

API接口文档

网极科技



目 录

API接口文档概述

阿里云市场GET/POST方法

php版本

java版本

python3版本

短信验证码接口

在Epi51.cn购买短信接口说明

单发接口

群发接口

无参数单发接口

国家代码表

代码示例

【测试推荐】纯前端ajax代码

Python代码示例

php代码示例

java代码示例

curl代码示例

ObjectC代码示例

C#代码示例

在阿里云市场购买的短信接口说明

短信调用demo(ajax版)

前端代码

ajaxsms.php

php post示例代码

错误码

常用模板

返回json说明

频率限制

申请签名的常见问题

金融数据接口

产品说明

金融数据调用代码 (php)

金融数据调用代码(java)

金融数据返回结果说明

金融数据图表代码下载

贵金属代码表

大宗商品代码表

外汇代码表

数字货币数据

股指代码表
债券代码表
股指期货代码表
ETF代码表
快递查询接口
 快递公司代码表
 快递查询代码(PHP)
 快递查询返回结果
股票数据接口
 股票接口说明
 实时数据
 K线数据
 分时数据
 5日分时
 F10数据查询
 公司概况
 行情_排序
 沪深股票列表
 香港股票列表
 美国股票列表
 板块查询说明
 期货列表查询
 股票代码说明
 股票常见问题
 交易状态
 股票demo说明
外汇高频数据接口
图形验证码识别接口
关于调用次数的说明

API接口文档概述

目前已上架的api接口

详情请登录：[阿里云API云市场——网极科技店铺](#)

[查看阿里云GET/POST方法优化](#)

appcode获取方法

在阿里云市场购买相应的api接口后，登陆阿里云控制台，左侧菜单最下方有云市场，选择相应的api点击使用api，即可查看自己的appcode



The screenshot shows the Alibaba Cloud console interface. On the left, there is a navigation menu with '云市场' (Cloud Market) selected. The main content area displays '已购买的服务' (My Purchased Services) with a table of services. The table has columns for '商品名称' (Product Name), '状态' (Status), and '到期时间' (Expiration Time). The '金融数据接口' (Financial Data Interface) service is listed multiple times, and the '使用API' (Use API) button for one of these entries is highlighted with a red box. At the bottom right, there is a pagination control showing '共有5条，每页显示：20条' (Total 5 items, 20 items per page).

| 商品名称 | 状态 | 到期时间 | 操作 |
|-----------|-----|------|---|
| 阿里短信服务-按量 | 已开通 | - | 使用API 管理 |
| 金融数据接口 | 已开通 | - | 使用API 管理 再次购买 |
| 快递查询 | 已开通 | - | 使用API 管理 再次购买 |
| 金融数据接口 | 已开通 | - | 使用API 管理 再次购买 |
| 金融数据接口 | 已开通 | - | 使用API 管理 再次购买 |

阿里云市场GET/POST方法

[php版本](#)

[java版本](#)

[python3版本](#)

php版本

```
<?php
/*
优化阿里云默认提供代码示例
php 5.3以上版本 支持 curl

*/

$host = "http://smsapi.api51.cn";//如需https请修改为 https://smsapi51.cn
$path = "/single_sms/";//path为 single_sms_get 时为GET请求
$method = "1";//post=1 get=0
$appcode = "购买后可获得appcode";

$url = $host . $path;

$data = array(
    'mobile'=>'1328888888',//接收手机号
    'params'=>'22,123',//模板变量, 多个以英文逗号隔开
    'sign'=>'12',//登陆www.api51.cn申请成功的签名
    'tpl_id'=>'47331',//登陆www.api51.cn申请成功的模版ID, 我们提供常用模板 : http://
help.api51.cn/425602
);
$data = http_build_query($data);

$result = api51_curl($url,$data,$method,$appcode);
echo $result;

function api51_curl($url,$data=false,$ispost=0,$appcode){
    $headers = array();
    //根据阿里云要求, 定义Appcode
    array_push($headers, "Authorization:APPCODE " . $appcode);
    array_push($headers, "Content-Type".":". "application/x-www-form-urlencoded;
charset=UTF-8");

    $httpInfo = array();

    $ch = curl_init();
    curl_setopt( $ch, CURLOPT_HTTP_VERSION , CURL_HTTP_VERSION_1_1 );
    curl_setopt( $ch, CURLOPT_USERAGENT , 'api51.cn' );
    curl_setopt( $ch, CURLOPT_CONNECTTIMEOUT , 300 );
    curl_setopt( $ch, CURLOPT_TIMEOUT , 300);
    curl_setopt( $ch, CURLOPT_RETURNTRANSFER , true );
    curl_setopt($ch, CURLOPT_FOLLOWLOCATION, true);
    curl_setopt($ch, CURLOPT_HTTPHEADER, $headers);
    curl_setopt($ch, CURLOPT_HEADER, false);
    curl_setopt($ch, CURLOPT_FAILONERROR, false);
    if (1 == strpos("$".$url, "https://"))
    {
```

```
        curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);
        curl_setopt($ch, CURLOPT_SSL_VERIFYHOST, false);
    }
    if($ispost)
    {
        curl_setopt($ch, CURLOPT_CUSTOMREQUEST, 'POST');
        curl_setopt( $ch , CURLOPT_POST , true );
        curl_setopt( $ch , CURLOPT_POSTFIELDS , $data );
        curl_setopt( $ch , CURLOPT_URL , $url );
    }
    else
    {
        curl_setopt($ch, CURLOPT_CUSTOMREQUEST, 'GET');
        if($data){
            curl_setopt( $ch , CURLOPT_URL , $url.'?'.$data );

        }else{
            curl_setopt( $ch , CURLOPT_URL , $url);
        }
    }
    $response = curl_exec( $ch );

    if ($response === FALSE) {
        // echo "cURL Error: " . curl_error($ch);
        return false;
    }
    $statusCode = curl_getinfo( $ch , CURLINFO_HTTP_CODE );
    $httpInfo = array_merge( $httpInfo , curl_getinfo( $ch ) );
    curl_close( $ch );
    return $response;
}
?>
```

