How To Customise Python Module

One of the greatest things in Python is that it has heaps of cool modules to use. They make programming fun and easy. [Here](https://www.mydatahack.com/salesforce-api-with-simple-salesforce-for-python/) is the example of an amazing salesforce API module.

What makes it even better is that these python modules are easy to customise for your need by editing the source code and recompiling it (if necessary). In this post, I will show you how to customise python modules by using the example of the [sharepy](https://pypi.python.org/pypi/sharepy/1.1.3) module.

Sharepy handles authentication for SharePoint Online/Office365 site and allows you to make API calls. For the further details of the example of sharepoint data ingestion, have a look [here](https://www.mydatahack.com/how-to-get-data-from-sharepoint-with-python/).

Sharepy authentication works like this:

[cc lang="python" tab\_size="4" lines="-1" theme="GeSHi" line\_numbers="false"]

import sharepy

s = sharepy.connect("example.sharepoint.com")

[/cc]

For initialisation, the connect method will ask you to input user name and password in the console. To deploy the code in the server, I need to include user name and password when we initialise without typing it in the console.

We want to get sharepy to initialise with user name and password parameters as below:

[cc lang="python" tab\_size="4" lines="-1" theme="GeSHi" line\_numbers="false"]

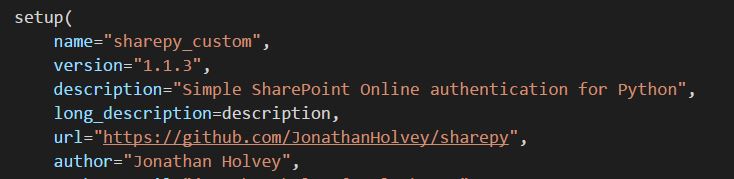
import sharepy\_custom as sharepy

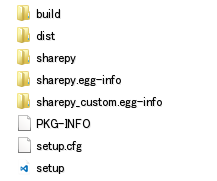
s = sharepy.connect("example.sharepoint.com", <username>, <password>)

[/cc]

Sharepy is well documented and you can check the source code [here](https://github.com/JonathanHolvey/sharepy).

Steps

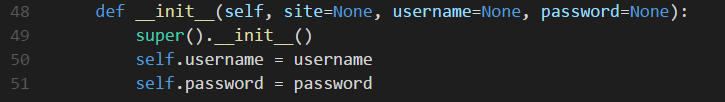
1. Download the sharepy module as a zip file from [here](https://pypi.python.org/pypi/sharepy/1.1.3).
2. Unzip the file and edit name to “sharepy\_custom” in setup.py. [This thread](https://stackoverflow.com/questions/1471994/what-is-setup-py) explains setup.py nicely.
3. Open session.py in the sharepy folder.



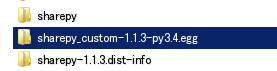
1. In line 17 and 18, add username and password in the method argument and class construction.



1. Add username and password in the class constructor as below. Delete input in line 56 and 72.



1. Go to the sharepy folder and run the command, `python setup.py install`. This will create a folder in your python site-packages library folder. In the screenshot, sharepy and sharepy-1.1.3.dist-info were created py pip install sharepy command. They contain the native module. Running setup.py install created the folder sharepy\_custom-1.1.3-py3.4.egg, which has compiled modules with the custom code.



1. Go into the the folder sharepy\_custom-1.1.3-py3.4.egg and change the sharepy folder name to sharepy\_custom. This name will be used to import the module.



Now we can import the customised module and use it as we planned!

[cc lang="python" tab\_size="4" lines="-1" theme="GeSHi" line\_numbers="false"]

import sharepy\_custom as sharepy

s = sharepy.connect("example.sharepoint.com", <username>, <password>)

[/cc]

Go check out how this customised sharepy is used to [ingest data from Sharepoint](https://www.mydatahack.com/how-to-get-data-from-sharepoint-with-python/)!