Salesforce API with Simple Salesforce For Python

Python has a plethora of modules that makes programming fun and easy. If you need to use Salesforce API with Python, the simple-salesforce module is your best friend. The module takes care of boring stuff like authentication and let you use different APIs. You can check the documentation and source code here (https://github.com/simple-salesforce/simple-salesforce). If you want to know what methods are available for the module, take a look at the api.py file (https://github.com/simple-salesforce/simple-salesforce/blob/master/simple\_salesforce/api.py).

**How it works**

All you need to do is to create a Salesforce object with a user name, password, security token and the version of API you are using. If the instance is not production, you need to add sandbox=True. By using this object, you can call different APIs (Rest API, APEX Rest, Bulk API and SOQL query) with specialised methods with query url or SOQL query in their arguments. In summary, calling Salesforce API only requires 2 steps.

**API steps**

# Create Salesforce Object

sf = Salesforce(username=<uname>, password=<pw>,\

security\_token=<token>, version='41.0', sandbox=True)

# (1) Make a direct API call to list avaialble REST API Versions

result = sf.restful('limits/', params=None)

print(result)

# (2) Make a SOQL query to get the record count of lead object

result = sf.query('Select count() From Lead')

print('No of records: {}'.format(result['totalSize']))

# (3) Make an HTTP rquest to an APEX Rest Endpoint

result = sf.apexecute('Account/accountId')

print(result)

**Simple Salesforce Application**

Now that we understand how awesome Simple Salesforce is, I have a small application to read metadata from a Salesforce object and create a Create Table statement for Postgres database. It takes an object name as an argument (not case sensitive). You can call the script as ‘python script.py account’ to generate a text file that contains a create table statement for the account object and csv file containing its metadata.

Here comes the code.

Enjoy!

**Code**