java版本

```
import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;
import java.io.PrintWriter;
import java.net.URL;
import java.net.URLConnection;
import java.util.List;
import java.util.Map;

public class chart {
    /**
     * 向指定URL发送GET方法的请求
     *
     * @param url
     *         发送请求的URL
     * @param param
     *         请求参数，请求参数应该是 name1=value1&name2=value2 的形式。
     * @return URL 所代表远程资源的响应结果
     */
    public static void main(String[] args) {
        //发送 GET 请求
        //实时数据
        String s=real.sendGet("http://finance.api51.cn/real", "en_prod_code=UKOIL"
    );

        System.out.println(s);
        //k线数据
        String s=chart.sendGet("http://finance.api51.cn/chart", "candle_period=1&dat
a_count=8&prod_code=UKOIL");
        System.out.println(s);
        /*
        //发送 POST 请求
        String sr=HttpRequest.sendPost("http://localhost:6144/Home/RequestPostString
", "key=123&v=456");
        System.out.println(sr);
        */
    }
    public static String sendGet(String url, String param) {
        String result = "";
        BufferedReader in = null;
        try {
            String urlNameString = url + "?" + param;
            URL realUrl = new URL(urlNameString);
            // 打开和URL之间的连接
            URLConnection connection = realUrl.openConnection();
            // 设置通用的请求属性
            connection.setRequestProperty("accept", "*/*");
```



```

connection.setRequestProperty("connection", "Keep-Alive");
connection.setRequestProperty("user-agent",
    "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1;SV1)");
connection.setRequestProperty("Authorization", "APPCODE "+"您的appcode");

// 建立实际的连接
connection.connect();
// 获取所有响应头字段
Map<String, List<String>> map = connection.getHeaderFields();
// 遍历所有的响应头字段
for (String key : map.keySet()) {
    // System.out.println(key + "--->" + map.get(key));
}
// 定义 BufferedReader输入流来读取URL的响应
in = new BufferedReader(new InputStreamReader(
    connection.getInputStream()));
String line;
while ((line = in.readLine()) != null) {
    result += line;
}
} catch (Exception e) {
    System.out.println("发送GET请求出现异常!" + e);
    e.printStackTrace();
}
// 使用finally块来关闭输入流
finally {
    try {
        if (in != null) {
            in.close();
        }
    } catch (Exception e2) {
        e2.printStackTrace();
    }
}
return result;
}

/**
 * 向指定 URL 发送POST方法的请求
 *
 * @param url
 *         发送请求的 URL
 * @param param
 *         请求参数，请求参数应该是 name1=value1&name2=value2 的形式。
 * @return 所代表远程资源的响应结果
 */
public static String sendPost(String url, String param) {
    PrintWriter out = null;
    BufferedReader in = null;
    String result = "";
    try {
        URL realUrl = new URL(url);

```

```
// 打开和URL之间的连接
URLConnection conn = realUrl.openConnection();
// 设置通用的请求属性
conn.setRequestProperty("accept", "*/*");
conn.setRequestProperty("connection", "Keep-Alive");
conn.setRequestProperty("user-agent",
    "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1;SV1)");
// 发送POST请求必须设置如下两行
conn.setDoOutput(true);
conn.setDoInput(true);
// 获取URLConnection对象对应的输出流
out = new PrintWriter(conn.getOutputStream());
// 发送请求参数
out.print(param);
// flush输出流的缓冲
out.flush();
// 定义BufferedReader输入流来读取URL的响应
in = new BufferedReader(
    new InputStreamReader(conn.getInputStream()));
String line;
while ((line = in.readLine()) != null) {
    result += line;
}
} catch (Exception e) {
    System.out.println("发送 POST 请求出现异常!" + e);
    e.printStackTrace();
}
//使用finally块来关闭输出流、输入流
finally{
    try{
        if(out!=null){
            out.close();
        }
        if(in!=null){
            in.close();
        }
    }
    catch(IOException ex){
        ex.printStackTrace();
    }
}
return result;
}
}
```

python3版本

```
import requests

headers = {"Authorization": "APPCODE 您的appcode"}
#定义请求地址
host = ' http://stock.api51.cn '
#定义路径
path = '/trend'
# 定义参数
querys = 'prod_code=000001.SS'
url = host + path + '?' + querys

datas = requests.get(url=url, headers=headers, timeout=15).content
print(datas)
```

短信验证码接口

短信接口说明

尊敬的客户您好：

我们短信可以在阿里云云市场购买使用，也可以直接在我们平台预存任意金额使用，国内短信每条四分，行业最低价。

注意：在阿里云购买与在我们平台购买使用，费用一致，但是接口不同，请仔细留意分辨

如果在阿里云市场购买

在阿里云市场购买相应的api接口后，登陆阿里云控制台，左侧菜单最下方有云市场，选择相应的api点击使用api，即可查看自己的appcode

申请签名和模板，需要登录我们平台 www.api51.cn 进行申请

当然您也可以用已经申请好的签名和模板，直接使用

如果在我们平台 www.api51.cn 购买

由于阿里云市场不方便支持申请短信签名与模版，有类似需要的客户请登陆API无忧官网 www.api51.cn 进行注册，并申请短信模版。

我们平台特色功能

- A、自定义签名与模版
- B、支持短信群发
- C、短信4分每条
- D、一元起充

在api51.cn购买短信接口说明

注意，此处文档是在我们平台 www.api51.cn 充值使用的接口，而非阿里云市场接口

API无忧介绍

西安网极网络科技有限公司成立于2014年并推出API服务。网极是一支年轻充满活力的创业团队；公司期望整合商业、科技和设计，让工作更简单快乐，成为受人尊重的企业服务商。

单发接口

注意，此处文档是在我们平台 www.api51.cn 充值使用的接口，而非阿里云市场接口

POST/GET请求

地址：<http://www.api51.cn/api/smsApi/sendcode>

如需https请修改为 <https://www.api51.cn>

参数说明

| 参数 | 说明 | 备注 |
|--------------|--------------------|---|
| token | 识别用户的唯一令牌 | 在 www.api51.cn 注册后可获得 |
| mobile | 手机号 | 如：180XXXXXXXX |
| sign | 短信签名 | 确保已经申请通过，默认发送：API 无忧 |
| tpl_id | 短信模版ID | 确保已经申请通过 |
| params | 短信模版变量 | 确保已经申请通过，多个以英文逗号隔开。如：张三,1234 |
| country_code | 国家代码，可选参数，默认中国（86） | 如：86 参考国家代码表： http://help.api51.cn/432250 |

返回示例：

发送成功

```
{
  "result": 0,
  "errmsg": "OK",
  "ext": "",
  "sid": "8:5Kong28XqC78HBEDn0y20171019",
  "fee": 1
}
```

