# TOP API 请求与响应

## 1．TOP API请求

介绍调用环境入口地址，接口的参数、签名算法、注意事项等

### 1.1 环境入口地址

TOP 请求调用是基于REST 协议的，环境入口地址：

[*http://gw.api.taobao.com/router/rest*](http://gw.api.taobao.com/router/rest)

同时，为了方便合作伙伴在应用上线前进行充分的测试， TOP对外提供一套沙箱测试环境，入口地址：

[*http://gw.api.tbsandbox.com/router/rest*](http://gw.api.tbsandbox.com/router/rest)

Content-Type 支持：

*application/x-www-form-urlencoded 和 multipart/form-data*

### 1.2 签名算法

通过加密签名的方式，保证请求TOP的信息由ISV发送。同时在发送过程中信息没有丢失或被篡改。

TOP目前对外提供两种签名方式：MD5 和 HMAC ，需要在系统参数sign\_method 中指定使用哪种签名方式。具体可参考：

[*http://open.taobao.com/dev/index.php/API%E7%AD%BE%E5%90%8D%E7%AE%97%E6%B3%95*](http://open.taobao.com/dev/index.php/API%E7%AD%BE%E5%90%8D%E7%AE%97%E6%B3%95)

### 1.3 Session授权

部分API调用是需要用户授权才可以的。目前TOP的授权方式有三种：

* 不需要：调用API不需要获取用户的授权信息，如：taobao.items.search
* 需 要：调用API，必须要取得用户的授权信息，如：taobao.item.add
* 可 选：调用API如果传入授权信息，将获取隐私数据，如：taobao.user.get

API的授权类型，可通过API 文档获取，如taobao.item.add的授权类型如图：



需要授权调用的API，需要将用户授权的SessionKey 作为系统参数session的值传递给TOP。 应用获取用户授权的方式可参考：

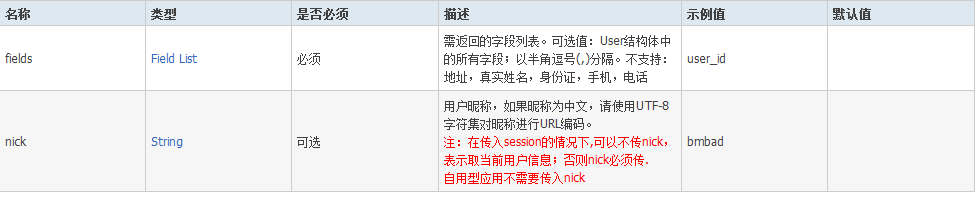
<http://open.taobao.com/dev/index.php/%E6%B7%98%E5%AE%9D%E7%94%A8%E6%88%B7%E6%8E%88%E6%9D%83%E6%96%B9%E5%BC%8F>

### 1.4 系统级参数

|  |  |  |  |
| --- | --- | --- | --- |
| **名称** | **类型** | **可选** | **描述** |
| method | String | N | API接口名称，更多接口详见[API文档](http://my.open.taobao.com/apidoc/main.htm" \t "_blank" \o "http://my.open.taobao.com/apidoc/main.htm) |
| session | String | Y | TOP分配给用户的SessionKey，自用型应用不需要传入该参数的；非自用型的应用对于需要用户授权的API必须要传入该参数，对于不需要用户授权的API则不必传入该参数。sessionKey获取方法详见文档：[获取sessionkey](http://open.taobao.com/dev/index.php/%E6%B7%98%E5%AE%9D%E7%94%A8%E6%88%B7%E6%8E%88%E6%9D%83%E6%96%B9%E5%BC%8F" \t "_blank" \o "http://open.taobao.com/dev/index.php/%E6%B7%98%E5%AE%9D%E7%94%A8%E6%88%B7%E6%8E%88%E6%9D%83%E6%96%B9%E5%BC%8F) |
| timestamp | String | N | 时间戳，格式为yyyy-MM-dd hh:mm:ss，例如：2008-01-25 20:23:30。淘宝API服务端允许客户端请求时间误差为10分钟 |
| format | String | Y | 可选，指定响应格式。默认xml,目前支持格式为xml,json |
| [app\_key](http://open.taobao.com/dev/index.php/AppKey) | String | N | 在[合作伙伴后台](http://my.open.taobao.com/isv/index.htm" \t "_blank" \o "http://my.open.taobao.com/isv/index.htm)创建应用时TOP分配的的AppKey |
| v | String | N | API协议版本，可选值:2.0 |
| [sign](http://open.taobao.com/dev/index.php/%E7%AD%BE%E5%90%8D) | String | N | 对调用API时所有输入参数（包括应用级参数）进行签名，参考[API签名算法](http://open.taobao.com/dev/index.php/API%E7%AD%BE%E5%90%8D%E7%AE%97%E6%B3%95) |
| sign\_method | String | N | 参数的加密方法选择，可选值是：md5,hmac |

