

Getting Started with the Kosha SDK in Python

Kosha provides you with a SDK in a variety of languages to get your connectors integrated into your code and processes with minimal toil. Each connector has its own SDK as a collection of client libraries for the connector API. Using this SDK will allow for faster development and cleaner code.

What you'll learn

[x] How to interact with the Kosha user interface to obtain the client library [x] How to add the Kosha client library to your project [x] How to programatically connect to a Kosha endpoint using the client SDK

Installation

Download the SDK

1. Navigate to **My Connectors** in the Connectors tab and click into your connector of choice.
2. Click into Usage & SDK
3. You should see this screen with a variety of language options presented.
4. Click into your language of choice and select the option to download its SDK.

Install Into Project

1. Unzip the SDK into your project

```
tar -xvf <generated_client>.zip -C <your_project_directory>
```

NOTE: There is no need to modify code in the client library.

2. Install into pip via the Setuptools based setup.py script

For a specific user:

```
python setup.py install --user
```

For all users:

```
sudo python setup.py install
```

Import and Configure

```
import time
import openapi_client
from pprint import pprint

# Import any clients needed from openapi_client.api
# Import any models needed

configuration = openapi_client.Configuration(
    host = "https://<your_connector_name>.<your_company>.dev.kosha.app/<your_endpoint_path>"
)
```

```
# Enter a context with an instance of the API client
with openapi_client.ApiClient(configuration) as api_client:
    # Create an instance of of the API class
    api_client_instance = <imported_client>.<client_object>(api_client)

    try:
        api_response = api_client_instance.<endpoint>()
        pprint(api_response)
    except openapi_client.ApiException as e:
        print("Exception when calling <client> <endpoint>: %s\n" % e)
```

The world is your oyster!

Now that you've implemented one endpoint for one connector, you can implement any other endpoint in any other connector within Kosha. The process is universal across connectors and allows for clean, consistent code across your applications.