发送失败

```
{
  "result": 1016,
  "errmsg": "手机号格式错误",
  "ext": ""
}
```

php代码示例

```
<?php
/*
优化阿里云默认提供代码示例
php 5.3以上版本 支持 curl
*/

$host = "http://www.api51.cn";//如需https请修改为 https://www.api51.cn
$path = "/api/smsApi/sendcode";//path为 single_sms_get 时为GET请求
$method = "0";//post=1 get=0 本接口都支持

$url = $host . $path;

$data = array(
  'token'=>'写你的token',
  'mobile'=>'1892584891',//接收手机号
  'params'=>'22,123',//模板变量, 多个以英文逗号隔开
  'sign'=>'API无忧',//登陆www.api51.cn申请成功的签名
  'tpl_id'=>'47331',//登陆www.api51.cn申请成功的模版ID, 我们提供常用模板 : http://
help.api51.cn/425602
);
$data = http_build_query($data);

$result = api51_curl($url,$data,$method);
echo $result;

function api51_curl($url,$data=false,$ispost=0){
  $headers = array();

  array_push($headers, "Content-Type".":". "application/x-www-form-urlencoded;
charset=UTF-8");

  $httpInfo = array();

  $ch = curl_init();
  curl_setopt( $ch, CURLOPT_HTTP_VERSION , CURL_HTTP_VERSION_1_1 );
  curl_setopt( $ch, CURLOPT_USERAGENT , 'api51.cn' );
  curl_setopt( $ch, CURLOPT_CONNECTTIMEOUT , 300 );
  curl_setopt( $ch, CURLOPT_TIMEOUT , 300);
  curl_setopt( $ch, CURLOPT_RETURNTRANSFER , true );
  curl_setopt($ch, CURLOPT_FOLLOWLOCATION, true);
```

```
curl_setopt($ch, CURLOPT_HTTPHEADER, $headers);
curl_setopt($ch, CURLOPT_HEADER, false);
curl_setopt($ch, CURLOPT_FAILONERROR, false);
if (1 == strpos("$".$url, "https://"))
{
    curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);
    curl_setopt($ch, CURLOPT_SSL_VERIFYHOST, false);
}
if($ispost)
{
    curl_setopt($ch, CURLOPT_CUSTOMREQUEST, 'POST');
    curl_setopt( $ch , CURLOPT_POST , true );
    curl_setopt( $ch , CURLOPT_POSTFIELDS , $data );
    curl_setopt( $ch , CURLOPT_URL , $url );
}
else
{
    curl_setopt($ch, CURLOPT_CUSTOMREQUEST, 'GET');
    if($data){
        curl_setopt( $ch , CURLOPT_URL , $url.'?'.$data );
    }else{
        curl_setopt( $ch , CURLOPT_URL , $url);
    }
}
$response = curl_exec( $ch );

if ($response === FALSE) {
    // echo "cURL Error: " . curl_error($ch);
    return false;
}
$httpCode = curl_getinfo( $ch , CURLINFO_HTTP_CODE );
$httpInfo = array_merge( $httpInfo , curl_getinfo( $ch ) );
curl_close( $ch );
return $response;
}
?>
```


群发接口

注意，此处文档是在我们平台 www.api51.cn 充值使用的接口，而非阿里云市场接口

POST请求

POST 地址：<http://www.api51.cn/api/smsApi/send>

POST 参数：

| 参数 | 说明 | 备注 |
|--------|-----------|---|
| token | 识别用户的唯一令牌 | 在www.api51.cn注册后可获得 |
| mobile | 手机号 | 可单个或多个，多个之间以英文逗号隔开，如： 180XXXXXXXX, 180XXXXXXXX |
| sign | 短信签名 | 确保已经申请通过，默认发送：API无忧 |
| tpl_id | 短信模版ID | 确保已经申请通过 |
| params | 短信模版变量 | 确保已经申请通过，多个以英文逗号隔开。如：张三,1234 |

注意：群发不支持国际短信

返回示例：

单条发送

```
{
  "code": 1,
  "msg": [
    {
      "result": 0, //发送成功
      "errmsg": "OK",
      "mobile": "18092584891",
      "nationcode": "86", //国家代码 默认86中国
      "sid": "8:vhsM04aihCoMt5tpB3P20171009", //短信ID 用于查询状态
      "fee": 1 //计费条数
    }
  ]
}
```

多条发送

```
{
  "code": 1,
  "msg": [
    {
      "result": 1016, //发送失败
      "errmsg": "error mobile phone number format", //失败原因
      "mobile": "180999999",
      "nationcode": "86"
    },
    {
      "result": 0,
      "errmsg": "OK",
      "mobile": "18092584891",
      "nationcode": "86",
      "sid": "8:Igm1SCmoTx8lqwpjniR20171009",
      "fee": 1
    }
  ]
}
```

无参数单发接口

注意，此处文档是在我们平台 www.api51.cn 充值使用的接口，而非阿里云市场接口

POST/GET请求

地址：<http://www.api51.cn/api/smsApi/sendmsg>

如需https请修改为 <https://www.api51.cn>

参数说明

| 参数 | 说明 | 备注 |
|--------------|---------------------|---|
| token | 用户令牌 | 在 www.api51.cn 注册后可获得 |
| mobile | 手机号 | 如：180XXXXXXXX |
| msg | 短信内容 | 确保已经申请通过，已申请成功的签名+模板内容。必须和模板分毫不差，标点符号多一个少一个都不行。 |
| type | 短信类型，可选参数，默认0（普通短信） | 0：普通短信；1：营销短信 |
| country_code | 国家代码，可选参数，默认中国（86） | 如：86 参考国家代码表： http://help.api51.cn/432250 |

返回示例：

发送成功

```
{
  "result": 0,
  "errmsg": "OK",
  "ext": "",
  "sid": "8:5K0mg28XqC78HBEDn0y20171019",
  "fee": 1
}
```

发送失败

```
{
  "result": 1016,
  "errmsg": "手机号格式错误",
  "ext": ""
}
```

php代码示例

```
<?php
/*
优化阿里云默认提供代码示例
php 5.3以上版本 支持 curl

*/

$host = "http://www.api51.cn";//如需https请修改为 https://www.api51.cn
$path = "/api/smsApi/sendmsg";//path为 single_sms_get 时为GET请求
$method = "0";//post=1 get=0 本接口都支持

$url = $host . $path;

$data = array(
    'token'=>'写你的token',
    'mobile'=>'1892584891',//接收手机号
    'msg'=>'【API无忧】您的验证码为1234',//短信签名+内容，必须与申请的分毫不差，大小写
    'sign'=>'API无忧',//登陆www.api51.cn申请成功的签名
    'type'=>'0',//0:普通短信，可选参数 可不传
);
$data = http_build_query($data);

$result = api51_curl($url,$data,$method);
echo $result;

function api51_curl($url,$data=false,$ispost=0){
    $headers = array();

    array_push($headers, "Content-Type".":". "application/x-www-form-urlencoded; charset=UTF-8");

    $httpInfo = array();

    $ch = curl_init();
    curl_setopt( $ch, CURLOPT_HTTP_VERSION , CURL_HTTP_VERSION_1_1 );
    curl_setopt( $ch, CURLOPT_USERAGENT , 'api51.cn' );
    curl_setopt( $ch, CURLOPT_CONNECTTIMEOUT , 300 );
```

```

curl_setopt( $ch, CURLOPT_TIMEOUT , 300);
curl_setopt( $ch, CURLOPT_RETURNTRANSFER , true );
curl_setopt($ch, CURLOPT_FOLLOWLOCATION, true);
curl_setopt($ch, CURLOPT_HTTPHEADER, $headers);
curl_setopt($ch, CURLOPT_HEADER, false);
    curl_setopt($ch, CURLOPT_FAILONERROR, false);
    if (1 == strpos("$".$url, "https://"))
    {
        curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);
        curl_setopt($ch, CURLOPT_SSL_VERIFYHOST, false);
    }
if($ispost)
{
    curl_setopt($ch, CURLOPT_CUSTOMREQUEST, 'POST');
    curl_setopt( $ch , CURLOPT_POST , true );
    curl_setopt( $ch , CURLOPT_POSTFIELDS , $data );
    curl_setopt( $ch , CURLOPT_URL , $url );
}
else
{
    curl_setopt($ch, CURLOPT_CUSTOMREQUEST, 'GET');
    if($data){
        curl_setopt( $ch , CURLOPT_URL , $url.'?'.$data );

    }else{
        curl_setopt( $ch , CURLOPT_URL , $url);
    }
}
$response = curl_exec( $ch );

if ($response === FALSE) {
    // echo "cURL Error: " . curl_error($ch);
    return false;
}
$httpCode = curl_getinfo( $ch , CURLINFO_HTTP_CODE );
$httpInfo = array_merge( $httpInfo , curl_getinfo( $ch ) );
curl_close( $ch );
return $response;
}
?>

