

# iWebShop Api 接口开发文档

V4.7

# 目录

iWebShop Api 接口开发文档.....	1
Api 存在的意义.....	3
Api 存放位置.....	3
Api 开发说明.....	4
应用范围.....	6

## Api 存在的意义

iWebShop 商城系统中继承了强大的 api 接口功能，支持自主开发编写，不仅功能强大，复用性强，更可以把前端页面的设计人员从繁杂的 query 标签中解脱出来，更可以方便其他平台甚至是手机 app 进行调用。

Api 机制的存在极大的降低了开发模板的难度，可以让不懂任何程序的美工和前端设计人员轻松的获取到数据库中的数据，同时 api 接口也是程序封装的一个重要体现，更多复杂，通用型强的代码可以进行封装方便调用，开发人员只需要维护一份 api 定义即可，所以学习和掌握 api 是提高开发效率的一件利器！

iWebShop3.3 版本版本中 Api 接口还增加了 cache 属性，可以无缝对接 file 或者 memcache 等高级缓存接口，提高系统负载能力，增加并发量！

iWebShop4.7 版本支持在主题目录下的 config.php 中定义 API 标签，每个主题都有自己独立的 API 接口，代码结构更加松散，可用性，维护性更高！

## Api 存放位置

接口的主体由两部分组成：**【核心类调用】** + **【接口定义】**

第一部分核心类调用：

[/classes/api.php](#)

这是 api 的核心，也是程序中调用的总入口，调用任何 api 都必须是通过此文件中定义的 Api 类来进行使用，例如 api 接口：

```
$data = Api::run( 'getArtList', 参数1, 参数2...);
```

这样一句代码就可以调用到商城系统的前 10 篇文章，怎么样？是不是很方便？

第二部分接口定义：

[/classes/api/api\\_resource.php](#) (公共 API 接口) **【可以在任何地方调用】**

[/views/主题名称/config.php](#) 里面的 api 键中定义 (个性化主题模板 API 接口) **【只能在所属主题模板中调用】**

这里存放着 api 接口的配置和定义，2 种接口定义格式写法相同，要调用的 api 必须至少要在这 2 个文件中的一个里面进行声明才可以调用，这里算是 api 的一个路由，里面是以数组方式进行存储，如：

```
<?php
return array(
    'getGoodsList' => array('file' => 'goods.php','class'=> 'APIGoods',),
    'getArtList' => array('query' => array('name' => 'article','where' => 'vi
    'getRegimentList' => array('query' => array('name' => 'regiment','where' => 'i
    'getPromotionList'=> array('query' => array('name' => 'promotion as p','join'
    'getCommendNew' => array('query' => array('name' => 'commend_goods as co','j
    'getCommendPrice' => array('query' => array('name' => 'commend_goods as co','j
    'getCommendHot' => array('query' => array('name' => 'commend_goods as co','j
    'getCommendRecom' => array('query' => array('name' => 'commend_goods as co','j
    'getOrderDistributed' => array('query' => array('name' => 'order','where' => '
);
```

(图 1) 公共接口定义

```
1 <?php
2 return array(
3     'name' => '锦绣前程',
4     'author' => 'aircheng',
5     'time' => '2014/12/29 10:03:05',
6     'version' => '4.5',
7     'thumb' => 'preview.jpg',
8     'info' => 'aircheng旗下, IwebShop产品首款默认主题方案, 此主题适用于IwebShop4.5+系列
9     'type' => '商城前台-PC',
10    'ad' => array(
11        "页面顶部通栏广告条960*70(default)",
12        "首页中部通栏960*70(default)",
13        "首页右上方198*104(default)",
14        "文章-公告内容页左册198*120(default)",
15        "商品搜索结果页左侧198*120(default)",
16    ),
17    'api' => array(
18        'getSpecData' => array(
19            'query' => array(
20                'name' => 'goods as go,products as pro',
21                'where' => 'pro.id = #id# and pro.goods_id = go.id and go.is_del = 0',
22                'fields' => 'pro.sell_price,pro.weight,pro.id as product_id,pro.spec_array
23                'type' => 'row',
24            )
25        )
26    ),
27 );
```

