1. **基本概念说明**
   1. Wish 店铺设置
      1. Merchant Id

Merchant Id 是 店铺在wish 平台上 对应的商家Id,用以区别wish 平台上的销售商

* + 1. App Name

App Name 是集成Api调用 的项目名称或是应用名称，只供参考，不作为程序参数

* + 1. Client Id

接入wish平台与本店铺对应的client id，用于 获取授权之类的操作

* + 1. Client Secret

接入wish平台上对应店铺的密钥串，用于获取授权之类的操作

* + 1. Redirect URI

接入 wish 平台上对应店铺的返回Url,用于接收预授权码，必须是安全链接，如 https://www.eastmall.vip/wish\_return

* 1. Api调用参数说明

每个调用函数都有sessionType 参数，此参数为SessionType 枚举类型，共有两个选值：SessionType.Prod 为正式环境的Api调用，SessionType.Sandbox 为沙盒环境的Api调用

* 1. Wish返回结果集

Wish 返回的结果为json，基本的数据如下

{

"message":"Tracking provider is not one of the accepted providers on Wish",

"code":1000,

"data":2003

}

其中 data 节点 视具体的调用返回结果而定，可能是数字，也可能是其他的复杂类型。Code为0表示调用成功，非零数字代表出错，Api调用 函数的返回结果集为根据相应的返回json的结构定义的实体。

1. **授权部分**
   1. 获取预授权码Uri

在Wish平台或是Wish 沙盒平台 上获取预授权码的Uri为

[https://china-merchant.wish.com/oauth/authorize?client\_id={client\_id}](https://china-merchant.wish.com/oauth/authorize?client_id=%7bclient_id%7d) 或

[https://sandbox.merchant.wish.com/oauth/authorize?client\_id={client\_id}](https://sandbox.merchant.wish.com/oauth/authorize?client_id=%7bclient_id%7d)

这一步发送之后，客户会转向到授权页面，无论同意或拒绝，都会返回redirectUrl页面。

如果用户同意授权，页面将跳转至 redirect\_uri?code={authorization\_code}。这里的code用于换取access\_token ; 若用户禁止授权，则重定向后不会带上code参数

函数定义：

String GetAuthorizeUrl(string clientId, string redirectUrl, string grantType = "authorization\_code", SessionType sessionType = SessionType.Prod)

调用示例 ：

string proAuthUrl = Wl.Wish.Open.OAuthAPIs.OAuthApi.GetAuthorizeUrl(

clientId: "58d5c99a2bc5a20f60343036",

redirectUrl: @"https://vlan.com/WishApi/InstantShop.html",

sessionType: SessionType.Sandbox);

此处的clientId 和 redirectUrl 必须与 wish平台上相应店铺中的信息完全一致。

从wish返回的code为预授权码，有效期为5分钟，请在5分钟内换取access\_token

* 1. 获取AccessToken

在wish 平台鉴权后，会引导到店铺设置中相应设定的returnUrl并带入code作为参数，此参数为预授权码，有效期5分钟；使用预授权码获取accesstoken 的函数用下

函数定义：

OAuthAccessTokenResult GetAccessToken(string clientId, string clientSecret, string preAuthCode, string redirectUrl, SessionType sessionType = SessionType.Prod);

调用示例：

var accesstoken= Wl.Wish.Open.OAuthAPIs.OAuthApi.GetAccessToken(

clientId:client\_id,

clientSecret:client\_secret,

preAuthCode:pre\_auth\_code,

redirectUrl:redirect\_url,

sessionType:SessionType.Sandbox);

* 1. 刷新AccessToken

由于某些原因，需要更换access\_token，可以使用 refresh\_token 参数，此数据在上次获取access\_token 时已经一并获取并与access\_token 一起存入在某个数据仓库。

函数定义：

OAuthAccessTokenResult RefreshAccessToken(string clientId, string clientSecret, string refreshToken, SessionType sessionType = SessionType.Prod);

调用示例：

var newAccessToken= RefreshAccessToken(

clientId:client\_id,

clientSecret:client\_secret,

refreshToken:refresh\_token,

sessionType:SessionType.Sandbox

);

* 1. 测试 AccessToken

获取的access\_token 可以用此方法来测试是否正确

函数定义 ：

OAuthAccessTokenTestResult TestAccessToken(string accessToken, SessionType sessionType = SessionType.Prod)；

调用示例：

Var tokenTest= TestAccessToken(accessToken:access\_token,sessionType:SessionType.Sandbox);

1. **Product 商品部分**
   1. CommonApi
      1. 获取access\_token

在wish 平台鉴权后，会引导到店铺设置中相应设定的returnUrl并带入code作为参数，此参数为预授权码，有效期5分钟；使用预授权码获取accesstoken 的函数用下

函数定义：

AccessTokenResult GetAccessToken(string clientId, string clientSecret, string preAuthCode, string redirectUrl, SessionType sessionType = SessionType.Prod);

调用示例：

var accesstoken= GetAccessToken(

clientId:client\_id,

clientSecret:client\_secret,

preAuthCode:pre\_auth\_code,

redirectUrl:redirect\_url,

sessionType:SessionType.Sandbox);

* + 1. 刷新access token

由于某些原因，需要更换access\_token，可以使用 refresh\_token 参数，此数据在上次获取access\_token 时已经一并获取并与access\_token 一起存入在某个数据仓库。

函数定义：

AccessTokenResult RefreshAccessToken(string clientId, string clientSecret, string refreshToken, SessionType sessionType = SessionType.Prod);

调用示例：

var newAccessToken= RefreshAccessToken(

clientId:client\_id,

clientSecret:client\_secret,

refreshToken:refresh\_token,

sessionType:SessionType.Sandbox

);

* + 1. 创建商品

创建商品必须的参数为,其中标示为 optional 的为可选 参数

name 商品的名称

description 商品的描述

tags 商品的标签 以逗号相隔，最多允许10个标签

sku 此商品在用户平台上的唯一标识

inventory 库存数量 最大允许数量 500,000

price 销售价格 最大值 100,000

shipping 运费 最大值 1000

main\_image 商品主图url, 图片最少100\*100 像素

color 颜色optional

size 尺码 optional

msrp 厂家建议零售价 optional

shipping\_time 运输时间 optional

parent\_sku 如果本身是个规格数据，就要指定隶属于哪个商品sku , optional

brand 品牌optional

landing\_page\_url 在自己网站上的商品链接optional

upc UPC全球统一 商品编码 optional

extra\_images 其他的图片的url,以’|’相隔optional

函数定义：

ProductResult ProductCreate(string clientId,string accessToken, Wl.Wish.Entities.Request.Product product, SessionType sessionType = SessionType.Prod, int timeOut = Config.TIME\_OUT);

调用示例：

var product= ProductCreate(

accessToken:access\_token,

product:product\_para,

sessionType:SessionType.Prod);

如果调用成功，返回的商品数据里会包含wish平台分配的商品id

* + 1. 按productId来获取商品

函数定义：

ProductResult ProductRetrieveViaId(string clientId,string accessToken, string productId, SessionType sessionType = SessionType.Prod)；

调用示例：

var product= ProductRetrieveViaId(accessToken:access\_token,productId:product\_id,sessionType:SessionType.Sandbox);