```

国家代码表

国家代码表

代码表下载地址：http://www.api51.cn/sms_price.xls

代码示例

【测试推荐】 [纯前端ajax代码](#)

[Python代码示例](#)

[php代码示例](#)

[java代码示例](#)

[curl代码示例](#)

[ObjectC代码示例](#)

[C#代码示例](#)

【测试推荐】纯前端ajax代码

本代码复制到本地，直接用浏览器运行即可使用，仅作调试只用。实际上线，请讲TOKEN写入后端程序，以免盗用。

[点此网页打开](#)

```

<html>
<head>
  <meta charset="UTF-8">
<title>API接口测试 www.api51.cn</title>
</head>
<body>
<br/>
<h1>必填参数</h1>
<br/>
Token : <input type='text' id='token' value='' />用户令牌 在www.api51.cn注册后可获得<br>
手机号 : <input type='text' id='mobile' value='' />手机号 | 如 : 180XXXXXXXX<br>
国家 : <input type='text' id='country_code' value='86' />国家代码 | 参考国家代码表 : http://
/help.api51.cn/432250<br>
<br/>
<h1>方式一、无参数单发接口</h1>
<br/>

短信内容 : <input type='text' id='msg' value='【API无忧】您的验证码为1234' style='min-widt
h:400px' /><button onclick='sendmsg()'>发送</button>
<br/>确保已经申请通过，已申请成功的签名+模板内容。必须和模板分毫不差，标点符号多一个少一个都不行。如
您的签名为：API无忧，申请成功的模板为：您的验证码为{1}。那么此处的内容应为：【API无忧】您的验证码为
1234 <br>

<br>
<h1>方式二、指定模板ID发送接口</h1>
<br/>
短信签名 : <input type='text' id='sign' value='API无忧' />确保已经申请通过，默认发送：API无忧
<br>
模板ID : <input type='text' id='tpl_id' value='46792' />常用模板 : http://help.api51.cn/4
25602<br>
模板参数 : <input type='text' id='params' value='1234' />多个以英文逗号隔开。如：张三,1234<b
r>

<button onclick='sendSMS()'>发送</button>

<br/><br/><br/><br/>
API接口测试 www.api51.cn
<script src="https://cdn.bootcss.com/jquery/1.11.1/jquery.min.js"></script>
<script>

```

```
//无参数单发接口
function sendmsg(){
var mobile = $('#mobile').val();
var token = $('#token').val();
var msg = $('#msg').val();
var country_code = $('#country_code').val();
$.ajax({
    type: 'POST',
    url: 'http://www.api51.cn/api/smsApi/sendmsg',
    data: {
        'mobile': mobile,
        'msg': msg,
        'country_code': country_code,
        'token': token
    },
    dataType: "json",
    success: function(data){
        console.log(data)
        if(data.result==0){
            alert('发送成功');
        }else{
            alert(data.errmsg);
        }
    }
})
}

//指定模板ID发送接口
function sendSMS(){
var mobile = $('#mobile').val();
var token = $('#token').val();
var sign = $('#sign').val();
var tpl_id = $('#tpl_id').val();
var params = $('#params').val();
var country_code = $('#country_code').val();
$.ajax({
    type: 'POST',
    url: 'http://www.api51.cn/api/smsApi/sendcode',
    data: {
        'mobile': mobile,
        'tpl_id': tpl_id,
        'country_code': country_code,
        'params': params,
        'sign': sign,
        'token': token
    },
    dataType: "json",
    success: function(data){
        console.log(data)
        if(data.result==0){
```

```
        alert('发送成功');
    }else{
        alert(data.errmsg);
    }

    }

})
}
</script>
</body>
</html>
```

```
\n\n\n\n~~~\n ","options":{"plugins_host":null,"context":"ebook","base":"./","features":
["internal_link"],"style":""}
```

Python代码示例

```
import urllib, urllib2, sys

host = 'http://www.api51.cn'
path = '/api/smsApi/send'
method = 'POST'

querys = ''
bodys = {}
url = host + path

bodys['mobile'] = '13288888888'
bodys['params'] = '1024,张三'
bodys['sign'] = '网极科技'
bodys['token'] = 'token'
bodys['tpl_id'] = '46792'
post_data = urllib.urlencode(bodys)
request = urllib2.Request(url, post_data)

//根据API的要求, 定义相对应的Content-Type
request.add_header('Content-Type', 'application/x-www-form-urlencoded; charset=UTF-8')
response = urllib2.urlopen(request)
content = response.read()
if (content):
    print(content)
```

php代码示例

```
<?php
    $host = "http://www.api51.cn";
    $path = "/api/smsApi/send";
    $method = "POST";

    $headers = array();

    //根据API的要求, 定义相对应的Content-Type
    array_push($headers, "Content-Type".":". "application/x-www-form-urlencoded; char
set=UTF-8");
    $querys = "";
    $bodys = "mobile=13288888888&params=1024%2C%E5%BC%A0%E4%B8%89&sign=%E7%BD%91%E6%
9E%81%E7%A7%91%E6%8A%80&token=token&tpl_id=46792";
    $url = $host . $path;

    $curl = curl_init();
    curl_setopt($curl, CURLOPT_CUSTOMREQUEST, $method);
    curl_setopt($curl, CURLOPT_URL, $url);
    curl_setopt($curl, CURLOPT_HTTPHEADER, $headers);
    curl_setopt($curl, CURLOPT_FAILONERROR, false);
    curl_setopt($curl, CURLOPT_RETURNTRANSFER, true);
    curl_setopt($curl, CURLOPT_HEADER, true);
    if (1 == strpos("$".$host, "https://"))
    {
        curl_setopt($curl, CURLOPT_SSL_VERIFYPEER, false);
        curl_setopt($curl, CURLOPT_SSL_VERIFYHOST, false);
    }
    curl_setopt($curl, CURLOPT_POSTFIELDS, $bodys);
    var_dump(curl_exec($curl));
?>
```

java代码示例

```
public static void main(String[] args) {
    String host = "http://www.api51.cn";
    String path = "/api/smsApi/send";
    String method = "POST";
    String appcode = "你自己的AppCode";
    Map<String, String> headers = new HashMap<String, String>();

    //根据API的要求, 定义相对应的Content-Type
    headers.put("Content-Type", "application/x-www-form-urlencoded; charset=UTF-8");

    Map<String, String> querys = new HashMap<String, String>();
    Map<String, String> bodys = new HashMap<String, String>();
    bodys.put("mobile", "13288888888");
    bodys.put("params", "1024,张三");
    bodys.put("sign", "网极科技");
    bodys.put("token", "token");
    bodys.put("tpl_id", "46792");

    try {
        /**
         * 重要提示如下:
         * HttpUtils请从
         * https://github.com/aliyun/api-gateway-demo-sign-java/blob/master/src/main/java/com/aliyun/api/gateway/demo/util/HttpUtils.java
         * 下载
         *
         * 相应的依赖请参照
         * https://github.com/aliyun/api-gateway-demo-sign-java/blob/master/pom.xml
         */
        HttpResponse response = HttpUtils.doPost(host, path, method, headers, querys, bodys);
        System.out.println(response.toString());
        //获取response的body
        //System.out.println(EntityUtils.toString(response.getEntity()));
    } catch (Exception e) {
        e.printStackTrace();
    }
}
```

curl代码示例

```
curl -i -X POST 'http://www.api51.cn/api/smsApi/send' --data 'mobile=13288888888&params=1024%2C%E5%BC%A0%E4%B8%89&sign=%E7%BD%91%E6%9E%81%E7%A7%91%E6%8A%80&token=token&tpl_id=46792'
```