### 1.5 应用级参数

TOP将参数根据是否与业务相关，分成系统级参数和应用级参数。系统级参数与API的具体业务没有关系，每个API在调用过程中都必须要有的，或者是有默认值。API的应用级参数可以在API文档上看到，如taobao.user.get的应用级参数如下图：



其中：

文档中指定了该接口接受的应用参数列表，以及这些参数的类型，是否必须等信息。

如果参数是必须的，则一定要传，否则报40错误。

如果参数有默认值，当传入为空时，则取默认值。

传入的参数类型格式，一定要符合该参数类型接受的格式，否则报41错误。应用级参数的类型格式可参考：

[*http://open.taobao.com/dev/index.php/%E5%8F%82%E6%95%B0%E7%B1%BB%E5%9E%8B%E8%AF%B4%E6%98%8E*](http://open.taobao.com/dev/index.php/%E5%8F%82%E6%95%B0%E7%B1%BB%E5%9E%8B%E8%AF%B4%E6%98%8E)

## 2．TOP API响应

TOP API 响应分为异常响应和正常响应 。 以下情况下会导致返回异常响应：

* 传入的参数错误：包括系统级和应用级参数格式、信息错误、加密校验失败、超频、没权限等
* 业务处理失败：当调用API 执行相应的业务逻辑时，相应的规则限制导致调用失败
* 系统异常：响应超时等

正常响应将返回接口的调用结果，具体信息可参考API接口的返回参数部分。

### 2.1 异常响应格式

对于异常响应格式，返回结果的根节点是error\_response ，同时将请求参数列表在args节点中返回。每个参数被封装在arg节点中，以key-value的方式。

code 节点指定了具体的错误码，*如果错误码在100以内，并且不等于15*，具体错误信息及解决方案可参考：

[*http://open.taobao.com/dev/index.php/%E9%94%99%E8%AF%AF%E7%A0%81%E4%B8%80%E8%A7%88%E8%A1%A8*](http://open.taobao.com/dev/index.php/%E9%94%99%E8%AF%AF%E7%A0%81%E4%B8%80%E8%A7%88%E8%A1%A8)

*如果错误码等于15或者大于100 ，则这时需要依据 sub\_code和sub\_msg 进行问题定位*，具体可参考API文档上对应的错误码列表进行解决。



#### 2.1.1 XML格式

<?xml version="1.0" encoding="utf-8" ?>

<error\_response>

<args list="true">

<arg><key>app\_key</key><value>12129701</value></arg>

<arg><key>fields</key><value>user\_id,uid</value></arg>

<arg><key>format</key><value>xml</value></arg>

<arg><key>method</key><value>taobao.user.get</value></arg>

<arg><key>nick</key><value>a</value></arg>

<arg><key>partner\_id</key><value>top-apitools</value></arg>

<arg><key>sign</key><value>9DCB20E358CF2D0887EAAEA5C030A410</value></arg>

<arg><key>timestamp</key><value>2011-05-25 15:44:50</value></arg>

<arg><key>v</key><value>2.0</value></arg>

</args>

*<code>500</code>*

*<msg>Remote service error</msg>*

*<sub\_code>isv.user-not-exist:invalid-nick</sub\_code>*

*<sub\_msg>根据用户昵称:a查询不到对应的用户信息</sub\_msg>*

</error\_response>

#### 2.1.2 JSON格式

{"error\_response":