返回的json 如下 ：

{'code': 0,

'data': {'Product': {'description': 'JIMMY CRACK CORN',

'id': '51e0a2c47d41a236cfffe3a0',

'name': 'TEST!!!',

'number\_saves': '12529',

'number\_sold': '14',

'review\_status': 'approved',

'tags': [{'Tag': {'id': 'jimmy', 'name': 'jimmy'}},

{'Tag': {'id': 'crack', 'name': 'Crack'}},

{'Tag': {'id': 'corn', 'name': 'Corn'}},

{'Tag': {'id': 'but', 'name': 'but'}},

{'Tag': {'id': 'i', 'name': 'I'}},

{'Tag': {'id': 'dont', 'name': 'dont'}},

{'Tag': {'id': 'care', 'name': 'Care'}}],

'variants': [{'Variant': {'enabled': 'True',

'id': '51e0a2c67d41a236cfffe3a2',

'product\_id': '51e0a2c47d41a236cfffe3a0',

'inventory': '10000',

'msrp': '20000.0',

'price': '10000.0',

'shipping': '101.0',

'sku': 'Bottle'}}]}},

'message': ''}

* + 1. 按 parentSku 来获取商品

函数定义：

ProductResult ProductRetrieveViaParentSku(string clientId,string accessToken, string parentSku, SessionType sessionType = SessionType.Prod);

调用示例：

var product= ProductRetrieveViaParentSku(

accessToken:access\_token,

parentSku:parent\_sku,

sessionType:SessionType.Sandbox);

返回数据参见上条

* + 1. 通过 product\_id 或 parent\_sku 更新 商品

商品的参数

Id wish平台上的商品id, id与parent\_sku 必须填写一个

parent\_sku wish平台上的parent\_sku id与parent\_sku 必须填写一个

name 商品名称，optional

description 商品描述，optional

tags 商品标签，以逗号相隔，最多10个，optional

brand 品牌，optional

landing\_page\_url 在自营站点上的相应的uri ，optional

upc 全球统一商品编码，optional

extra\_images 商品的附加图片url,以’|’相隔optional

函数定义：

ProductResultBase ProductUpdate(string clientId,string accessToken, Wl.Wish.Entities.Request.ProductToUpdate product, SessionType sessionType = SessionType.Prod);

调用示例

var productUpdate=ProductUpdate(

accessToken:access\_token,

product:product\_para,

sessionType:SessionType.Sandbox

);

返回最基本的json数据，code为0表示 成功

* + 1. 商品上架 (通过 product\_id)

函数定义：

ProductResultBase ProductEnableViaId(string clientId,string accessToken, string productId, SessionType sessionType = SessionType.Prod);

调用示例：

var productEnable=ProductEnableViaid(

accessToken:access\_token,

productId:product\_id,

sessionType:SessionType.Sandbox

);

返回最基本的json数据，code为0表示 成功

* + 1. 商品上架(通过 parent\_sku)

函数定义：

ProductResultBase ProductEnbaleViaParentSku(string clientId,string accessToken, string parentSku, SessionType sessionType = SessionType.Prod);

调用示例：

var productEnable=ProductEnbaleViaParentSku(

accessToken:access\_token,

parentSku:parent\_sku,

sessionType:SessionType.Sandbox

);

返回最基本的json数据，code为0表示 成功

* + 1. 商品下架 (通过 product\_id)

函数定义：

ProductResultBase ProductDisableViaId(string clientId,string accessToken, string productId, SessionType sessionType = SessionType.Prod);

调用示例：

var productDisabled=ProductDisableViaId(

accessToken:access\_token,

productId:product\_id,

sessionType:SessionType.Sandbox

);

返回最基本的json数据，code为0表示 成功

* + 1. 商品下架 (通过 parent\_sku)

函数定义：

ProductResultBase ProductDisableViaParentSku(string clientId,string accessToken, string parentSku, SessionType sessionType = SessionType.Prod);

调用示例：

var productDisable=ProductDisableViaParentSku(

accessToken:access\_token,

parentSku:parent\_sku,

sessionType:SessionType.Sandbox

);

返回最基本的json数据，code为0表示 成功

* 1. InstantApi
     1. 获取所有商品

函数定义:

ProductList ProductAllRetrieve(string clientId,string accessToken, int? posStart = null, int? numLimit = null, string sinceDate = null, bool? showRejected = null, SessionType sessionType = SessionType.Prod);

参数说明：

posStart 起始位置，默认为0

numLimit 限制返回的商品数量,默认为50

sinceDate 起始日期（商品更新）

showRejected 为true 包含了wish review 中拒绝的商品

以上参数均为选填

调用示例:

var products= Wl.Wish.Product.InstantAPIs.InstantApi.ProductAllRetrieve(

accessToken:access\_token,

posStart:20,

numLimit:50,

sinceDate:”2017-01-01”,

showRejected:true

);

返回json格式：

{'code': 0,

'data': [{'Product': {'description': 'Example description',

'id': '5284e18fb5baba49d5xxxxxx',

'name': 'Cute Ring',

'number\_saves': '6',

'number\_sold': '0',

'parent\_sku': 'Cute Bow Ring',

'review\_status': 'approved',

'tags': [{'Tag': {'id': 'womensring', 'name': 'womens ring'}},

{'Tag': {'id': 'jewelry', 'name': 'Jewelry'}},

{'Tag': {'id': 'bow', 'name': 'Bow'}}],

'variants': [{'Variant': {'color': 'green',

'enabled': 'True',

'id': '5284e192b111ba49d5xxxxxx',

'product\_id': '5284e18fb5baba49d5xxxxxx',

'inventory': '11',

'msrp': '113.9',

'price': '110.9',

'shipping': '10.5',

'sku': 'AA1'}},

{'Variant': {'color': 'blue',

'enabled': 'True',

'id': '5284e19qqqbaba49d5bbbbbb',

'product\_id': '5284e18fb5baba49d5xxxxxx',

'inventory': '100',

'msrp': '19.9',

'price': '15.9',

'shipping': '10.5',

'sku': 'ZZ1'}}]}},

{'Product': {'description': 'Example Description',

'id': '5284efafb5bab119d1zzzzzz',

'name': 'Fairisle Scarf',

'number\_saves': '0',

'number\_sold': '0',

'parent\_sku': 'Fairisle Scarf',

'review\_status': 'rejected',

'tags': [{'Tag': {'id': 'cottonscarf', 'name': 'cotton scarf'}},

{'Tag': {'id': 'fashionaccessorie', 'name': 'Fashion Accessories'}},

{'Tag': {'id': 'fashion', 'name': 'Fashion'}},

{'Tag': {'id': 'scarf', 'name': 'scarf'}}],

'variants': [{'Variant': {'color': 'gray',

'enabled': 'True',

'id': '5284efb1b111ba49d1qqqqqq',

'product\_id': '5284efafb5bab119d1zzzzzz',

'inventory': '1010',

'msrp': '25.9',

'price': '21.91',

'shipping': '10.0',

'sku': 'AA4'}},

{'Variant': {'color': 'red',

'enabled': 'True',

'id': '5284efbaaababa49d1eiwqdf',

'product\_id': '5284efafb5bab119d1zzzzzz',

'inventory': '1100',

'msrp': '21.99',

'price': '21.91',

'shipping': '11.0',

'sku': 'AAB'}}]}}],

'message': '',

'paging': {'next': 'https://china-merchant.wish.com/api/v2/product/multi-get?start=22&limit=2&since=2014-10-15&access\_token=an\_example\_access\_token',

'previous': 'https://china-merchant.wish.com/api/v2/product/multi-get?start=18&limit=2&since=2014-10-15&access\_token=an\_example\_access\_token'}}

* + 1. 移除商品的图片，只保留主图（通过 product\_id或 parent\_sku ）

函数定义：

ProductResultBase ProductExtraPhotoRemove(string clientId,string accessToken, string productId, string parentSku, SessionType sessionType = SessionType.Prod);

调用示例：

var productBase=ProductExtraPhotoRemove(

accessToken:access\_token,

productId:product\_id,

parentSku:parent\_sku,

sessionType:SessionType.Sandbox

);

返回简单json, code 为 0 成功

* + 1. 更改商品的运费(统一)

函数定义:

ProductResultBase ProductShippingPriceEditSingle(string clientId,string accessToken, Wl.Wish.Entities.Request.ProductShippingPriceSingle productShippingPriceSingle, SessionType sessionType = SessionType.Prod);