ObjectC代码示例

```
NSString *host = @"http://www.api51.cn";
NSString *path = @"/api/smsApi/send";
NSString *method = @"POST";
NSString *querys = @"";
NSString *url = [NSString stringWithFormat:@"%@@%@", host, path, querys];
NSString *bodys = @"mobile=13288888888&params=1024%2C%E5%BC%A0%E4%B8%89&sign=%E7%BD%91%E6%9E%81%E7%A7%91%E6%8A%80&token=token&tpl_id=46792";

NSMutableURLRequest *request = [NSMutableURLRequest requestWithURL:[NSURL URLWithString:url] cachePolicy:1 timeoutInterval: 5];
request.HTTPMethod = method;

[request addValue: @"application/x-www-form-urlencoded; charset=UTF-8" forHTTPHeaderField: @"Content-Type"];
NSData *data = [bodys dataUsingEncoding: NSUTF8StringEncoding];
[request setHTTPBody: data];
NSURLSession *requestSession = [NSURLSession sessionWithConfiguration:[NSURLSessionConfiguration defaultSessionConfiguration]];
NSURLSessionDataTask *task = [requestSession dataTaskWithRequest:request
    completionHandler:^(NSData * _Nullable body, NSURLResponse * _Nullable response, NSError * _Nullable error) {
    NSLog(@"Response object: %@", response);
    NSString *bodyString = [[NSString alloc] initWithData:body encoding:NSUTF8StringEncoding];

    //打印应答中的body
    NSLog(@"Response body: %@", bodyString);
}];

[task resume];
```

C#代码示例

```
//using System.IO;
//using System.Text;
//using System.Net;
//using System.Net.Security;
//using System.Security.Cryptography.X509Certificates;

private const String host = "http://www.api51.cn";
private const String path = "/api/smsApi/send";
private const String method = "POST";

static void Main(string[] args)
{
    String querys = "";
    String bodys = "mobile=13288888888&params=1024%2C%E5%BC%A0%E4%B8%89&sign
=%E7%BD%91%E6%9E%81%E7%A7%91%E6%8A%80&token=token&tpl_id=46792";
    String url = host + path;
    HttpWebRequest httpRequest = null;
    HttpWebResponse httpResponse = null;

    if (0 < querys.Length)
    {
        url = url + "?" + querys;
    }

    if (host.Contains("https://"))
    {
        ServicePointManager.ServerCertificateValidationCallback = new Remote
CertificateValidationCallback(CheckValidationResult);
        httpRequest = (HttpWebRequest)WebRequest.CreateDefault(new Uri(url))
;
    }
    else
    {
        httpRequest = (HttpWebRequest)WebRequest.Create(url);
    }
    httpRequest.Method = method;

    //根据API的要求, 定义相对应的Content-Type
    httpRequest.ContentType = "application/x-www-form-urlencoded; charset=UT
F-8";

    if (0 < bodys.Length)
    {
        byte[] data = Encoding.UTF8.GetBytes(bodys);
        using (Stream stream = httpRequest.GetRequestStream())
        {
            stream.Write(data, 0, data.Length);
        }
    }
}
```

```
    }
    try
    {
        httpResponse = (HttpWebResponse)httpRequest.GetResponse();
    }
    catch (WebException ex)
    {
        httpResponse = (HttpWebResponse)ex.Response;
    }

    Console.WriteLine(httpResponse.StatusCode);
    Console.WriteLine(httpResponse.Method);
    Console.WriteLine(httpResponse.Headers);
    Stream st = httpResponse.GetResponseStream();
    StreamReader reader = new StreamReader(st, Encoding.GetEncoding("utf-8"))
);

    Console.WriteLine(reader.ReadToEnd());
    Console.WriteLine("\n");

}

public static bool CheckValidationResult(object sender, X509Certificate certificate, X509Chain chain, SslPolicyErrors errors)
{
    return true;
}
```

在阿里云市场购买的短信接口说明

注意，此处文档是在阿里云市场购买的短信接口

POST/GET请求

POST 地址：http://smsapi.api51.cn/single_sms/

GET 地址：http://smsapi.api51.cn/single_sms_get/

| 参数 | 说明 | 备注 |
|---------|-----------|---------------------------------|
| appcode | 识别用户的唯一令牌 | 在阿里云购买后可获得 购买地址 |
| mobile | 手机号 | 如：180XXXXXXXX |
| sign | 短信签名 | 确保已经申请通过，默认发送：API无忧 |
| tpl_id | 短信模版ID | 确保已经申请通过 |
| params | 短信模版变量 | 确保已经申请通过，多个以英文逗号隔开。如：张三,1234 |

常用模版

备注：{1}{2}为参数params字段顺序，请将params按英文逗号分开，如params="张三,1234"

如需自定义签名与模版 请登录 www.api51.cn

| 模版ID | 模版内容 |
|-------|---------------------------------------|
| 46792 | 您的验证码为{1} |
| 47331 | 尊敬的{1}，您的验证码为{2} |
| 49011 | 验证码{1}，您正在进行{2}身份验证，请不要告诉别人哦！ |
| 49014 | 验证码{1}，您正在登录{2}，若非本人操作请忽略。 |
| 49016 | 验证码{1}，您正在注册成为{2}用户，感谢您的支持！若非本人操作请忽略。 |

| | |
|-------|--|
| 49019 | 验证码{1}，您正在尝试修改{2}登录密码，请妥善保管账户信息。若非本人操作请忽略。 |
| 49020 | 验证码{1}，您正在尝试变更{2}重要信息，请妥善保管账户信息。如非本人操作请忽略。 |
| 49023 | 亲爱的{1}，您的验证码是{2}。有效期为{3}分钟，请尽快验证 |
| 46795 | 尊敬的用户，我们的{1}服务升级啦，详情请联系您的客服{2}。回T退订 |

短信发送错误码

错误码 原因 解决方案

1002 短信/语音内容中含有敏感词 应答包会返回该敏感词，请去掉该敏感词后重新发送

1012 签名格式错误或者签名未审批 签名只能由中英文、数字组成，要求2~8个字;如果符合签名格式规范，请核查签名是否已审批

1014 模版未审批或请求的内容与审核通过的模版内容不匹配

1015 手机号在黑名单库中,通常是用户退订或者命中运营商黑名单导致的

1016 手机号格式错误 请核查下发短信/语音的用户手机号格式是否正确

1017 请求的短信内容太长 短信内容长度超过限制，请调整短信长度

1018 语音验证码格式错误 请遵守API接口说明规范

短信调用demo(ajax版)