{"args":{"arg":[

{"key":"app\_key","value":"12129701"},

{"key":"fields","value":"user\_id,uid"},

{"key":"format","value":"json"},

{"key":"method","value":"taobao.user.get"},

{"key":"nick","value":"a"},{"key":"partner\_id","value":"top-apitools"},

{"key":"sign","value":"28A6B53506CBA785A7373B1B14B99E67"},

{"key":"timestamp","value":"2011-05-25 15:46:19"},

{"key":"v","value":"2.0"}]

},

*"code":500,*

*"msg":"Remote service error",*

*"sub\_code":"isv.user-not-exist:invalid-nick",*

*"sub\_msg":"根据用户昵称:a查询不到对应的用户信息"*

}}

### 2.2 TOP API 正常响应格式

1. TOP对外提供支持XML和JSON 两种格式的返回。
2. 用户传入正确的参数请求TOP，获取正常的响应信息。
3. API文档中返回的参数结构规定了接口实际返回信息与格式。
4. XML格式执行字符转义， 不加CDATA ，转义列表如下：

< 🡪 &lt;

> 🡪 &gt;

‘ 🡪 &apos;

& 🡪 &amp;

“ 🡪 &quot;

1. XML格式将过滤部分不可见字符，列表：

**if** ((current == 0x9) || (current == 0xA) || (current == 0xD)

|| ((current >= 0x20) && (current <= 0xD7FF))

|| ((current >= 0xE000) && (current <= 0xFFFD))

|| ((current >= 0x10000) && (current <= 0x10FFFF)))

1. XML格式中，集合节点带上list=”true” ，方便做流解析。
2. JSON格式将做一下转义：

“ 🡪 \”

\ 🡪 \\

#### 3.2.1 根节点命名：

根节点命名取api名字去掉taobao前缀，把“.”转换成“\_”，再加上response组成，如：

taobao.user.get 🡪 user\_get\_response

**注意**：淘江湖API由于历史原因，格式有些不同，根节点为API名称后面两段部分，把“.”改成“\_”，再加上response组成，如：

taobao.jianghu.feed.get 🡪 feed\_get\_response

#### 3.2.2基本类型格式

对外响应的基本类型包括：String ， Number ， Price ， Date ， Boolean。

**其中：**

1. String 、 Date、Price的value值在JSON格式中以双引号包含，
2. date的格式为：yyyy-MM-dd HH:mm:ss，例如：2010-11-11 11:11:11
3. Number是数字串， 可以作为Long类型进行解析
4. Price 是 包含小数点后两位的正数，可以以float 或 double 进行解析
5. Boolean的结果包含： true 和 false

**XML的格式**：

<num\_iid>1489161932</num\_iid>

<nick>tbtest1061</nick>

<created>2010-11-11 11:11:11</ created>

<has\_more\_pic>false</has\_more\_pic >

<price>12.15</price>

**JSON 格式：**

"num\_iid": 1489161932

" nick": " tbtest1061"

" created": "2010-11-11 11:11:11"

" has\_more\_pic": false

"price": "12.15"

**其中** ： num\_iid、nick、created、has\_more\_pic、price等是API文档中返回参数的名称或返回参数包含的数据结构中基本属性的名称。

#### 3.2.3基本类型列表格式

基本类型列表是由基本类型组成的数组，他们是这些基本类型的值通过逗号分割组成一个字符串集合。在文档中显示的类型如：Number[] , String[] , Date[] , Price[] , Boolean[] 。如下图所示：



**XML格式：**

<list\_ids list="true">

<number>

1534234，1234，123

</number>

</list\_ids>

**JSON格式：**

"list\_ids": {

"number": [1534234，1234，123]

