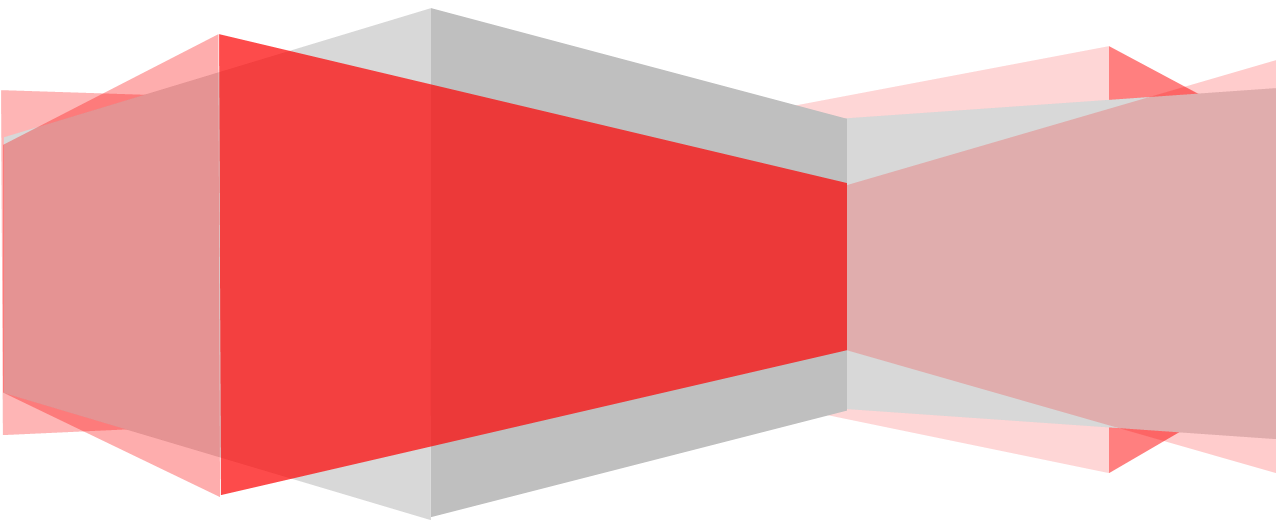


**Using the Custom Policy Template**

****

# Requirements

1. Python. The policy creation script has been tested with versions 2.7.13 and 3.6.2. Therefore, any up-to-date version of Python 2 or 3 is expected to work.
2. The SDK. The basic SDK requires minimally 3 Java archives – namely: - oracle.apiplatform.policies.sdk.jar, json-2016212.jar, groovy-all.jar. Each of these JARs can be found in various locations within the Gateway installation.
3. Eclipse IDE is recommended.
4. The PolicyTemplate.zip archive
5. Either a Microsoft Windows or any Unix / Linux environment

# Getting started

1. Ensure that you have installed either Python 2 or 3 and that the python executable is in your path
2. Extract the PolicyTemplate.zip archive to a location of your choice
3. Create a directory in a location of your choice to contain the SDK and populate it with the required JARs as indicated in point #2 of Requirements

# Usage

1. Decide where you want your Custom Policy project to reside. This should be a directory that currently does not exist
2. In a command window / shell, change directory to wherever you extracted the PolicyTemplate
3. Execute PolicyTemplate.py as described below

# Executing PolicyTemplate.py

The PolicyTemplate Python script usage is as follows:-

python PolicyTemplate.py –t targetDirectory –p policyName –s sdkDirectory [-S|--style] [-h|--help]

targetDirectory must be a non-existent directory on your filesystem where your Custom Policy project will be generated

policyName is any string of letters of your choosing that will be prepended to certain Java class names and added into various other supporting files within the project. This string will also be used to name certain generated directories. Therefore, it is important that this string is going to be valid in both the Java class naming and folder creation contexts.

sdkDirectory is the location of the SDK. It is important that you provide an absolute path to this directory

-S|--style is an optional feature that changes the case of the policyName such that in the Java context it will become uppercase but in the context of directory naming, it will be lowercase

-h|--help prints the same information that’s documented here

# Working with the Custom Policy

Let’s consider a practical example.

1. You have created a directory /home/yourname/sdk and you have populated it with the required JARs
2. You have extracted the PolicyTemplate.zip to /home/yourname/PolicyTemplate
3. You want your new policy to be located in /home/yourname/MyCustomPolicy

Here is the sequence of Linux commands that you would execute:-

cd /home/yourname/PolicyTemplate

python PolicyTemplate.py –t /home/yourname/MyCustomPolicy –p CheckBody –s /home/yourname/sdk

The generated policy is designed to work with Eclipse’s project structure and can therefore be imported directly into that IDE as follows:-

Start Eclipse

File->Import

Highlight “Existing Projects into Workspace” -> Next

Enable the Select root directory button if it isn’t already

Click Browse then navigate to (in this example) /home/yourname/MyCustomPolicy

Finish

You now have your new Custom Policy imported into Eclipse and you’re ready to start coding.