前端代码

[ajaxsms.php](#)

前端代码

前端代码

```

<html>
<body>
手机号:<input type='text' id='mobile' value='' /><br>
短信签名:<input type='text' id='sign' value='' />登陆www.api51.cn申请成功的签名<br>
模板ID:<input type='text' id='tpl_id' value='' />登陆www.api51.cn申请成功的模版ID<br>
模版参数:<input type='text' id='params' value='' /><br>

<button onclick='sendSMS()'>发送</button>
<script src="https://cdn.bootcss.com/jquery/1.11.1/jquery.min.js"></script>
<script>
function sendSMS(){
var mobile = $('#mobile').val();
var sign = $('#sign').val();
var tpl_id = $('#tpl_id').val();
var params = $('#params').val();

$.ajax({
  type:'POST',
  url:'ajaxsms.php',
  data:{
    'mobile':mobile,
    'sign':sign,
    'tpl_id':tpl_id,
    'params':params
  },
  dataType: "json",
  success: function(data){

    if(data.result==0){
      alert('发送成功');
    }else{
      alert(data.errmsg);
    }
    console.log(data)
  }
})
}
</script>
</body>
</html>

```

\n\n\n\n~::~", "options": {"plugins_host": null, "context": "ebook", "base": "./", "features": ["internal_link"], "style": ""}

ajaxsms.php

```
<?php
/*
优化阿里云默认提供代码示例
php 5.3以上版本 支持 curl

*/

$host = "http://smsapi.api51.cn";//如需https请修改为 https://smsapi51.cn
$path = "/single_sms/";//path为 single_sms_get 时为GET请求
$method = "1";//post=1 get=0
$appcode = "购买后可获得appcode";

$url = $host . $path;

$data = array(
    'mobile'=>$_POST['mobile'],//接收手机号
    'params'=>$_POST['params'],//模板变量, 多个以英文逗号隔开
    'sign'=>$_POST['sign'],//登陆www.api51.cn申请成功的签名
    'tpl_id'=>$_POST['tpl_id'],//登陆www.api51.cn申请成功的模版ID, 我们提供常用模
板 : http://help.api51.cn/425602
);
$data = http_build_query($data);

$result = api51_curl($url,$data,$method,$appcode);
echo $result;

function api51_curl($url,$data=false,$ispost=0,$appcode){
    $headers = array();
    //根据阿里云要求, 定义Appcode
    array_push($headers, "Authorization:APPCODE " . $appcode);
    array_push($headers, "Content-Type".":". "application/x-www-form-urlencoded;
charset=UTF-8");

    $httpInfo = array();

    $ch = curl_init();
    curl_setopt( $ch, CURLOPT_HTTP_VERSION , CURL_HTTP_VERSION_1_1 );
    curl_setopt( $ch, CURLOPT_USERAGENT , 'api51.cn' );
    curl_setopt( $ch, CURLOPT_CONNECTTIMEOUT , 300 );
    curl_setopt( $ch, CURLOPT_TIMEOUT , 300);
    curl_setopt( $ch, CURLOPT_RETURNTRANSFER , true );
    curl_setopt($ch, CURLOPT_FOLLOWLOCATION, true);
    curl_setopt($ch, CURLOPT_HTTPHEADER, $headers);
    curl_setopt($ch, CURLOPT_HEADER, false);
    curl_setopt($ch, CURLOPT_FAILONERROR, false);
    if (1 == strpos("$".$url, "https://"))
    {
```



```
        curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);
        curl_setopt($ch, CURLOPT_SSL_VERIFYHOST, false);
    }
    if($ispost)
    {
        curl_setopt($ch, CURLOPT_CUSTOMREQUEST, 'POST');
        curl_setopt( $ch , CURLOPT_POST , true );
        curl_setopt( $ch , CURLOPT_POSTFIELDS , $data );
        curl_setopt( $ch , CURLOPT_URL , $url );
    }
    else
    {
        curl_setopt($ch, CURLOPT_CUSTOMREQUEST, 'GET');
        if($data){
            curl_setopt( $ch , CURLOPT_URL , $url.'?'.$data );

        }else{
            curl_setopt( $ch , CURLOPT_URL , $url);
        }
    }
    $response = curl_exec( $ch );

    if ($response === FALSE) {
        // echo "CURL Error: " . curl_error($ch);
        return false;
    }
    $httpCode = curl_getinfo( $ch , CURLINFO_HTTP_CODE );
    $httpInfo = array_merge( $httpInfo , curl_getinfo( $ch ) );
    curl_close( $ch );
    return $response;
}
?>
```

php post示例代码

```
<?php
/*
优化阿里云默认提供代码示例
php 5.3以上版本 支持 curl

*/

$host = "http://smsapi.api51.cn";//如需https请修改为 https://smsapi51.cn
$path = "/single_sms/";//path为 single_sms_get 时为GET请求
$method = "1";//post=1 get=0
$appcode = "购买后可获得appcode";

$url = $host . $path;

$data = array(
    'mobile'=>'1328888888',//接收手机号
    'params'=>'22,123',//模板变量, 多个以英文逗号隔开
    'sign'=>'12',//登陆www.api51.cn申请成功的签名
    'tpl_id'=>'47331',//登陆www.api51.cn申请成功的模版ID, 我们提供常用模板 : http://
help.api51.cn/425602
);
$data = http_build_query($data);

$result = api51_curl($url,$data,$method,$appcode);
echo $result;

function api51_curl($url,$data=false,$ispost=0,$appcode){
    $headers = array();
    //根据阿里云要求, 定义Appcode
    array_push($headers, "Authorization:APPCODE " . $appcode);
    array_push($headers, "Content-Type".":". "application/x-www-form-urlencoded;
charset=UTF-8");

    $httpInfo = array();

    $ch = curl_init();
    curl_setopt( $ch, CURLOPT_HTTP_VERSION , CURL_HTTP_VERSION_1_1 );
    curl_setopt( $ch, CURLOPT_USERAGENT , 'api51.cn' );
    curl_setopt( $ch, CURLOPT_CONNECTTIMEOUT , 300 );
    curl_setopt( $ch, CURLOPT_TIMEOUT , 300);
    curl_setopt( $ch, CURLOPT_RETURNTRANSFER , true );
    curl_setopt($ch, CURLOPT_FOLLOWLOCATION, true);
    curl_setopt($ch, CURLOPT_HTTPHEADER, $headers);
    curl_setopt($ch, CURLOPT_HEADER, false);
    curl_setopt($ch, CURLOPT_FAILONERROR, false);
    if (1 == strpos("$".$url, "https://"))
    {
```

```
        curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);
        curl_setopt($ch, CURLOPT_SSL_VERIFYHOST, false);
    }
    if($ispost)
    {
        curl_setopt($ch, CURLOPT_CUSTOMREQUEST, 'POST');
        curl_setopt( $ch , CURLOPT_POST , true );
        curl_setopt( $ch , CURLOPT_POSTFIELDS , $data );
        curl_setopt( $ch , CURLOPT_URL , $url );
    }
    else
    {
        curl_setopt($ch, CURLOPT_CUSTOMREQUEST, 'GET');
        if($data){
            curl_setopt( $ch , CURLOPT_URL , $url.'?'.$data );

        }else{
            curl_setopt( $ch , CURLOPT_URL , $url);
        }
    }
    $response = curl_exec( $ch );

    if ($response === FALSE) {
        // echo "cURL Error: " . curl_error($ch);
        return false;
    }
    $httpCode = curl_getinfo( $ch , CURLINFO_HTTP_CODE );
    $httpInfo = array_merge( $httpInfo , curl_getinfo( $ch ) );
    curl_close( $ch );
    return $response;
}
?>
```

