SONY

AITRIOS

Cloud SDK
Console Access Library
Python
Tutorial

Version 0.2.0

2023 - 1 - 30

TOC

1. Objective

To explain the various APIs exposed by Console Access Library and help the user to build and use the library.

2. Prerequisites

- Python 3.8.10
- Python dependency modules
  - requests>=2.28.1
  - jsonschema>=4.6.2
  - setuptools>=63.2.0
  - wheel>=0.37.1
  - PyYAML>=6.0
  - marshmallow>=3.17.0
  - pytest>=7.1.2
  - pytest-html-reporter>=0.2.9
  - pytest-ordering>=0.6
  - sphinx==4.3.2
  - myst-parser>=0.18.0
  - sphinx-rtd-theme>=1.0.0
  - sphinxcontrib-napoleon>=0.7
  - rst2pdf>=0.99
  - streamlit==1.10.0
  - pytest-timeout>=2.1.0
  - PyJWT>=2.6.0
  - nassl>=4.0
  - cryptography>=36.0
  - validators>=0.18

Console Access Library API Specification

- docs/api-reference/index.html

3. Usage

Installation

1. Clone the repository
2. Install Python dependencies for demo application

   ```
   pip install -r requirements.txt
   ```

3. Run the following command to install into current Python environment

   ```
   python -m pip install lib/python_client.
   ```

4. Run the following command to install into current Python environment

   ```
   python -m pip install src/.
   ```

5. Network proxy setting

   To use the Console Access Library in a proxy environment, set the `https_proxy` environment variable

   ```
   export https_proxy=http://username:password@proxyhost:port
   ```

Set configuration parameter to start Console Access Library sample application

Option 1: Using console access setting file

- Step 1: Create console access setting parameters with real values

  samples/console_access_settings.yaml

  ```
      console_access_settings:
          console_endpoint: "__console_endpoint__"
          portal_authorization_endpoint: "__portal_authorization_endpoint__"
          client_secret: "__client_secret__"
          client_id: "__client_id__"
  ```

- Step 2: Import modules `ReadConsoleAccessSettings`, `Config` and `Client` into Console Access Library sample application

  samples/console_access_client_api_demo.py

  ```
      from console_access_library.common.read_console_access_settings import
  ReadConsoleAccessSettings
      from console_access_library.common.config import Config
      from console_access_library.client import Client
  ```

- Step 3: Set path for console access setting file, and then instantiate Console Access Library `ReadConsoleAccessSettings`
  To create `console_access_settings.yaml file`, please refer step 1

  samples/console_access_client_api_demo.py

  ```
      SETTING_FILE_PATH = os.path.join(os.getcwd(), "samples",
  "console_access_settings.yaml")
      read_console_access_settings_obj =
  ReadConsoleAccessSettings(SETTING_FILE_PATH)
  ```

- Step 4: Instantiate Console Access Library `Config`

  samples/console_access_client_api_demo.py

  ```
      config_obj = 
  Config(console_endpoint=read_console_access_settings_obj.console_endpoint,

  portal_authorization_endpoint=read_console_access_settings_obj.portal_authoriz
  ation_endpoint,
                          client_id=read_console_access_settings_obj.client_id,

  client_secret=read_console_access_settings_obj.client_secret)
  ```

Option 2: Exporting the real values to environment without creating a console access setting file

- Step 1: Export the real values to environment

  ```
  export CONSOLE_ENDPOINT="__console_endpoint__"
  export PORTAL_AUTHORIZATION_ENDPOINT="__portal_authorization_endpoint__"
  export CLIENT_SECRET="__client_secret__"
  export CLIENT_ID="__client_id__"
  ```

- Step 2: Import modules `Config` and `Client` into Console Access Library sample application

  samples/console_access_client_api_demo.py

  ```
  from console_access_library.common.config import Config
  from console_access_library.client import Client
  ```

- Step 3: Instantiate Console Access Library `Config` with real values

  samples/console_access_client_api_demo.py

  ```
  config_obj = Config(console_endpoint=None,
                      portal_authorization_endpoint=None,
                      client_id=None,
                      client_secret=None)
  ```

Option 3: Passing real values to Console Access Library Config

- Step 1: Import modules `Config` and `Client` into Console Access Library sample application

  samples/console_access_client_api_demo.py

  ```
      from console_access_library.common.config import Config
      from console_access_library.client import Client
  ```

- Step 2: Instantiate Console Access Library `Config` with real values

  samples/console_access_client_api_demo.py

  ```
      config_obj = Config(console_endpoint="__console_endpoint__",

  portal_authorization_endpoint="__portal_authorization_endpoint__",
                          client_id="__client_id__",
                          client_secret="__client_secret__")
  ```

samples/demo_config.yaml

```
    demo_configuration:
        device_id: "__device_id__"
        number_of_images: __number_of_images__
        skip: __skip__
        sub_directory_name: "__sub_directory_name__"
        number_of_inference_results: __number_of_inference_results__
        filter: "__filter__"
        raw: __raw__
        time: "__time__"
        converted: "__converted__"
        vendor_name: "__vendor_name__"
        get_images_order_by: "__get_images_order_by__"
        get_last_inference_and_image_data_order_by:
"__get_last_inference_and_image_data_order_by__"
        key: "__key__"
        app_name: "__app_name__"
        version_number: "__version_number__"
        model: "__model__
        model_id: "__model_id__"
        compiled_flg: "__compiled_flg__"
        file_name: "__file_name__"
        comment: "__comment__"
        input_format_param: "__input_format_param__"
        network_config: "__network_config__"
        network_type: "__network_type__"
        labels: "__labels__"
        config_id: "__config_id__"
        sensor_loader_version_number: "__sensor_loader_version_number__"
        sensor_version_number: "__sensor_version_number__"
        model_version_number: "__model_version_number__"
        ap_fw_version_number: "__ap_fw_version_number__"
        device_ids: "__device_ids__"
        deploy_parameter: "__deploy_parameter__"
        replace_model_id: "__replace_model_id__"
        timeout: __timeout__
```

To run API import_device_app

Place the contents of aot file as base64 in tests/unit/device_app_file_content.txt

Starting the App

To run the CLI demo, open cmd terminal from the root folder, and run the following command.

```
python samples/console_access_client_api_demo.py
```