ProductShippingPriceSingle 说明 ：

Id 商品的product\_id

parent\_sku 商品在wish上的sku

country 两位国家代码，见https://www.merchant.wish.com/documentation/api/v2#countries

price 运费(美元)，最大1000

use\_product\_shipping 如果为true，则对此国家使用商品规格中设置的到此国家的运费optional

enabled 对应国家是否发货optional

wish\_express 检查此商品是否在到相应国家的快运项目中注册，如有，货物要在5个工作日送达 optional

调用示例：

var productShippingSingle=ProductShippingPriceEditSingle(

accessToken:access\_token,

productShippingPriceSingle:product\_shipping\_price\_single,

sessionType:SessionType.Sandbox

);

返回简单json, code 为 0 成功

* + 1. 更改商品的运费(个别，多选)

函数定义：

ProductResultBase ProductShippingPriceEditMultiple(string clientId,string accessToken, Wl.Wish.Entities.Request.ProductShippingPriceMultiple productShippingPriceMultiple, SessionType sessionType = SessionType.Prod);

参数 ProductShippingPriceMultiple 说明

Id 必填 product\_id

AE, AR, AT, AU, BE, BG, BR, BY, CA, CH, CL, CO, CR, CZ, DE, DK, EC, EE, EG, ES, FI, FR, GB, GR, HK, HR, HU, ID, IE, IL, IN, IT, JM, JO, JP, KR, KW, LI, LT, LU, LV, MA, MC, MX, MY, NL, NO, NZ, PE, PH, PK, PL, PR, PT, RO, RU, SA, SE, SG, SI, SK, TH, TR, TW, UA, US, VE, VG, VI, VN, ZA, 各个不同国家的运费，美元，最大1000, 两位国家代码为参数，运费值 为参数值 ，具体能设定哪些参数请参看 <https://www.merchant.wish.com/documentation/api/v2#countries>

use\_product\_shipping\_countries 对哪些国家的运费使用商品规格中设置的值，以逗号相隔的两位国家代码

disabled\_countries 哪些国家的客户不能查看或是购买此商品

wish\_express\_add\_countries 商品必须在5个工作日内送达此国家的客户。促销商品也不可禁止加入Wish快运保障的国家，如果弄错了，要马上更正。

wish\_express\_remove\_countries 排除在Wish快运保障外的国家代码，以逗号分隔

调用示例：

var productShippingPriceMultiple=ProductShippingPriceEditMultiple(

accessToken:access\_token,

productShippingPriceMultiple:new{

id=”11111”,

AE=10.0,

US=12.00,

use\_product\_shipping\_countries=” AU,CA,US”,

disabled\_contries=”JP,KR”,

wish\_express\_add\_countries=”AB”,

wish\_express\_remove\_countries =”GH”},

sessionType:SessionType.Sandbox

);

返回简单json, code 为 0 成功

* + 1. 获得商品到某国家的运费

函数定义：

ProductShippingPriceSingle ProductShippingPriceRetrieveSingle(string clientId,string accessToken, Wl.Wish.Entities.Request.ProductShippingPriceSingle productShippingPriceSingle, SessionType sessionType = SessionType.Prod);

参数 ProductShippingPriceSingle说明

id product\_id

parent\_sku wish平台上的商品sku, parent\_sku 与 product\_id 必填其一

country 两位国家代码

调用示例：

var productShippingPriceSingle=ProductShippingPriceRetrieveSingle(

accessToken:access\_token,

productShippingPriceSingle:prduct\_shipping\_price\_single,

sessionType:SessionType.Sandbox

);

返回类型:

{

'code': 0,

'data': {

'ProductCountryShipping': {

'country\_code': 'CA',

'id': '123456789009876543211234',

'shipping\_price': '3.00',

'use\_product\_shipping': 'False',

'enabled': 'True',

'wish\_express': 'False'

}

},

'message': ''

}

* + 1. 获得商品的所有运费

函数定义：

ProductShippingPriceMultiple ProductShippingPriceRetrieveMultiple(string clientId,string accessToken, string productId = null, string parentSku = null, SessionType sessionType = SessionType.Prod);

调用示例：

var productShippingPriceMultiple=ProductShippingPriceRetrieveMultiple(

accessToken:access\_token,

productId:product\_id,

parentSku:parent\_sku,

sessionType:SessionType.Sandbox

);

返回类型：

{'code': 0,

'data': {'ProductCountryAllShipping': {

'id': '123456789009876543211234',

'shipping\_prices': [

{'ProductCountryShipping': {'country\_code': 'AU',

'enabled': 'True',

'id': '123456789009876543211234',

'shipping\_price': '1.0',

'use\_product\_shipping': 'False',

'wish\_express': 'True'}},

{'ProductCountryShipping': {'country\_code': 'CA',

'enabled': 'True',

'id': '123456789009876543211234',

'shipping\_price': 'Use Product Shipping Price',

'use\_product\_shipping': 'True',

'wish\_express': 'False'}},

{'ProductCountryShipping': {'country\_code': 'GB',

'enabled': 'True',

'id': '123456789009876543211234',

'shipping\_price': 'Use Product Shipping Price',

'use\_product\_shipping': 'True',

'wish\_express': 'True'}},

{'ProductCountryShipping': {'country\_code': 'US',

'enabled': 'True',

'id': '123456789009876543211234',

'shipping\_price': '0.9',

'use\_product\_shipping': 'False',

'wish\_express': 'False'}}]}},

'message': ''}

* + 1. 获得多个商品的所有运费设置

函数定义：

ProductShippingPriceMultipleOfMany ProductShippingPriceRetrieveMultipleOfMany(string clientId,string accessToken, string productIds = null, SessionType sessionType = SessionType.Prod);

调用示例：

var productsShippingPrice=ProductShippingPriceRetrieveMultipleOfMany(

accessToken:access\_token,

productIds:”123,456,789”, //最多50个id

sessionType:SessionType.Sandbox

);

返回json

{'code': 0,

'data': [

{'ProductCountryAllShipping': {'id': '123456789009876543211234',

'shipping\_prices': [

{'ProductCountryShipping': {'country\_code': 'AU',

'enabled': 'True',

'id': '123456789009876543211234',

'shipping\_price': '1.0',

'use\_product\_shipping': 'False',

'wish\_express': 'True'}},

{'ProductCountryShipping': {'country\_code': 'CA',

'enabled': 'True',

'id': '123456789009876543211234',

'shipping\_price': 'Use Product Shipping Price',

'use\_product\_shipping': 'True',

'wish\_express': 'True'}},

{'ProductCountryShipping': {'country\_code': 'GB',

'enabled': 'True',

'id': '123456789009876543211234',

'shipping\_price': 'Use Product Shipping Price',

'use\_product\_shipping': 'True',

'wish\_express': 'False'}},

{'ProductCountryShipping': {'country\_code': 'US',

'enabled': 'True',

'id': '123456789009876543211234',

'shipping\_price': '0.9',

'use\_product\_shipping': 'False',

'wish\_express': 'True'}}]}},

{'ProductCountryAllShipping': {'id': '111122223333444455556666',

'shipping\_prices': [

{'ProductCountryShipping': {'country\_code': 'AU',

'enabled': 'True',

'id': '111122223333444455556666',

'shipping\_price': '1.0',

'use\_product\_shipping': 'False',

'wish\_express': 'False'}},

{'ProductCountryShipping': {'country\_code': 'CA',

'enabled': 'True',

'id': '111122223333444455556666',

'shipping\_price': '2.0',

'use\_product\_shipping': 'False',

'wish\_express': 'True'}},

{'ProductCountryShipping': {'country\_code': 'GB',

'enabled': 'True',

'id': '111122223333444455556666',

'shipping\_price': '3.0',

'use\_product\_shipping': 'False',

'wish\_express': 'False'}},

{'ProductCountryShipping': {'country\_code': 'US',

'enabled': 'True',

'id': '111122223333444455556666',

'shipping\_price': 'Use Product Shipping Price',

'use\_product\_shipping': 'True',

'wish\_express': 'False'}}]}}],

'message': ''}

* + 1. 获得运费设置

函数定义:

