
yandex_oauth

Igor Kuptsov

июл. 24, 2022

1	Установка библиотеки	3
2	Подключение библиотеки	5
3	Функции <code>yandex_oauth</code>	7
3.1	Пакет <code>yandex_oauth</code>	7
3.1.1	Описание пакета	7
3.1.2	Модули	7
3.1.3	Модуль <code>yandex_oauth.yao</code>	7
4	Примеры	9
4.1	№1: Интерактивная функция получения токенов для приложения CLI	9
4.2	№2: Функция загрузки токенов и параметров	10
4.3	№3: Функции загрузки токена и параметра	11
5	MIT License	13
6	Предметный указатель	15
	Содержание модулей Python	17
	Алфавитный указатель	19

Спасибо, что обратились к нашей библиотеке!

yandex_oauth - библиотека функций для работы с OAuth токенами Яндекс API

Примечание: Библиотека находится в активной стадии разработки.

На данный момент реализованы функции получения токенов по коду авторизации, сохранения в и загрузку из pickle хранилища.

Установка библиотеки

Установка библиотеки производится из репозитория [PYPI](#) командой:

```
$ pip install yandex-oauth
```

Подключение библиотеки

Библиотека является пакетом *yandex_oauth* с модулем функций *yao*

```
from yandex_oauth import yao
```


3.1 Пакет `yandex_oauth`

3.1.1 Описание пакета

Библиотека Yandex OAuth

3.1.2 Модули

3.1.3 Модуль `yandex_oauth.yao`

Модуль функций библиотеки

`yandex_oauth.yao.check_expire_token(token, delta)`

Функция проверки на истечение времени жизни токена

Параметры

- `token` (*dict*) – Словарь токенов
- `delta` (*datetime.delta*) – Временной интервал, если разница между датой истечения и текущей меньше этого значения, функция вернет `True`

Результат

`True` или `False`

Тип результата

`bool`

`yandex_oauth.yao.get_token_by_code(code, client_id, client_secret)`

Функция получения токена по коду авторизации

Параметры

- `code` (*str*) – код подтверждения

- `client_id (str)` – id приложения
- `client_secret (str)` – пароль приложения

Результат

Словарь токенов

Тип результата

dict

`yandex_oauth.yao.load_token(path)`

Функция загрузки токенов из хранилища

Параметры

`path (str)` – путь к хранилищу

Результат

Словарь токенов или False, если нет хранилища

`yandex_oauth.yao.refresh_token(token)`

Функция обновления токена

Параметры

`token (dict)` – Словарь токенов

Результат

Обновленный словарь токенов

Тип результата

dict

`yandex_oauth.yao.save_token(path, token)`

Функция сохранения токенов в pickle хранилище

Параметры

- `path (str)` – путь для сохранения хранилища
- `token (dict)` – словарь токенов

Результат

True или False

Тип результата

bool

4.1 №1: Интерактивная функция получения токенов для приложения CLI

```
from . import __path__ as path
from yandex_oauth import yao

def make_config():
    """Функция создания первичного конфигурационного файла"""

    print('\n\nПЕРВОНАЧАЛЬНАЯ КОНФИГУРАЦИЯ ПРИЛОЖЕНИЯ\n')
    print('1. Авторизуйтесь на Яндексе. Укажите логин и пароль администратора_
    ↪организации,\n    от имени которого будут выполняться запросы к API\n')
    print('    https://passport.yandex.ru/')
    print('\n')
    print('2. Зарегистрируйте приложение\n')
    print('    https://oauth.yandex.ru/client/new')
    print('\n    Заполните поля:')
    print('    Название сервиса: ya360')
    print('    Для какой платформы нужно приложение: Веб-сервисы')
    print('    Какие данные вам нужны: Яндекс 360 Admin API, Яндекс 360 Directory API_
    ↪(отметить все галочки)')
    print('    Callback URI: https://oauth.yandex.ru/verification_code')
    print('\n')
    print('3. Перейдите на страницу созданных приложений и выберите ya360\n')
    print('    https://oauth.yandex.ru/')
    print('\n')
    try:
        client_id = input('    Введите ClientID: ')
    except:
        print('\n')
```

(continues on next page)

```

        exit(0)
    try:
        client_secret = input('    Введите Client secret: ')
    except:
        print('\n')
        exit(0)
    try:
        adminemail = input('    Введите e-mail администратора организации, от имени
↪ которого будут выполняться запросы к API: ')
    except:
        print('\n')
        exit(0)
    url = 'https://oauth.yandex.ru/authorize?response_type=code&client_id='+str(client_
↪ id)+'&login_hint='+str(adminemail)+'&force_confirm=yes'
    print('\n4. Перейдите по следующей ссылке и получите код подтверждения\n')
    print('        '+url)
    print('\n')
    try:
        code = input('    Введите код подтверждения: ')
    except:
        print('\n')
        exit(0)
    token = yao.get_token_by_code(code, client_id, client_secret)
    if 'error' in token and 'error_description' in token:
        print('\nОШИБКА '+token['error']+': '+token['error_description']+'\n')
        exit(1)
    print('5. Зайдите в профиль организации\n')
    print('        https://admin.yandex.ru/company-profile\n')
    try:
        orgid = input('    Введите ID организации: ')
    except:
        print('\n')
        exit(0)
    token.update({'orgid':orgid,'adminemail':adminemail})
    if yao.save_token(path[0], token):
        return True
    else:
        print('\nОшибка создания конфигурационного файла\n')
        exit(1)

```

4.2 №2: Функция загрузки токенов и параметров

```

from . import __path__ as path
from yandex_oauth import yao

def load_config():
    """Функция загрузки словаря токенов и параметров

    :returns: Словарь с токенами и параметрами или False, если нет pickle хранилища

```

(continues on next page)

(продолжение с предыдущей страницы)

```
"""  
  
return yao.load_token(path[0])
```

4.3 №3: Функции загрузки токена и параметра

В примере №1: *Интерактивная функция получения токенов для приложения CLI* мы добавили в pickle хранилище кроме токенов еще два дополнительных параметра (orgid и adminemail)

```
from . import __path__ as path  
from yandex_oauth import yao  
  
def load_token():  
    """Функция загрузки токена  
  
    :returns: токен  
    """  
  
    return yao.load_token(path[0])['access_token']  
  
def load_orgID():  
    """Функция загрузки id организации  
  
    :returns: ID организации  
    """  
  
    return yao.load_token(path[0])['orgid']
```


Copyright (c) 2022 Igor Kuptsov

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the «Software»), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED «AS IS», WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Предметный указатель

- genindex
- modindex

y

yandex_oauth, [7](#)

yandex_oauth.yao, [7](#)

C

`check_expire_token()` (в модуле `yandex_oauth.yao`), 7

G

`get_token_by_code()` (в модуле `yandex_oauth.yao`), 7

L

`load_token()` (в модуле `yandex_oauth.yao`), 8

R

`refresh_token()` (в модуле `yandex_oauth.yao`), 8

S

`save_token()` (в модуле `yandex_oauth.yao`), 8

Y

`yandex_oauth`
модуль, 7

`yandex_oauth.yao`
модуль, 7

модуль

`yandex_oauth`, 7
`yandex_oauth.yao`, 7