_from_mainframe(*jcl_path*)

Submit a job from a given dataset.

Parameters

jcl_path (*str*) – The dataset where the JCL is located

Returns

A JSON containing the result of the request execution

Return type

json

submit_plaintext(*jcl*)

Submit a job from plain text input.

Parameters

jcl (*str*) – The plain text JCL to be submitted

Returns

A JSON containing the result of the request execution

Return type

json

3.5 z/OS TSO

Contains the z/OSMF TSO REST API functionalities.

3.5.1 Installing

```
pip install zowe.zos_tso_for_zowe_sdk
```

3.5.2 Reference

z/OS TSO classes

Tso

class `zos_tso.zowe.zos_tso_for_zowe_sdk.Tso(connection)`

Class used to represent the base z/OSMF TSO API.

...

connection

Connection object

session_not_found

Constant for the session not found tso message id

end_tso_session(session_key)

Terminates an existing TSO session.

Parameters

session_key (*str*) – The session key of an existing TSO session

Returns

A string informing if the session was terminated successfully or not

Return type

str

issue_command(command)

Issues a TSO command.

This function will first initiate a TSO session, retrieve the session key, send the command and finally terminate the session

Parameters

command (*str*) – TSO command to be executed

Returns

A list containing the output from the TSO command

Return type

list

parse_message_ids(response_json)

Parse TSO response and retrieve only the message ids.

Parameters

response_json (*dict*) – The JSON containing the TSO response

Returns

A list containing the TSO response message ids

Return type

list

ping_tso_session(*session_key*)

Ping an existing TSO session and returns if it is still available.

Parameters

session_key (*str*) – The session key of an existing TSO session

Returns

A string informing if the ping was successful or not. Where the options are: 'Ping successful' or 'Ping failed'

Return type

str

retrieve_tso_messages(*response_json*)

Parse the TSO response and retrieve all messages.

Parameters

response_json (*dict*) – The JSON containing the TSO response

Returns

A list containing the TSO response messages

Return type

list

send_tso_message(*session_key, message*)

Send a command to an existing TSO session.

Parameters

- **session_key** (*str*) – The session key of an existing TSO session
- **message** (*str*) – The message/command to be sent to the TSO session

Returns

A non-normalized list from TSO containing the result from the command

Return type

list

start_tso_session(*proc='IZUFPROC', chset='697', cpage='1047', rows='204', cols='160', rsize='4096', acct='DEFAULT'*)

Start a TSO session.

Parameters

- **proc** (*str, optional*) – Proc parameter for the TSO session (default is "IZUFPROC")
- **chset** (*str, optional*) – Chset parameter for the TSO session (default is "697")
- **cpage** (*str, optional*) – Cpage parameter for the TSO session (default is "1047")
- **rows** (*str, optional*) – Rows parameter for the TSO session (default is "204")
- **cols** (*str, optional*) – Cols parameter for the TSO session (default is "160")

- **rsiz**(*str*, *optional*) – Rsize parameter for the TSO session (default is “4096”)
- **str** (*acctL*) – Acct parameter for the TSO session (default is “DEFAULT”)
- **optional** – Acct parameter for the TSO session (default is “DEFAULT”)

Returns

The ‘servletKey’ key for the created session (if successful)

Return type

str

3.6 z/OSMF

Contains the z/OSMF REST API functionalities.

3.6.1 Installing

```
pip install zowe.zosmf_for_zowe_sdk
```

3.6.2 Reference

z/OSMF classes

Zosmf

class zosmf.zowe.zosmf_for_zowe_sdk.**Zosmf**(*connection*)

Class used to represent the base z/OSMF API.

connection

Connection object

get_info()

Return a JSON response from the GET request to z/OSMF info endpoint.

Returns

A JSON containing the z/OSMF Info REST API data

Return type

json

CHAPTER

FOUR

CLASSES

CONTRIBUTING

This document contains the contribution guidelines for the Zowe Client Python SDK.

Notice that the Python SDK is still in early development stages, meaning that major architectural changes might be made by the development team at any given moment. **For this reason, only bug fixes and documentation changes are being accepted as contribution at this moment** . This document will be updated as soon as the stable release of the SDK is published (v1.0.0)

- *Git branching model*
- *Pull requests*
- *Code standards*

5.1 Git branching model

This project follows the [Git flow](#) branching model.

5.2 Pull requests

Consider the following when you interact with pull requests:

- Pull request reviewers should be assigned to a same-team member.
- Pull requests should remain open for at least 24 hours, or until close of the business next business day (accounting for weekends and holidays).
- Anyone can comment on a pull request to request delay on merging or to get questions answered.

5.3 Code standards

This project follows the [PEP 8](#) style guide.