错误码

短信发送错误码

| 错误码 | 原因 | 解决方案 |
|------|--------------------------|---|
| 1002 | 短信/语音内容中含有敏感词 | 应答包会返回该敏感词，请去掉该敏感词后重新发送 |
| 1012 | 签名格式错误或者签名未审批 | 签名只能由中英文、数字组成，要求2~8个字;如果符合签名格式规范，请核查签名是否已审批 |
| 1014 | 模版未审批或请求的内容与审核通过的模版内容不匹配 | 核对模板参数内容 |
| 1015 | 手机号在黑名单库中 | 通常是用户退订或者命中运营商黑名单导致的 |
| 1016 | 手机号格式错误 | 请核查下发短信/语音的用户手机号格式是否正确 |
| 1017 | 请求的短信内容太长 | 短信内容长度超过限制，请调整短信长度 |
| 1018 | 语音验证码格式错误 | 请遵守API接口说明规范 |
| 1024 | 发送频率超限 | 详见频率限制介绍 |

常用模板

常用模版

备注：{1}{2}为参数params字段顺序，请将params按英文逗号分开，如params="张三,1234"

如需自定义签名与模版 请登录 www.api51.cn

| 模版ID | 模版内容 |
|-------|--|
| 46792 | 您的验证码为{1} |
| 47331 | 尊敬的{1}，您的验证码为{2} |
| 49011 | 验证码{1}，您正在进行{2}身份验证，请不要告诉别人哦！ |
| 49014 | 验证码{1}，您正在登录{2}，若非本人操作请忽略。 |
| 49016 | 验证码{1}，您正在注册成为{2}用户，感谢您的支持！若非本人操作请忽略。 |
| 49019 | 验证码{1}，您正在尝试修改{2}登录密码，请妥善保管账户信息。若非本人操作请忽略。 |
| 49020 | 验证码{1}，您正在尝试变更{2}重要信息，请妥善保管账户信息。如非本人操作请忽略。 |
| 49023 | 亲爱的{1}，您的验证码是{2}。有效期为{3}分钟，请尽快验证 |
| 46795 | 尊敬的用户，我们的{1}服务升级啦，详情请联系您的客服{2}。回T退订 |

返回json说明

```
{  
  "result": 0, //0表示成功(计费依据), 非0表示失败  
  "errmsg": "OK", //result非0时的具体错误信息  
  "ext": "", //拓展字段, 暂未使用  
  "sid": "xxxxxxx", //标识本次发送id, 标识一次短信下发记录  
  "fee": 1 //短信计费的条数  
}
```

频率限制

频率限制

为了保障业务和通道安全，减少业务被刷后的经济损失，短信默认的频率限制策略为：

- 1、对同一个手机号，30秒内发送短信条数不超过3条
- 2、对同一个手机号，1小时内发送短信条数不超过5条
- 3、对同一个手机号，1个自然日内发送短信条数不超过15条
- 4、相同内容短信对同一个手机号，10秒内发送条数不超过1条