ProductShippingSettings ProductShippingSettingsRetrieve(string clientId,string accessToken, SessionType sessionType = SessionType.Prod);

调用示例：

var productShippingSetting=ProductShippingSettingsRetrieve(

accessToken:access\_token,

sessiionType:SessionType.Sandbox

);

返回类型：

{'code': 0,

'data': {'ShippingSetting': {'country\_settings': [

{'CountryShippingSetting': {'country\_code': 'AU',

'shipping\_price': '1.00',

'use\_product\_shipping': 'False'}},

{'CountryShippingSetting': {'country\_code': 'CA',

'shipping\_price': '2.00',

'use\_product\_shipping': 'False'}},

{'CountryShippingSetting': {'country\_code': 'GB',

'shipping\_price': '3.00',

'use\_product\_shipping': 'False'}},

{'CountryShippingSetting': {'country\_code': 'US',

'shipping\_price': 'Use Product Shipping Price',

'use\_product\_shipping': 'True'}}],

'ships\_worldwide': 'False'}},

'message': ''}

1. **Variant 规格部分**
   1. 添加一个商品规格

函数定义:

VariantResult ProductVariantCreate(string clientId,string accessToken, Wl.Wish.Entities.Request.Variant variant, SessionType sessionType = SessionType.Prod);

参数 Variant 说明

parent\_sku 所属商品sku,

sku 商品规格 sku ,

inventory 库存 ,

price 价格,

shipping 运费,

color 可空,

size 可空,

msrp 建议零售价 可空,

shipping\_time 货运时间，可空,

main\_image 主图片 可空

调用示例：

var variant= ProductVariantCreate(

accessToken:access\_token,

variant:variant\_para,

sessionType:SessionType.Sandbox

);

返回数据：

{'code': 0,

'data': {'Variant': {'enabled': 'True',

'id': '529e6c2cf94aaa0cfe02846f',

'inventory': '10',

'price': '10',

'shipping': '3',

'size': '12',

'sku': 'red-shoe-12'}},

'msg' : '',

}

* 1. 获得商品规格数据

函数定义：

VariantResult ProductVariantRetrieve(string clientId,string accessToken, string sku, SessionType sessionType = SessionType.Prod);

调用示例：

var variant=ProductVariantRetrieve(

accessToken:access\_token,

sku:”111111”,

sessionType:SessionType.Sandbox);

返回数据：

{'code': 0,

'data': {'Variant': {'enabled': 'True',

'id': '51e0a2c61111a236cfffe3a2',

'product\_id' : '51e0a2c61111axxxcfffeyyy',

'inventory': '10100',

'msrp': '21100.0',

'price': '100.0',

'shipping': '10.0',

'sku': 'red-shoe-8'}},

'message': ''}

* 1. 修改商品规格

函数定义：

VariantResultBase ProductVariantUpdate(string clientId,string accessToken, Wl.Wish.Entities.Request.Variant variant, SessionType sessionType = SessionType.Prod);

参数 variant 说明

/// sku 商品唯一标识,

/// inventory 库存数，可空,

/// price 价格，可空,

/// shipping 运费，可空,

/// enabled 上/下架，可空,

/// color 可空,

/// size 可空,

/// msrp 建议零售价，可空,

/// shipping\_time 运输时间，可空,

/// main\_image 主图，可空

调用示例：

var variant= ProductVariantUpdate(

accessToken:access\_token,

varint:variant\_para,

sessionType:SessionType.Sandbox

);

返回 最基本json, code为0 ,成功

* 1. 变更商品规格的SKU

函数定义：

VariantResultBase ProductVariantSkuChange(string clientId,string accessToken, string previousSku, string newSku, SessionType sessionType = SessionType.Prod);

调用示例：

var variant=ProductVariantSkuChange(

accessToken:access\_token,

previousSku:previous\_sku,

newSku:new\_sku,

sessionType:SessionType.Sandbox

);

返回最简单json , code 为0 成功

* 1. 启用一个商品规格

函数定义 ：

VariantResultBase ProductVariantEnable(string clientId,string accessToken, string sku, SessionType sessionType = SessionType.Prod);

调用示例：

var variant=ProductVariantEnable(

accessToken:access\_token,

sku:”123456”,

sessionType:SessionType.Sandbox

);

返回简单json code 为0 成功

* 1. 禁用一个商品规格

函数定义：

VariantResultBase ProductVariantDisable(string clientId,string accessToken, string sku, SessionType sessionType = SessionType.Prod);

调用示例：

var variant=ProductVariantDisable(

accessToken:access\_token,

sku:”123456”,

sessionType:SessionType.Sandbox

);

返回基本json , code 为 0 成功

* 1. 更新商品规格的库存数量

函数定义：

VariantResultBase ProductVariantInventoryUpdate(string clientId,string accessToken, string sku, int inventory, SessionType sessionType = SessionType.Prod);

调用示例：

var variant=ProductVariantInventoryUpdate(

accessToken:access\_token,

sku:”123456”,

inventory:50,

sessionType:SessionType.Sandbox

);

返回 基本 json, code 为0 成功

* 1. 获取所有商品规格数据

函数定义：

VariantList ProductVariantListAll(string clientId,string accessToken, int start = 0, int limit = 50, SessionType sessionType = SessionType.Prod);

调用示例：

var variants=ProductVariantListAll(

accessToken:access\_token,

start:10,

limit:50,

sessionType:SessionType.Sandbox

);

返回json

{'code': 0,

'data': [{'Variant': {'enabled': 'True',

'id': '5215451b31111f73ff2xxxxxx',

'product\_id' : '51e0a2c61111axxxcfffeyyy',

'inventory': '1314',

'msrp': '150.0',

'price': '100.0',

'shipping': '21.0',

'sku': 'MMM'}},

{'Variant': {'color': 'blue',

'enabled': 'True',

'id': '5214c1111c238837cdiiiiii',

'product\_id' : '51e0a2c61111axxxcfffeyyy',

'inventory': '917',

'msrp': '100.0',

'price': '50.0',

'shipping': '9.81',

'sku': 'DD1111'}}],

'message': '',

'paging': {'next': 'https://china-merchant.wish.com/api/v2/variant/multi-get?start=22&limit=2&access\_token=an\_example\_access\_token',

'previous': 'https://china-merchant.wish.com/api/v2/variant/multi-get?start=18&limit=2&access\_token=an\_example\_access\_token'}}

1. **Notification 通知消息部分**
   1. 获取未查看的通知消息

函数定义 ：

NotificationList NotificationUnviewedRetrieve(string clientId,string accessToken, int startNumber = 0, int limitNumber = 50, SessionType sessionType = SessionType.Prod);

调用示例：

var notifications=NotificationUnviewedRetrieve(

accessToken:access\_token,

startNumber:0,

limitNumber:50,

sessionType:SessionType.Sandbox

);

返回json:

{"message":"","code":0,"data":[

{"GetNotiResponse":{

"perma\_link":"http:\/\/merchant.wish.com\/release\/57180d6c49xxxxxxxxxxxxxx",

"message":"Sizing Chart XXXXX",

"id":"571810da7exxxxxxxxxxxxxx",

"title":"There is a new xxxxxxxxxx"}},

{"GetNotiResponse":{

"perma\_link":"http:\/\/merchant.wish.com\/release\/57180d6c49xxxxxxxxxxxxxx",

"message":"Sizing Chart XXXXX",

"id":"571812abf9xxxxxxxxxxxxxx",

"title":"There is a new xxxxxxxxxx"}}],

"paging":{"next":"https:\/\/merchant.wish.com\/api\/v2\/noti\/fetch-unviewed?start=12","previous":"https:\/\/merchant.wish.com\/api\/v2\/noti\/fetch-unviewed?start=8"}}