}

**其中：**

* + 1. list\_ids是API文档中返回结果或数据结构中属性的名称 。
    2. number是该结构存的基本类型名称转小写，可选值：number，string，price ， date ， boolean。
    3. 列表中的结果以逗号隔开

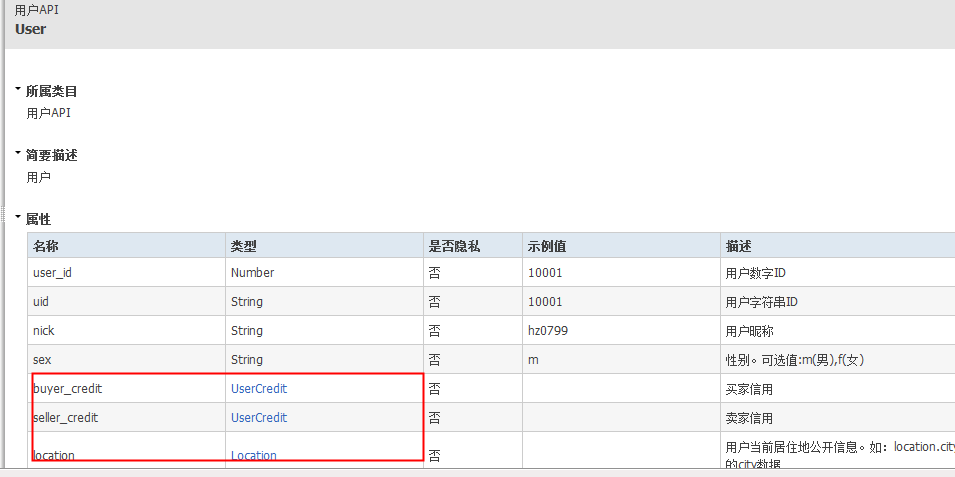
#### 3.2.4对象格式

对象相当于我们系统中用到的数据结构DO。对象中可包含基本类型、基本类型列表、对象、对象列表等。

比如taobao.user.get 获取单个用户对象信息，那么用户信息（如：uid ， sex ， location等）将统一放在User这个数据结构，方便理解。taobao.user.get接口返回结果如下图：



对象User的结构如（user\_id是基本类型， buyer\_credit、seller\_credit、location都是对象结构）：



**XMl格式：**

**根节点**

<**user\_get\_response**>

<**user**>

<user\_id>

**user节点的名称是API文档中录入的名称**

10001

</user\_id>

<nick>

hz0799

</nick>

**数据结构中嵌套数据结构**

<location>

<country>

中国

</country>

<district>

西湖区

</district>

</location>

</user>

**根节点**

</user\_get\_response>

**JSON格式：**

{

"**user\_get\_response**": {

"user": {

"user\_id": 10001,

"nick": "hz0799",

"location": {

"country": "中国",

"district": "西湖区"

}

}

}

}

#### 3.2.5对象列表格式

以taobao.users.get 接口为例，对象列表的文档显示如下图(接口返回的是User的列表结构)：



**XML格式：**

**users节点名称取的是返回结果的名称。**

<users\_get\_response>

**<users list="true">**

<user>

<user\_id>

10001

**users下可能包含一个或多个user的数据对象，user节点的名称取至User数据结构的名称，将驼峰转换为下划线**

</user\_id>

<nick>

hz0799

</nick>

<location>

<country>

中国

</country>

<district>

西湖区

</district>

</location>

</user>

**第二个user对象**

<user>

<user\_id>

10001

</user\_id>

<nick>

hz0799

</nick>

<location>

<country>

中国

</country>

<district>

西湖区

</district>

</location>

</user>

</users>

</users\_get\_response>

**JSON格式：**

{

"users\_get\_response": {

"users": {

"user ": [{

"user\_id": 10001,

"nick": "hz0799",

"sex": "m",

"location": {

"country": "中国",

"district": "西湖区"

}

},

{

"user\_id": 10001,

"nick": "hz0799",

"sex": "m",

"location": {

"country": "中国",

"district": "西湖区"

}

}]

}

}

}