
pyfatsecret Documentation

Release 1

walexnelson, trtg

October 03, 2014

1	pyfatsecret setup	3
2	pyfatsecret usage	5
3	Indices and tables	7

This library provides a lightweight python wrapper for the Fatsecret API with the goal of making it easier to visualize the data retrieved from the API. To that end, this library will usually return lists of identical elements for ease of plotting, discarding extra header fields that the Fatsecret API otherwise includes. All API calls return a either a single or list of JSON dictionaries.

Fatsecret supports both delegated and undelegated calls. Only through delegated calls can you access Fatsecret user profile data.

Contents:

pyfatsecret setup

pyfatsecret usage

Undelegated API Calls

```
from fatsecret import Fatsecret
fs = Fatsecret(consumer_key, consumer_secret)
foods = fs.foods_search("Tacos")
```

Delegated API Calls With Callback URL

If you provide a callback URL then Fatsecret will return the Verifier PIN in the request object.

```
from fatsecret import Fatsecret
fs = Fatsecret(consumer_key, consumer_secret)
auth_url = fs.get_authorized_url(callback_url='http://example.com')
```

Redirect to the auth_url in your web app for the user to authorize access. Once authorized you will automatically be redirected to the callback_url. From there you can pull the oauth_verifier from the request object if the user allowed access

```
pin = request.args.get('oauth_verifier') session_token = fs.authenticate(pin)
```

Delegated API Call Without Callback URL If your app can't support callback URLs then you will have to provide a way for the user to enter the PIN

```
from fatsecret import Fatsecret
fs = Fatsecret(consumer_key, consumer_secret)
print("Browse to the following URL in your webbrowser: {}".format(fs.get_authorize_url()))
session_token = fs.authenticate(input("Enter PIN: "))
```

Open Existing Delegated Session If you store the session_token returned by fs.authenticate() or fs.profile_get_auth() then you can open a session by handing the session_token when you create your new object

```
from fatsecret import Fatsecret
session_token = None # You will have to use your imagination for how you plan to store / retrieve these tokens
fs = Fatsecret(consumer_key, consumer_secret, session_token=session_token)
```

Indices and tables

- *genindex*
- *modindex*
- *search*