* 1. 标记通知为已读

函数定义：

NotificationResultBase NotificationMarkViewed(string clientId,string accessToken, string notificationId, SessionType sessionType = SessionType.Prod);

调用示例：

var notificationMark=NotificaionMarkViewed(

accessToken:access\_token,

notificationId:”x090909099”,

sessionType:SessionType.Sandbox

);

返回简单JSON code 为 0 成功

* 1. 获取未查看通知的条数

函数定义：

NotificationResultBase NotificationUnviewedCountRetrieve(string clientId,string accessToken, SessionType sessionType = SessionType.Prod);

调用示例：

var notificationCount=NotificationUnviewedCountRetrieve(

accessToken:access\_token,

sessionType:SessionType.Sandbox

);

返回 json

{

'code': 0,

'data': {'count': 8},

'message': ''

}

* 1. 获取公告

函数定义 ：

AnnouncementList AnnouncementRetrieve(string clientId,string accessToken, SessionType sessionType = SessionType.Prod);

调用示例：

var announcements=AnnouncementRetrieve(

accessToken:access\_token,

sessionType:sessionType.Sandbox

);

返回数据

{

"message": "",

"code": 0,

"data": {

"GetBDAnnResponse": {

"title": "51734e8e4e0b8c03XXXXXXXXXXXXXXXXXXX",

"created": "Apr 01, 2016",

"bd\_id": "55781a50d29XXXXXXXXXX",

"bd\_name": "TraXX XXXX",

"message": "611f8c2259275bb6957fXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX",

"viewed": "False"

}

}

}

* 1. 获取系统更改通知

函数定义 ：

SystemUpdateNotification SystemUpdateNotificationRetrieve(string clientId,string accessToken, SessionType sessionType = SessionType.Prod);

调用示例：

var systemUpdateNotifications=SystemUpdateNotificationRetrieve(

accessToken:access\_token,

sessionType:SessionType.Sandbox

);

返回 json

{

"message": "",

"code": 0,

"data": [

{

"SystemUpdatesResponse": {

"release\_date": "Thursday May 19, 2016",

"id": "573e46218db5d1487ee4bdd8",

"features": [

{

"Feature": {

"cn\_title": "51734e8e66f465b06bcf4e2a4ea754c156fd96458fd08d39768465b0529f80fd",

"body": "We are very excited to announce that you can set international shipping prices per product now. No matter what sizes and weights your products are or what countries you are shipping to, you can always set shipping prices accordingly. You can start with changing overall default shipping prices on shipping settings page. Next step is to update per product shipping prices for specific countries by updating manually or a CSV file.\n\nFor a step to step guide, please visit our FAQ page for more information:",

"title": "New Feature About Updating International Shipping Prices Per Product",

"link": "http://merchantfaq.wish.com/hc/en-us/articles/218812008",

"cn\_body": "5c0a656c768455466237ff0c\n\ \n\n5982679c60a860f3898183b753d666f4591a4fe1606f548c51774f5364cd4f5c6b659aa4ff0c8bf78bbf95ee5e3889c195ee9898:",

"cn\_link": "http://merchantfaq.wish.com/hc/zh-cn/articles/218812008"

}

}

]

}

},

{

"SystemUpdatesResponse": {

"release\_date": "Monday August 03, 2015",

"id": "55bfae216daddd41f93512ce",

"features": [

{

"Feature": {

"cn\_title": "55466237APP(Android)670065b04e0a7ebf",

"body": "We are happy to announce a new mobile app for merchants. This app will be released initially on Android and will allow merchants to keep up to date with the latest metrics about their store. The app will also enable you to get the latest system updates and notifications right to your phone.\n\nGet the app from Google play, Xiaomi store, or direct download here: ",

"title": "Announcing New Merchant Android App",

"link": "http://merchant.wish.com/mobile",

"cn\_body": "62114eec5f889ad8517457305ba35e03ff1aWish55466237APP4e0a7ebf4e8630028be5APP4f1a9996514857285b8953537cfb7edf4e0a7ebf, 8ba95e7f5927554662375728624b673a4e0a4fbf80fd67e5770b5e9794fa51857684670065b06570636e52a86001ff0c4e8689e3670065b076847cfb7edf66f465b0ff0c5e766536523066f465b04fe1606f768453ca65f6901a77e53002\n\n8bf74eceGoogle playff0cXiaomi5e94752855465e974e0b8f7d30024e5f53ef76f463a570b951fb4ee54e0b94fe63a54e0b8f7dff1a",

"cn\_link": "http://merchant.wish.com/mobile"

}

}

]

}

}

]

}

* 1. 获取违规通知列表

函数定义：

InfractionList InfractionListRetrieve(string clientId,string accessToken, int startNumber = 0, int limitNumber = 50, SessionType sessionType = SessionType.Prod);

调用示例：

var infractions=InfractionListRetrieve(

accessToken:access\_token,

startNumber:0,

limitNumber:50,

sessionType:SessionType.Sandbox

);

返回json

{

"message": "",

"code": 0,

"data": {

"InfractionsResponse": {

"infractions": [

{

"Infraction": {

"link": "https://merchant.wish.com/warning/view/5475d08d6fa88c67d5416fb9",

"id": "5475d08d6fa88cxxxxxxxxxx"

}

},

{

"Infraction": {

"link": "https://merchant.wish.com/warning/view/547720da6fa88cXXXXXXXXXX",

"id": "547720da6fa88cXXXXXXXXXX"

}

},

{

"Infraction": {

"link": "https://merchant.wish.com/warning/view/547872746fa88cXXXXXXXXXX",

"id": "547872746fa88cXXXXXXXXXX"

}

},

{

"Infraction": {

"link": "https://merchant.wish.com/warning/view/5479c3dc6fa88cXXXXXXXXXX",

"id": "5479c3dc6fa88cXXXXXXXXXX"

}

},

{

"Infraction": {

"link": "https://merchant.wish.com/warning/view/5479c3e16fa88cXXXXXXXXXX",

"id": "5479c3e16fa88cXXXXXXXXXX"

}

}

]

}

},

"paging": {

"next": "https://merchant.wish.com/api/v2/get/infractions?access\_token=57685a3d1ce64a5479c3e16fa88cXXXXXXXXXX&start=5"

}

}

* 1. 获取违规通知次数

函数定义 ：

CountInfractionsResult InfractionCountRetrieve(string clientId,string accessToken, SessionType sessionType = SessionType.Prod);

调用示例：

var infractionCount=InfractionCountRetrieve(

accessToken:access\_token,

sessionType:SessionType.Sandbox

);

返回json

{

"message": "",

"code": 0,

"data": {

"CountInfractionsResponse": {

"count": "156"

}

}

}

1. **Order 订单部分**
   1. 获取订单

函数定义：

OrderResult OrderRetrieve(string clientId,string accessToken, string orderId, SessionType sessionType = SessionType.Prod);

调用示例：

var order=OrderRetrieve(

accessToken:access\_token,

orderId:”Od20170120340999120002”,

sessionType:SessionType.Sandbox

);

返回json

{

"code": 0,

"data": {

"Order": {

"ShippingDetail": {

"city": "North Bay",

"country": "US",

"name": "Mick Berry",

"phone\_number": "+1 555-181-7247",

"state": "NC",

"street\_address1": "2126 PO Box 5 Rt 49",

"zipcode": "13123"

},

"last\_updated": "2013-12-06T20:20:20",

"order\_time": "2013-12-06T20:20:20",

"order\_id": "123456789009876543210164",

"order\_total": "17.6",

"product\_id": "1113fad43deaf71536cb2c74",

"buyer\_id": "1234fad43deaf71536cb2c74",

"quantity": "2",

"price": "8",

"cost": "6.8",

"shipping": "2.35",

"shipping\_cost": "2",

"product\_name": "Dandelion Necklace",

"product\_image\_url": "http://d1zog42tnv26ho.cloudfront.net/4fea11fac43bf532f4001419-normal.jpg",

