

Portfolio 2016 RESTful API + API Token Authentication + Python

PRELIMINARIES

- ❑ HTTP (or HTTPS) access to a Portfolio 2016 server that is licensed for the Portfolio API.
 - ★ e.g. `http://portfolio.example.com:8090` or `https://portfolio.example.com:9443`
 - HTTP^S is strongly recommended if connecting via API Token to a Portfolio 2016 server over the Internet.
- ❑ An API Token (*alphanumeric sequence of characters that begins with **TOKEN-***).
 - Within the Portfolio Admin Web interface , select “Users” > “Add new API Token”.
- ❑ API Token has been granted a minimum of Reader level access to at least one Portfolio Catalog , Catalog contains multiple assets , and assets have been tagged with Keywords.
- ❑ Python 2.7.x running on an OSX , Windows , or Linux workstation.
- ❑ Access to a programmer’s text editor such a TextWrangler for OSX , Notepad++ for Windows , or Vim for Linux — but not Emacs (*just kidding !*).

CODE OVERVIEW

- ★ The included example code has been written with simplicity in mind so that only a minimum number of Python modules are required (“*errno*”, “*json*”, “*os*”, “*random*”, and “*urllib2*”).
- ★ Variable declarations are configured near the top portion of the example code (e.g. `SERVER_PROTOCOL` , `SERVER_ADDRESS` , `SERVER_PORT` , `API_TOKEN` , etc.).
- ★ Once the variable declarations have been properly set , simply issue the following command (via OSX Terminal , Windows Command Prompt , Linux XTerm , etc.) to execute the script :
`python portfolio_rest_token.py`

Functions : portfolio_rest_token.py	
Function	Description
getCatalogs	Retrieves an array of the available Catalogs.
findNumberAssets	Retrieves the current total number of available assets.
findRandomID	Makes use of the Portfolio API's advanced paging capabilities by changing the items per page to "1" and then returning a random page containing a single item. Note that this method is used because — even though Portfolio automatically assigns an "Item ID" during the cataloging process — there are often breaks in the Item ID sequence due to item deletion.
getAsset	Returns an array containing both the Filename and Keywords for the random Item ID.
saveRandomPreview	Saves the Item's JPEG "Preview". Note that during the cataloging process , Portfolio automatically generates both a "Thumbnail" (<i>256 pixels</i>) and an intermediary "Preview" (<i>default of 1600 pixels</i>).
saveRandomMetadata	Saves the Keyword metadata as a CSV text file.
Logout	Session logout. Note that this is not normally needed as there is no connection limit nor timeout for RESTful API connections.