EDC AUTOMATIONS

ABOUT THE PROJECT

To start a new project of Data Governance, we developed a (Python) script to automatise and make some actions faster:

- Create new EDC connections.
- Create new EDC resources.

BUILT WITH

- Pyhotn 3.x

PREREQUISITES

PYTHON LIBRARIES

At first, you need to install the libraries:

- Pyexcel - Provides one application programming interface to read, manipulate and write data in various Excel formats.
- Requests - Requests allows you to send HTTP/1.1 requests extremely easily.

This is an example of the command:

```
$ pip install pyexcel pyexcel-xls pyexcel-xlsx requests
```

DOWNLOAD PROBLEM

If you obtain this warnign:

```
WARNING: Retrying (Retry(total=4, connect=None, read=None, redirect=None,
status=None)) after connection broken by
'ConnectTimeoutError(<pip._vendor.urllib3.connection.HTTPSConnection object at
0x7f9a60282a90>, 'Connection to pypi.org timed out. (connect timeout=15)')':
/simple/requests/
```

You need to export the env variable for the proxy:

```
$ export https_proxy=http://inet.syssede.systest.sanpaoloimi.com:9090
```

XLRD VERSION

In the earlier version xlrd do not support .xlsm files.

To fix the problem, you need to install the version 1.2.0:

```
$ pip uninstall xlrd
$ pip install xlrd==1.2.0
```

ENV CONFIGURATIONS

The bank prepared an environment where we can install the needed libraries:

```
$ cd /opt/python/informatica/
$ source bin/activate
(informatica) [...]$
```

INSTALLATION

To install the script, copy and unzip the archive file in the correct folder:

```
$ unzip Automation.zip
```

GENERAL USAGE usage

```
usage:
<python> main.py
                -t, --tech <db2|hive|mongo|oracle|sqlsrv|teradata>
                -x, --xls <excel_file>
                [-c, --connections]
                [-r, --resources]
```

Use source Excel file to create JSON files and connections for Informatica service.

```
optional arguments:
  -h, --help            show this help message and exit
  -v, --version         show program version
  -t, --tech <technology> technology to use to create the resource
  -x, --xls <excel_file>  Excel file (source)
  -r, --resources       create resources into the Informatica service
  -c, --connections     create connections into the Informatica service
```

CREATE CONNECTIONS

```
<python> main.py --tech <db2|hive|mongo|oracle|sqlsrv|teradata> --xls <excel_file>
-c
```

The params you need are:

- `-x, --xls <excel_file>`, specify the Excel file to read.
- `-t, --tech <db2|hive|mongo|oracle|sqlsrv|teradata>`, define the technology you want to use.
- `-c, --connections`, say to the script to create connections.

Example for connections:

```
$ python main.py -t db2 -x xlsxFile/Test_Template_DB2.xlsx -c
```

The output files are saving in the result folder (see the Configuration file). This folder will also contain the ODBC file with the configurations you need to append into the `$INFA_HOME/ODBC7.1/odbc.ini` file.

CREATE RESOURCES

```
<python> main.py --tech <db2|hive|mongo|oracle|sqlsrv|teradata> --xls <excel_file>
-r
```

The params you need are:

- `-x, --xls <excel_file>`, specify the Excel file to read.
- `-t, --tech <db2|hive|mongo|oracle|sqlsrv|teradata>`, define the technology you want to use.
- `-r, --resources`, say to the script to create resources by API requests.

The script calls the Rest API to create a new resource. If it fails, it creates a JSON file in the results folder.

Example for resources:

```
$ python main.py -t db2 -x xlsxFile/Test_Template_DB2.xlsx -r
```

CONFIGURATION FILES

FILE 'globalparams.py'

The `globalparams.py` file contains the values for the specific environments.

For this reason, it is a best practice to **back up this file before installing a new version** of the script.

FILE 'config.py'

In the main folder, it is available a file called `config.py`. In this file, you can change some configurations:

- `infaHome`, the same value of the environment variable $INFA_HOME.
- `resultFolder`, where the script save the results (e.g. JSON files, ODBC files, etc.).
- `sheetName`, the name to assign to the Excel sheet to elaborate.

CONTACT

Lorenzo Lombardi - llombardi@informatica.com