"days\_to\_fulfill": "2",

"hours\_to\_fulfill": "49",

"sku": "Dandelion Necklace",

"state": "APPROVED",

"transaction\_id": "11114026a99e980d4e500269",

"tracking\_confirmed": "True",

"tracking\_confirmed\_date": "2013-12-13T05:07:12",

"variant\_id": "1111fad63deaf71536cb2c76"

}

},

"message": ""

}

* 1. 获取最近有变更的订单

函数定义：

OrderList OrdersRecentlyChangesRetrieve(string clientId,string accessToken, int start = 0, int limit = 50, string since = null, bool? wish\_express\_only = null, SessionType sessionType = SessionType.Prod);

参数说明：

start 起始位置

limit 获取订单数量

since 起始时间

wish\_express\_only 是否只返回 wish express 数据

调用示例：

var orders=OrdersRecentlyChangesRetrieve(

accessToken:access\_token,

start:0,

limit:100,

since:” 2014-01-20”,

wish\_express\_only:false,

sessionType:SessionType.Sandbox

);

返回 json

{

"code": 0,

"data": [

{

"Order": {

"ShippingDetail": {

"city": "New York City",

"country": "US",

"name": "Guadalupe Smith",

"phone\_number": "5554191609",

"state": "California",

"street\_address1": "3317 w 51st Street",

"zipcode": "60632"

},

"last\_updated": "2014-01-20T20:20:20",

"order\_time": "2014-01-20T20:20:20",

"order\_id": "123456789009876543210164",

"order\_total": "17.6",

"product\_id": "1113fad43deaf71536cb2c74",

"buyer\_id": "1234fad43deaf71536cb2c74",

"quantity": "2",

"price": "8",

"cost": "6.8",

"shipping": "2.35",

"shipping\_cost": "2",

"product\_name": "Dandelion Necklace",

"product\_image\_url": "http://d1zog42tnv26ho.cloudfront.net/4fea11fac43bf532f4001419-normal.jpg",

"days\_to\_fulfill": "2",

"hours\_to\_fulfill": "49",

"sku": "Dandelion Necklace",

"state": "APPROVED",

"transaction\_id": "11114026a99e980d4e500269",

"variant\_id": "1111fad63deaf71536cb2c76"

}

},

{

"Order": {

"ShippingDetail": {

"city": "Lake City",

"country": "US",

"name": "Marko Schroeder",

"phone\_number": "+1 555-399-7785",

"state": "NY",

"street\_address1": "20685 W Verona Ave",

"zipcode": "60046"

},

"last\_updated": "2014-01-20T20:20:20",

"order\_time": "2014-01-20T20:20:20",

"order\_id": "1114a7cfb2ec2d42d272b627",

"order\_total": "17.6",

"product\_id": "1113fad43deaf71536cb2c74",

"buyer\_id": "1234fad43deaf71536cb2c74",

"quantity": "2",

"price": "8",

"cost": "6.8",

"shipping": "2.35",

"shipping\_cost": "2",

"product\_name": "Dandelion Necklace",

"product\_image\_url": "http://d1zog42tnv26ho.cloudfront.net/4fea11fac43bf532f4001419-normal.jpg",

"days\_to\_fulfill": "2",

"hours\_to\_fulfill": "2",

"sku": "Dandelion Necklace",

"state": "APPROVED",

"transaction\_id": "11114026a99e980d4e500269",

"tracking\_confirmed": "True",

"tracking\_confirmed\_date": "2013-12-13T05:07:12",

"variant\_id": "1111fad63deaf71536cb2c76"

}

}

],

"message": "",

"paging": {

"next": "https://china-merchant.wish.com/api/v2/order/multi-get??start=300&limit=100&since=2014-01-20&access\_token=an\_example\_access\_token",

"previous": "https://china-merchant.wish.com/api/v2/order/multi-get?start=100&limit=100&since=2014-01-20&access\_token=an\_example\_access\_token"

}

}

* 1. 获取未履约订单

未履约订单 https://www.merchant.wish.com/documentation/api/v2#list-unfulfilled-orders

根据此处未履约订单的地址 为 https://china-merchant.wish.com/api/v2/order/get-fulfill

此处存疑

函数定义：

OrderList OrdersUnfulfilledRetrieve(string clientId,string accessToken, int start = 0, int limit = 50, string since = null, bool? wish\_express\_only = null, SessionType sessionType = SessionType.Prod);

调用示例：

var ordersUnfulfilled=OrdersUnfulfilledRetrieve(

accessToken:access\_token,

start:0,

limit:50,

since:”2017-01-01”,

wish\_express\_only:false,

sessionType:SessionType.Sandbox

);

返回 json 同上

* 1. 标记订单已履约

函数定义：

OrderResultBase OrderFulfill(string clientId,string accessToken, string orderId, string trackingProvider, string trackingNumber = null, string shipNote = null, SessionType sessionType = SessionType.Prod);

参数说明

orderId 订单编号

trackingProvider 物流

trackingNumber 运单号

shipNote 备注(仅卖家见)

调用示例:

var orderResult=OrderFulfill(

accessToken:access\_token,

orderId:”1114a7cfb2ec2d42d272b627”,

trackingProvider:”USPS”,

trackingNumber:”UT212009334449999”,

shipNote:”ship note ”,

sessionType:SessionType.Sandbox

);

返回 json

{

'code': 0,

'data': {'success': True},

'message': 'Your order is been processed right now!'

}

* 1. 订单取消/退款

函数定义:

OrderResultBase OrderRefundOrCancel(string clientId,string accessToken, string orderId, Wl.Wish.Entities.ResonCodeOfRefundOrCancel reasonCode = Wish.Entities.ResonCodeOfRefundOrCancel.Other, string reasonNote = null, SessionType sessionType = SessionType.Prod);

参数说明:

orderId 订单id

reasonCode

代码 表示原因

-1 其他

18 误下单了

20 配送时间过长

22 商品不合适

23 收到错误的商品

24 商品为假冒伪劣品

25 商品已损坏

26 商品功能与描述不符

27 商品与产品描述不符

30 产品被配送至错误的地址

31 用户提供了错误的地址

32 商品退还至发货人

33 Incomplete Order

34 店铺无法履行订单

35 此件显示已妥投，但客户未收到。

1001 Received the wrong color

1002 Item is of poor quality

1004 Product listing is missing information

1005 Item did not meet expectations

1006 Package was empty

Refund Reason Codes – 仅未履约订单

1 店铺无法履行订单

调用示例：

var orderResult= OrderRefundOrCancel(

accessToken:access\_token,

orderId:”111111”,

reasonCode: Wish.Entities.ResonCodeOfRefundOrCancel.MerchandiseNotAsDescribed,

reasonNote:””,

sessionType:SessionType.Sandbox

);

返回 json

{

'code': 0,

'data': {'success': True},

'message': ''

}

* 1. 修改已发货订单的跟踪信息

函数定义:

OrderResultBase OrderShippedModifyTracking(string clientId,string accessToken, string orderId, string trackingProvider, string trackingNumber = null, string shipNote = null, SessionType sessionType = SessionType.Prod);

参数说明：

orderId 订单id

trackingProvider 物流

trackingNumber 物流单号

shipNote 备注(仅卖家可见)

调用示例：

var orderResult=OrderShippedModifyTracking(

accessToken:access\_token,

orderId:order\_id,

trackingProvider:tracking\_provider,

trackingNumber:tracking\_number,

shipNote:ship\_note,

sessionType:SessionType.Sandbox

);

调用 json

{

'code': 0,

'data': {'success': True},

'message': ''

}

* 1. 修改还未标记发货订单的收货信息

函数定义：

OrderResultBase OrderNotShippedModifyShippingInformation(string clientId,string accessToken, string orderId, string streetAddress1, string city, string zipCode, string country, string streetAddress2 = null, string state = null, string phoneNumber = null, SessionType sessionType = SessionType.Prod);