(图 2) 主题模板个性化定义

我们看到所有的 api 名称都被列举出来，以数组 key 键的方式存在，然后在通过 Api 类进行调用。只有 api 接口在被定义后，才能在商城系统中进行调用。否则提示未定义错误。定义写法规格在后面有具体说明。

## Api 开发说明

我们有必要对 API 定义规则进行说明，可以让用户自己编写出更强大的接口，目前 api 支持 2 种方式的编写，即：query 类型和 file 类型。配置类型在具体的 api 中写在数组 key 键中。

### 1. query 类型

对于熟悉 iwebshop 的开发者来说，query 并不陌生，就是利用 IQuery 类来获取数据库中的数据，在模板中就是 {query:....} {/query} 而在 php 代码中就是 IQuery 类，这里 api 完全可以用 query 来获取到数据，用法和效果和模板标签 query 是保持一致的，比如：

```
'getArtList' => array('query' => array('name' => 'article', 'where' => 'visibility = 1 and top = 1', 'order' => 'sort ASC, id DESC', 'fields' => 'title, id, style, color', 'limit' => '10'));
```

对于这么一条 api 来说，我们就可以理解为 query 类型，其中的

name	数据表名称
where	条件子句
order	排序
fields	查询字段
limit	读取数量
page	分页数



file 类型的 api 把超高的自由度开放给了用户，用户可以自由编写自己的类，自由处理各种逻辑，实现各种需求。

```
'getGoodsList' => array('file' => 'goods.php', 'class' => 'APIGoods')
```

这个就是一个 file 类型的 api 接口，这个 getGoodsList 接口一共有 2 个参数，分别是 file 和 class。

file 参数：用户自己开发的 api 所存放的文件名称，注意这个文件必须存放到

`/classes/api/` 这个目录下才可以被系统读取到。这个也是存放 api 接口的目录。

class 参数：要执行的类名称，也就是说，这个 file 文件中必须要有 class 这个类，并且这个类中必须要有 api 接口名字，比如：`'getGoodsList'` class 类中的 getGoodsList 就是调用的入口。

```
10 class APIGoods
11 {
12     /**
13      * @brief 根据id获取商品信息
14      * @param int $gid 商品id,多个id时以逗号','分割, 如: 2,5,6,21
15      * @return array 商品结果集
16      */
17     public function getGoodsList($gid)
18     {
19         $fields = ' id , name , goods_no , sell_price , market_price , cost_price , store_nums , img , weight
20         $dbObj = IDBFactory::getDB();
21         $tableName = IWeb::$app->config['DB']['tablePre'].'goods';
22         return $dbObj->doSql('SELECT '.$fields.' FROM '.$tableName.' WHERE id in ( '.$gid.' ) AND is_del = 0 ');
23     }
24 }
25 }
```

只需要配置好参数以后，那么 api 被调用的时候就可以正常工作了，大体的工作流程就是内核找到 `goods.php` 这个文件，然后把 `APIGoods` 类进行实例化，最后再调用其中的 `getGoodsList` 方法。

当然，file 类型的接口参数依然是可以增加的，如图 getGoodsList 接口需要有 \$gid 参数，那么我们在调用时候直接在 `Api::run( 'getGoodsList' , ' 1,2,3' );` 追加上去即可。

## 应用范围

Api 接口大部分情况下用户 template 模板中，可以结合 `{foreach:}` 标签进行循环输出数据，也可以通过 `{set:}` 标签进行数据的暂时存储，使用方面比较灵活的，例如：

```
{foreach:items=Api::run( 'getGoodsList' , ' 1,2,3,4' )}
{$item[ 'name' ]}
{/foreach}
```

这样模板里面直接就可以输出 ID 为 1, 2, 3, 4 商品的名称。

```
{set:$apiData = Api::run( 'getGoodsList' , ' 1,2,3,4' )}
```

这样 \$apiData 变量中就存储着 ID 为 1, 2, 3, 4 商品的名称。

总之 api 的使用层面要只要记住 `Api::run( '接口名称' );` 就可以了！

我们在 iwebshop 系统中的 `/classes/api/` 中提供了部分 query 和 file 类型的实例，用户自己自由翻阅查看，如有问题欢迎到 <http://bbs.aircheng.com> 进行咨询留言，我们期待您的参与。