INDEX

A

`ApiConnection` (class in `core.zowe.core_for_zowe_sdk`),
8

C

`connection` (`core.zowe.core_for_zowe_sdk.SdkApi` attribute), 7

`connection` (`zos_console.zowe.zos_console_for_zowe_sdk.Console` attribute), 10

`connection` (`zos_files.zowe.zos_files_for_zowe_sdk.Files` attribute), 11

`connection` (`zos_jobs.zowe.zos_jobs_for_zowe_sdk.Jobs` attribute), 12

`connection` (`zos_tso.zowe.zos_tso_for_zowe_sdk.Tso` attribute), 14

`connection` (`zosmf.zowe.zosmf_for_zowe_sdk.Zosmf` attribute), 16

`Console` (class in `zos_console.zowe.zos_console_for_zowe_sdk`),
10

`create_custom_request_arguments()`
(`core.zowe.core_for_zowe_sdk.SdkApi`
method), 7

D

`default_url` (`core.zowe.core_for_zowe_sdk.SdkApi` attribute), 7

`download_dsn()` (`zos_files.zowe.zos_files_for_zowe_sdk.Files` method), 11

E

`end_tso_session()` (`zos_tso.zowe.zos_tso_for_zowe_sdk.Tso` method), 14

F

`Files` (class in `zos_files.zowe.zos_files_for_zowe_sdk`),
11

G

`get_dsn_content()` (`zos_files.zowe.zos_files_for_zowe_sdk.Files` method), 11

`get_info()` (`zosmf.zowe.zosmf_for_zowe_sdk.Zosmf` method), 16

`get_job_status()` (`zos_jobs.zowe.zos_jobs_for_zowe_sdk.Jobs` method), 12

H

`handle_ssl_warnings()`
(`core.zowe.core_for_zowe_sdk.RequestHandler` method), 9

`host_url` (`core.zowe.core_for_zowe_sdk.ApiConnection` attribute), 8

I

`InvalidRequestMethod` (class in
`core.zowe.core_for_zowe_sdk`), 10

`issue_command()` (`zos_console.zowe.zos_console_for_zowe_sdk.Console` method), 10

`issue_command()` (`zos_tso.zowe.zos_tso_for_zowe_sdk.Tso` method), 14

J

`Jobs` (class in `zos_jobs.zowe.zos_jobs_for_zowe_sdk`), 12

L

`list_dsn()` (`zos_files.zowe.zos_files_for_zowe_sdk.Files` method), 11

`list_dsn_members()` (`zos_files.zowe.zos_files_for_zowe_sdk.Files` method), 11

`list_jobs()` (`zos_jobs.zowe.zos_jobs_for_zowe_sdk.Jobs` method), 12

`load()` (`core.zowe.core_for_zowe_sdk.ZosmfProfile` method), 8

N

`normalize_response()`
(`core.zowe.core_for_zowe_sdk.RequestHandler` method), 9

P

`parse_message_ids()`
(`zos_tso.zowe.zos_tso_for_zowe_sdk.Tso` method), 14

`password` (`core.zowe.core_for_zowe_sdk.ApiConnection` attribute), 8

`perform_request()` (*core.zowe.core_for_zowe_sdk.RequestHandler*
method), 9

`ping_tso_session()` (*zos_tso.zowe.zos_tso_for_zowe_sdk.Tso*
method), 15

`profile_name` (*core.zowe.core_for_zowe_sdk.ZosmfProfile*
attribute), 8

`profiles_dir` (*core.zowe.core_for_zowe_sdk.ZosmfProfile*
property), 8

R

`RequestHandler` (class in *core.zowe.core_for_zowe_sdk*), 9

`retrieve_tso_messages()` (*zos_tso.zowe.zos_tso_for_zowe_sdk.Tso*
method), 15

S

`SdkApi` (class in *core.zowe.core_for_zowe_sdk*), 7

`send_request()` (*core.zowe.core_for_zowe_sdk.RequestHandler*
method), 9

`send_tso_message()` (*zos_tso.zowe.zos_tso_for_zowe_sdk.Tso*
method), 15

`session_arguments` (*core.zowe.core_for_zowe_sdk.RequestHandler*
attribute), 9

`session_not_found` (*zos_tso.zowe.zos_tso_for_zowe_sdk.Tso*
attribute), 14

`ssl_verification` (*core.zowe.core_for_zowe_sdk.ApiConnection*
attribute), 8

`start_tso_session()` (*zos_tso.zowe.zos_tso_for_zowe_sdk.Tso*
method), 15

`submit_from_local_file()` (*zos_jobs.zowe.zos_jobs_for_zowe_sdk.Jobs*
method), 13

`submit_from_mainframe()` (*zos_jobs.zowe.zos_jobs_for_zowe_sdk.Jobs*
method), 13

`submit_plaintext()` (*zos_jobs.zowe.zos_jobs_for_zowe_sdk.Jobs*
method), 13

T

`Tso` (class in *zos_tso.zowe.zos_tso_for_zowe_sdk*), 14

U

`upload_file_to_dsn()` (*zos_files.zowe.zos_files_for_zowe_sdk.Files*
method), 11

`user` (*core.zowe.core_for_zowe_sdk.ApiConnection* *at-*
tribute), 8

V

`valid_methods` (*core.zowe.core_for_zowe_sdk.RequestHandler*
attribute), 9

`validate_method()` (*core.zowe.core_for_zowe_sdk.RequestHandler*
method), 9

`validate_response()` (*core.zowe.core_for_zowe_sdk.RequestHandler*
method), 9

W

`write_to_dsn()` (*zos_files.zowe.zos_files_for_zowe_sdk.Files*
method), 11

Z

`Zosmf` (class in *zosmf.zowe.zosmf_for_zowe_sdk*), 16

`ZosmfProfile` (class in *core.zowe.core_for_zowe_sdk*), 8