参数说明 ：

orderId 订单id

streetAddress1 地址1

city 城市

zipCode 邮编

country 国家

streetAddress2 地址2

state 州

phoneNumber 联系电话

调用示例：

var orderResult=OrderNotShippedModifyShippingInformation(

accessToken:access\_token,

orderId:” 098765432112345678901234”,

streetAddress1:” 123 Fake St”,

city:” Montrea”,

zipCode:zipcode,

country:”US”,

streetAddress2:street\_address\_2,

state:”CA”,

phoneNumber:phone\_number,

sessionType:SessionType.Sandbox

);

返回 基本json

{

'code': 0,

'data': {'success': True},

'message': ''

}

* 1. 开始一个订单下载的任务

函数定义 ：

OrderDownloadTask OrderDownload(string clientId,string accessToken, string startDate = null, string endDate = null, int? limitNumber = null, string sortNotation = "desc", SessionType sessionType = SessionType.Prod);

参数说明：

startDate 起始日期

endDate 截止日期

limitNumber 下载数量

sortNotation 按订单的时间 顺序 或是 逆序

调用示例：

var orderDownload=OrderDownload(

accessToken:access\_token,

startDate:”2017-01-01”,

endDate:”2017-04-01”,

limitNumber:50,

sortNotation:”desc”,

sessionType:SessionType.Sandbox

);

返回 json

{

'code': 0,

'data': {'job\_id': "57bb5803ba2a1f181de31b01"},

'message': ''

}

Job\_id 可以作为 稍后的查询依据

* 1. 查询订单下载任务

函数定义:

OrderDownloadTask OrderDownloadQuery(string clientId,string accessToken, string jobId, SessionType sessionType = SessionType.Prod);

调用示例：

var orderDownload=OrderDownloadQuery(

accessToken:access\_token,

jobId:job\_id,

sessionType:SessionType.Sandbox

);

返回 json

未完成的情况

{

'code': 0,

'data': {

"status": "PENDING",

"created\_date": "2016-08-18 02:40:25.238000"

},

'message': ''

}

已完成的情况

{

"code": 0,

"data": {

"status": "FINISHED",

"total\_count": 20553,

"processed\_count": 20553,

"download\_link": "https://merchant.wish.com/static/sweeper-production-merchant-export/52f64e7aab980a038d62d61e-57bce38526778a33b3f8c375-2016-08-16-19:27:01.csv",

"created\_date": "2016-08-18 02:40:25.238000",

"start\_run\_time": "2016-08-18 03:12:07.135000",

"end\_run\_time": "2016-08-18 02:49:28.798000"

},

"message": ""

}

* 1. 取消一个订单下载任务

函数定义:

OrderDownloadTask OrderDownloadCancel(string clientId,string accessToken, string jobId, SessionType sessionType = SessionType.Prod);

调用示例：

var orderDown=OrderDownloadCancel(

accessToken:access\_token,

jobId:job\_id,

sessionType:SessionType.Sandbox

);

返回 json

取消成功

{

'code': 0,

'data': {'message': 'Job cancelled successfully.'},

'message': ''

}

取消已经完成的订单返回状态

{

'code': 0,

'data': {'message': 'Job has already finished.'},

'message': ''

}

取消已经取消的订单

{

'code': 0,

'data': {'message': 'Job has already been cancelled.'},

'message': ''

}

1. **Ticket 工单部分**
   1. 获取一个Ticket

函数定义：

TicketResult TicketRetrieve(string clientId,string accessToken, string ticketId, SessionType sessionType = SessionType.Prod);

调用示例：

var ticket=TicketRetrieve(

accessToken:access\_token,

ticketId:ticket\_id,

sessionType:SessionType.Sandbox

);

返回 json

{

"code": 0,

"data": {

"Ticket": {

"UserInfo": {

"id": "123456789012345678901234User",

"joined\_date": "2012-02-02T11:22:21",

"locale": "en",

"name": "George Forman"

},

"close\_date": "2015-03-08T22:31:02",

"closed\_by": "wish support",

"id": "T123456789012345678901234T",

"items": [

{

"Order": {

"ShippingDetail": {

"city": "",

"country": "US",

"name": "George Forman",

"phone\_number": "123456",

"state": "California",

"street\_address1": "123 Fake St",

"zipcode": "11111"

},

"buyer\_id": "123456789012345678901234",

"cost": "6.69",

"last\_updated": "2011-03-08T22:31:02",

"order\_id": "123456789012345678901234",

"order\_time": "2015-02-20T02:55:27",

"order\_total": "9.39",

"price": "6.4",

"product\_id": "123456789012345678901234",

"product\_image\_url": "https://google.com/not-a-image",

"product\_name": "A product",

"quantity": "1",

"refunded\_by": "REFUNDED BY WISH FOR MERCHANT",

"refunded\_reason": "Item did not work as described",

"refunded\_time": "2015-03-18",

"shipped\_date": "2015-02-20",

"shipping": "3.0",

"shipping\_cost": "2.7",

"shipping\_provider": "USPS",

"sk": "SKU123",

"state": "REFUNDED",

"tracking\_number": "TR1234ACK",

"transaction\_id": "1223456789012345678901234",

"variant\_id": "123456789012345678901234"

}

}

],

"label": "Return or Exchange",

"last\_update\_date": "2015-02-09T12:13:43",

"merchant\_id": "123456789012345678901234",

"open\_date": "2015-03-06T21:23:01",

"photo\_proof": "True",

"replies": [

{

"Reply": {

"date": "2015-03-06T21:23:01",

"image\_urls": "https://fake-url",

"message": "Where is my product?",

"sender": "user",

"translated\_message": "Where is my product?",

"translated\_message\_zh": "54EA91CC662F621176844EA754C1FF1F"

}

}

],

"state": "Awaiting your response",

"state\_id": "1",

"subject": "Return or Exchange",

"sublabel": "",

"transaction\_id": "123456789012345678901234"

}

},

"message": ""

}

* 1. 获取所有等待回复的Ticket

函数定义 ：

TicketList TicketRetrieveAll(string clientId,string accessToken, int startNumber = 0, int limitNumber = 50, SessionType sessionType = SessionType.Prod);

参数说明：

startNumber 起始位置

limitNumber 列表数量

调用 示例：

var tickets=TicketRetrieveAll(

accessToken:access\_token,

startNumber:start\_number,

limitNumber:limit\_number,

sessionType:SessionType.Sandbox

);

返回 json

{

"code": 0,

"data": [

{

"Ticket": {

"UserInfo": {

"id": "54f8e4ef2743dxxxxxxxxxxx",

"joined\_date": "2014-03-05T13:21:19",

"locale": "pt",

"name": "Mr. Rogers"

},

"default\_refund\_reason": "19",

"id": "54f8ecd2f0812xxxxxxxxxx",

"items": [

{

"Order": {

"ShippingDetail": {

"city": "San Fransicso",

"country": "BR",

"name": "Mr. Rogers",

"phone\_number": "1234567890",

"state": "Se3o Paulo",

"street\_address1": "123 Fake St",

"street\_address2": "Apt 5",

"zipcode": "10000000"

},

"buyer\_id": "54f8e4ef274\*\*\*\*\*\*\*\*\*\*\*\*\*",

"color": "black",

"cost": "16.2",

"last\_updated": "2014-03-06T07:37:16",

"order\_id": "123456789012345678901234",

"order\_time": "2015-03-05T23:40:53",

"order\_total": "22.5",

"price": "18.0",

"product\_id": "123456789012345678901234",

"product\_image\_url": "http://contestimg.wish.com/api/webimage/123456789012345678901234-normal.jpg",

