wechat-pay-sdk Documentation

Release 0.4

Zack Ren

Dec 21, 2017

Contents

wechat-pay-sdk is an unofficial sdk for WeChat pay. It contains almost all API of WeChat pay.

Code host at Github

CHAPTER 1

Install WeChat-Pay-SDK

Install with **pip**:

```
pip install wechat-pay-sdk
```

Or

Install from source

```
python setup.py install
```

CHAPTER 2

How To Use

Import the WeChatPay class and init it with your own parameters:

```python
from wechatpay import WeChatPay

WECHAT_APPID = 'your_app_id'
WECHAT_MCH_ID = 'your_mch_id'
WECHAT_NOTIFY_URL = 'your_notify_url'
WECHAT_PAY_SECRET = 'your_pay_secret'
WECHAT_CERT = 'path/to/your_cert.pem'
WECHAT_KEY = 'patch/to/your_key.pem'


sp = WeChatPay(WECHAT_APPID, WECHAT_MCH_ID,
                   WECHAT_NOTIFY_URL, WECHAT_PAY_SECRET, WECHAT_CERT, WECHAT_KEY)

# post order
result = sp.unifiedorder(body=body, out_trade_no=out_trade_no, total_fee=total_fee,
                             spbill_create_ip=client_ip)

# order query
result = sp.query_order(out_trade_no=out_trade_no)


# download bills
result = sp.downloadbill(bill_date=bill_date, bill_type=bill_type)

# refund (this method need your cert.pem and key.pem file which need you to download from the official website)
result = sp.refund(out_trade_no, out_refund_no, total_fee, total_fee)

# refund query
result = sp.refund_query(out_trade_no)
```

```
if result.success:
    # your code
    pass
```

CHAPTER 3

API

3.1 WeChatPay

class WeChatPay(*appid*, *mch_id*, *notify_url*, *pay_secret*, *cert=None*, *key=None*)

unifiedorder(*device_info*, *sign_type*, *fee_type*, *trade_type*, ***kwargs*)

Form a new order. more information at API

Parameters

- **device_info** – exception type
- **sign_type** – sign method(MD5 or HMAC-SHA256)
- **fee_type** – currency type
- **attach** – attach data
- **out_trade_no** – order id (should be unique in your system)
- **trade_type** – (JSAPINATIVEAPP)
- **body** – order description
- **detail** – item detail
- **total_fee** – the amount of fee
- **spbill_create_ip** – client ip address
- **time_start** – order generate time (format yyyyMMddHHmmss)
- **time_expire** – order expire time (format yyyyMMddHHmmss)
- **goods_tag** – discount info
- **product_id** – item id
- **limit_pay** – no_credit

- **openid** – user wechat openid
- **scene_info** – scenario info

Return type *UnifiedorderResult*

random_str()

Generate a random string

Return type string

verify_notify(*req*)

Verify WeChat payment notification

Parameters **req** – Request object from requests

Return type *NotifyResult* or *ErrorResult*

to_xml(*dct*)

Transfer a dict to xml string :param dct: data dict to send :rtype: string

sendbox_sign()

Generate sendbox secret key API

Return type *SendBoxKey*

query_order(***kwargs*)

Query order from WeChat API

Parameters

- **transaction_id** – wechat order id
- **out_trade_no** – order id (transaction_id and out_trade_no only need one)

Return type *OrderResult*

downloadbill(*bill_date*, *sign_type='MD5'*, *bill_type='ALL'*, *tar_type='GZIP'*)

Download bills API

Parameters

- **bill_date** – the date of bill (format yyyymmdd)
- **bill_type** – the type of bill (ALL, SUCCESS, REFUND, RECHARGE_REFUND)
- **tar_type** – tar type(GZIP)

Return type *DownLoadBillResult*

close_order(*out_trade_no*, *sign_type='MD5'*)

Close order API

Parameters **out_trade_no** – order id

Return type *CloseOrderResult*

short_url(*long_url*, *sign_type='MD5'*)

Convert long url to short url and make the Qr code more recognizable API

Parameters **long_url** – long url

Return type *ShortUrlResult*

refund(*out_trade_no*, *out_refund_no*, *total_fee*, *refund_fee*, *sign_type='MD5'*, *refund_fee_type='CNY'*, ***kwargs*)

Method for refund **this method need you to proide your cert.pem and key.pem file** API

Parameters

- **transaction_id** – wechat order id
- **out_trade_no** – order id (transaction_id and out_trade_no only need one)
- **out_refund_no** – refund id
- **total_fee** – total order fee
- **refund_fee** – refund total fee
- **refund_fee_type** – currency type (CNY)
- **refund_desc** – refund reason
- **refund_account** – refund account

Return type :py:class::*RefundResult*

refund_query(*out_refund_no*, *sign_type='MD5'*, ***kwargs*)

Query refund process API :param transaction_id: wechat order id :param out_trade_no: order id :param out_refund_no: refund id :param refund_id: refund id from WeChat (transaction_id, out_trade_no, out_refund_no, refund_id only need to one) :param offset: query offset :rtype: :py:class::*RefundQueryResult*

3.2 WeChatResult

class WeChatResult(*resp*)

resp

resp is the instance of Response class from requests

success

success is used to determine whether this request is successful

result

result is a python dict which contains the response data that parsed from xml

error_msg

error_msg contains the error message if anything went wrong

class UnifiedorderResult(*WeChatResult*)

trade_type

(JSAPINATIVEAPP)

prepay_id

generate by WeChat for more operation (valid in 2 hours)

code_url

Qr code url for payment

class NotifyResult(*WeChatResult*)

Every key in result dict refer to a attribute of this class API

class ErrorResult(*WeChatResult*)

success

in this case success is False

error_msg
error message dict

class SendBoxKey(*WeChatResult*)

key
send box secret key

class OrderResult(*WeChatResult*)

trade_state
the state of this order

out_trade_no
order id

total_fee
amount of fee of this order

class DownLoadBillResult(*WeChatResult*)

text
unicode of the bill content

content
bill content

class CloseOrderResult(*WeChatResult*)
pass

class ShortUrlResult(*WeChatResult*)

short_url
short url

class RefundResult(*WeChatResult*)
Every key in result dict refer to a attribute of this class

API

class RefundQueryResult(*WeChatResult*)
API

Because there are some field need to access by index, so use the result dict to fetch the data you need

CHAPTER 4

Indices and tables

- genindex
- search

Index