"product\_name": "Cloth",

"quantity": "1",

"refunded\_by": "CANCELLED BY CUSTOMER",

"refunded\_reason": "Customer cancelled the order",

"refunded\_time": "2015-03-05",

"shipping": "7.0",

"shipping\_cost": "6.3",

"size": "M",

"sku": "SKU123-XL",

"state": "REFUNDED",

"transaction\_id": "123456789012345678901234",

"variant\_id": "5285bd3fb5baba7\*\*\*\*\*\*\*\*\*"

}

}

],

"label": "Request Refund/Cancel Order: Request a Different Color",

"last\_update\_date": "2015-03-09T12:13:57",

"merchant\_id": "123456789012345678901234",

"open\_date": "2015-03-05T23:54:58",

"photo\_proof": "False",

"replies": [

{

"Reply": {

"date": "2015-03-05T23:54:58",

"image\_urls": "[]",

"message": "Quero meu reembolso ",

"sender": "user",

"translated\_message": "I want my refund",

"translated\_message\_zh": "62115e0c671b6211768490006b3e"

}

}

],

"state": "Awaiting your response",

"state\_id": "4",

"subject": "Solicitar Reembolso/ Cancelar Pedido",

"sublabel": "Request a Different Color",

"transaction\_id": "123456789012345678901234"

}

},

{

"Ticket": {

"UserInfo": {

"id": "123456789012345678901234",

"joined\_date": "2011-12-12T05:22:52",

"locale": "en",

"name": "George Bush"

},

"id": "123456789012345678901234",

"items": [

{

"Order": {

"ShippingDetail": {

"city": "Waterloo",

"country": "CA",

"name": "George Bush",

"phone\_number": "1123-123-1234",

"state": "Ontario",

"street\_address1": "Fake St",

"zipcode": "A0A 0A0"

},

"buyer\_id": "12345678901234567890",

"cost": "4.6",

"last\_updated": "2015-03-09T10:23:13",

"order\_id": "123456789012345678901234",

"order\_time": "2012-12-12T06:40:12",

"order\_total": "29.2",

"price": "5.11",

"product\_id": "123456789012345678901234",

"product\_image\_url": "http://contestimg.wish.com/api/webimage/5253fd0b1xxxxxxxxxxx-normal.jpg",

"product\_name": "Automatic Car",

"quantity": "4",

"shipped\_date": "2011-12-12",

"shipping": "3.0",

"shipping\_cost": "2.7",

"shipping\_provider": "SwissPost",

"sku": "SKU123",

"state": "SHIPPED",

"tracking\_number": "TR1ACK",

"transaction\_id": "123456789012345678901234",

"variant\_id": "123456789012345678901234"

}

}

],

"label": "Shipping Status or ETA Inquiry",

"last\_update\_date": "2015-03-09T11:02:28",

"merchant\_id": "123456789012345678901234",

"open\_date": "2012-03-09T04:11:00",

"photo\_proof": "False",

"replies": [

{

"Reply": {

"date": "2015-03-09T04:57:33",

"image\_urls": "[]",

"message": "I did not yet receive this item and it is know march 8 2015 it was estimated that it would be here Jan 2 2015 and it has not arrived ",

"sender": "user",

"translated\_message": "I did not yet receive this item and it is know march 8 2015 it was estimated that it would be here Jan 2 2015 and it has not arrived ",

"translated\_message\_zh": "62118fd84e0d63a553d78fd94e2a987976eeff0c8fd9662f4e8689e320155e7436708865e5ff0c636e4f308ba1ff0c8fd95c06662f8fd991cc20155e7416708265e5ff0c4e5f6ca1670952308fbe"

}

},

{

"Reply": {

"date": "2015-03-09T10:23:12",

"image\_urls": "[]",

"message": "Hello,friend\n\nSorry for caused you any troubles \n\nWe got the sad news from our shipping agent that your parcel may be lost due to the long time no information online. \n\nPlease don&quto;t worry, we won&quto;t let our customer lost their money, would you like us to sent your a new parcel for your order to your address:\n\nDain Steiner\nbox 203\nTeslin, Yukon, Y0A 1B0\nCanada\n\nOr you perfer a full refund for this order? Your reply will be highly appreciated, have a nice day :)\n\nRegards\nScreamprice Service Team",

"sender": "merchant"

}

},

{

"Reply": {

"date": "2015-03-09T11:02:28",

"image\_urls": "[]",

"message": "Please send me a new parcel thank you",

"sender": "user",

"translated\_message": "Please send me a new parcel thank you",

"translated\_message\_zh": "8bf77ed962114e004e2a65b07684530588f98c228c22"

}

}

],

"state": "Awaiting your response",

"state\_id": "4",

"subject": "Where is my order?",

"sublabel": "",

"transaction\_id": "548a8de6baa0830ffxxxxxxx"

}

}

],

"message": "",

"paging": {

"next": "https://china-merchant.wish.com/api/v2/ticket/get-action-required?start=2&limit=2&access\_token=an\_example\_access\_token"

}

}

* 1. 回复Ticket

函数定义：

TicketResultBase TicketReply(string clientId,string accessToken, string ticketId, string replyContent = "", SessionType sessionType = SessionType.Prod);

参数说明：

ticketId 工单id

replyContent 回复内容

调用示例：

var ticketReply=TicketReply(

accessToken:access\_token,

ticketId:ticket\_id,

replyContent:reply\_content,

sessionType:SessionType.Sandbox

);

返回 json

{

'code': 0,

'data': {'success': True},

'message': ''

}

* 1. 关闭一个Ticket

函数定义：

TicketResultBase TicketClose(string clientId,string accessToken, string ticketId, SessionType sessionType = SessionType.Prod);

调用示例：

var ticketClose=TicketClose(

accessToken:access\_token,

ticketId:ticket\_id,

sessionType:SessionType.Sandbox

);

返回 json

{

'code': 0,

'data': {'success': True},

'message': ''

}

* 1. 就某个Ticket 请求Wish 的支持

函数定义:

TicketResultBase TicketAppealWishSupport(string clientId,string accessToken, string ticketId, SessionType sessionType = SessionType.Prod);

调用示例：

var ticketSupport=TicketAppealWishSupport(

accessToken:access\_token,

ticketId:ticket\_id,

sessionType:SessionType.Sandbox

);

返回 json

{

'code': 0,

'data': {'success': True},

'message': ''

}

* 1. 重新开启一个Ticket

函数定义：

TicketResultBase TicketReOpen(string clientId,string accessToken, string ticketId, string reasonToReOpen, SessionType sessionType = SessionType.Prod);

调用示例：

var ticketOpen=TicketReOpen(

accessToken:access\_token,

ticketId:ticket\_id,

reasonToReOpen:reason\_to\_reopen,

sessionType:SessionType.Sandbox

);

返回示例：

{

'code': 0,

'data': {'success': True},

'message': ''

}

1. **与程序集成的步骤**
   1. **先去平台鉴权并返回预授权码**

**取出对应 店铺表中的clientId,returnUrl 调用** Wl.Wish.Open.OAuthAPIs.OAuthApi.GetAuthorizeUrl(clientId:client\_id, redirectUrl:redirect\_url,sessionType:SessionType.Sandbox)来获得鉴权页面链接，并访问。

至于 sessionType 是用 Sandbox (沙盒) 还是 Prod (正式) 依据实际情况

鉴权后wish平台 会转向到给定的 return url 并带入 code 参数,因 wish 要求 redirect url 是一个安全链接，如果未能启用安全链接，建议手动 取出 code 并置入店铺表中,并在5分钟内换取accessToken

* 1. **在容器中先注册**

Wl.Wish.Product.Containers.AccessTokenContainer.Register(string clientId, string clientSecret, string preAuthCode, string redirectUri, string name = null);

* 1. **用授权码换access\_token**

**调用** Wl.Wish.Open.OAuthAPIs.OAuthApi.GetAccessToken(string clientId, string clientSecret, string preAuthCode, string redirectUrl, SessionType sessionType = SessionType.Prod)

返回 一个 access\_token 实体