申请签名的常见问题

申请短信签名

签名是什么

短信签名是为客户提供一种快捷、方便的个性化签名方式。当客户发送短信时，短信平台会根据客户的设置，在短信内容里附加个性化签名，再发送给被叫手机客户。

签名示例

1、入驻企业为：西安网极网络科技有限公司

可以提交的签名为：【网极网络】

（上传资质必须为公司营业执照）

或者可以提交自己公司旗下的网站/公众号/APP签名：

上传资质必须为公司营业执照，

（1）网站类签名需在申请原因中详细说明网站地址及备案信息，签名必须与网站备案名称一致，可为简称，不可添加字段；

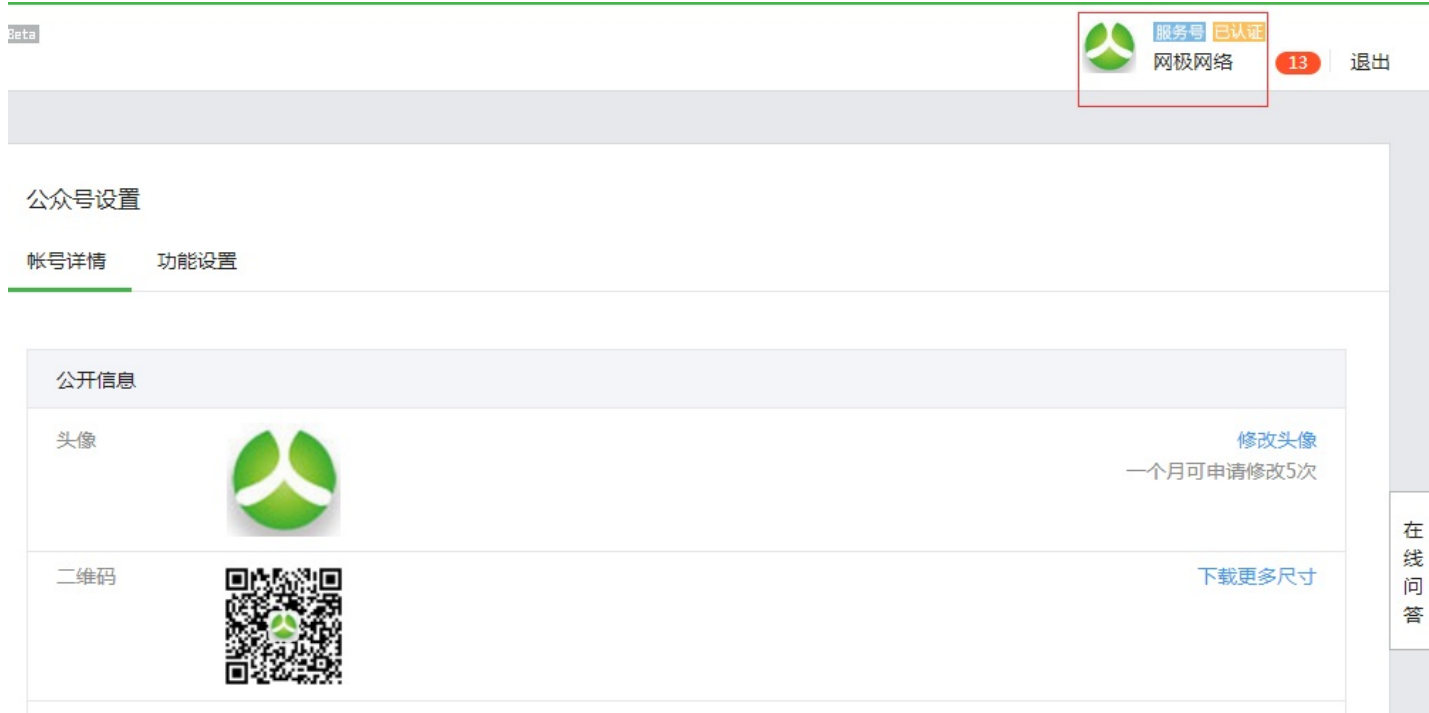
网站备案名称如：



签名可为【API无忧】

（2）公众号类签名需在申请原因中填写微信公众号，签名必须与公众号名称一致，可为简称，不可添加字段；

公众号名称如：



签名可为【网极科技】

(3) APP类签名需在申请原因中填写APP名称及下载地址，签名必须与APP名称一致，可为简称，不可添加字段

APP名称如：



下载地址为：<http://app.qq.com/#id=detail&appid=100317189>

签名可为【辣妈帮】

2、入驻为个人

直接在资质栏上传网站备案截图、公众号截图、APP截图等。

创建签名

1、通过控制台左侧导航，点击“短信接口”的“自定义签名”，进入签名管理页



2、点击“新增签名”进入创建签名页，根据用途上传相应证明材料

申请签名须知：

1、公司申请签名须知：

- (1) 签名为公司名称：请在申请原因中填写已备案官网地址并上传营业执照
- (2) 签名为微信公众号：请在申请原因中填写微信公众号名称并上传营业执照
- (3) 签名为产品名称：请在申请原因中填写产品体验地址并上传营业执照
- (4) 其它：请在申请原因中填写其它可证明签名归属的说明并上传营业执照

2、个人申请签名须知：

- (1) 签名类型为网站，必须为工信部备案的网站名称的全称或简称并提供域名备案截图；
- (2) 签名类型为公众号或小程序，必须为公众号（小程序）全称或简称并提供公众号（小程序）微信开放平台截图；
- (3) 签名类型为APP，必须为APP的全称或简称并提供任一应用商店的下载链接与该应用商店的后台管理截图；
- (4) 满足任一类型即可，请按照相关签名类型提交。

注意事项：

- 1、签名长度为2~ 8 个字，由中英文、数字组成，内容不包含【】；
- 2、签名内容必须与所提供资质相同，可为简称，不可添加字段；
- 3、公司申请必须上传营业执照；
- 4、网站类型签名必须与网站备案名称一致，可为简称，不可添加字段；

金融数据接口

[产品说明](#)

[金融数据调用代码 \(php \)](#)

[金融数据调用代码\(java\)](#)

[金融数据返回结果说明](#)

[金融数据图表代码下载](#)

[贵金属代码表](#)

[大宗商品代码表](#)

[外汇代码表](#)

[数字货币数据](#)

[股指代码表](#)

[债券代码表](#)

[股指期货代码表](#)

[ETF代码表](#)

产品说明

演示地址

<http://demo.api51.cn>

购买地址

[阿里云API市场](#)

产品说明

本接口可获取包括外汇、股票、股指、贵金属、大宗商品、债券等数据，数据包括实时数据，与分时图K线图数据，分为两个接口。

声明

1. 本接口所引用资料均来源于互联网，版权归原作者所有，如有侵权，请联系我们，将立即改正。
2. 本接口数据仅供参考，不构成任何投资建议，用户据此操作造成任何风险及后果，均由用户自行负责。
3. 未经协议授权，任何人（包括媒体、网站、公司、其他组织、个人）不得转载、引用、摘编、盗链本接口资源，任何人不得复制或仿造本接口在非网极科技所属的服务器上建立镜像，以及利用其它方式使用接口数据等，否则我们将追究其法律责任。

金融数据调用代码 (php)

```

<?php
// +-----
//wangjikeji.com
// 数据调用示例代码 - 网极科技

error_reporting(0);
header('Content-Type: text/html; charset=UTF-8');
$url = "http://finance.api51.cn/real?en_prod_code=000001";//请求地址, 可携带多个产品代码
, 用英文逗号隔开。如000001,000002

$content = wangji_curl($url);
echo $content;
function wangji_curl($url,$params=false,$ispost=0){
    $httpInfo = array();
    $ch = curl_init();
    $appcode = "*****";//购买后可获得 购买后在阿里云控制台 云市场 查看
    $headers = array();
    array_push($headers, "Authorization:APPCODE " . $appcode);
    curl_setopt( $ch, CURLOPT_HTTP_VERSION , CURL_HTTP_VERSION_1_1 );
    curl_setopt( $ch, CURLOPT_USERAGENT , 'JuheData' );
    curl_setopt( $ch, CURLOPT_CONNECTTIMEOUT , 60 );
    curl_setopt( $ch, CURLOPT_TIMEOUT , 60);
    curl_setopt( $ch, CURLOPT_RETURNTRANSFER , true );
    curl_setopt($ch, CURLOPT_FOLLOWLOCATION, true);
    if( $ispost )
    {
        curl_setopt( $ch , CURLOPT_POST , true );
        curl_setopt( $ch , CURLOPT_POSTFIELDS , $params );
        curl_setopt( $ch , CURLOPT_URL , $url );
    }
    else
    {
        if($params){
            curl_setopt( $ch , CURLOPT_URL , $url.'?'.$params );
        }else{
            curl_setopt( $ch , CURLOPT_URL , $url);
        }
        curl_setopt($ch, CURLOPT_HTTPHEADER, $headers);
    }
    $response = curl_exec( $ch );
    if ($response === FALSE) {
        //echo "CURL Error: " . curl_error($ch);
        return false;
    }
    $httpCode = curl_getinfo( $ch , CURLINFO_HTTP_CODE );
    $httpInfo = array_merge( $httpInfo , curl_getinfo( $ch ) );
    curl_close( $ch );
    return $response;
}

```

```
}

/*返回数据举例
{
  "code": 200,状态码
  "data": {
    "snapshot": {
      "000001": [ 产品代码
        "上证综指", 产品名称
        3217.96, 最新价格
        5.52, 涨跌幅
        0.17179999999999998, 涨跌额
        3219.52, 最高价格
        3195.29, 最低价格
        3212.44, 今日开盘价
        3212.44, 昨日收盘价
        1499412603 数据时间
      ],
      "fields": [
        "prod_name",
        "last_px",
        "px_change",
        "px_change_rate",
        "high_px",
        "low_px",
        "open_px",
        "preclose_px",
        "update_time"
      ]
    }
  }
}

*/
```

金融数据调用代码(java)

```
import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;
import java.io.PrintWriter;
import java.net.URL;
import java.net.URLConnection;
import java.util.List;
import java.util.Map;

public class chart {
    /**
     * 向指定URL发送GET方法的请求
     *
     * @param url
     *         发送请求的URL
     * @param param
     *         请求参数, 请求参数应该是 name1=value1&name2=value2 的形式。
     * @return URL 所代表远程资源的响应结果
     */
    public static void main(String[] args) {
        //发送 GET 请求
        //实时数据
        String s=real.sendGet("http://finance.api51.cn/real", "en_prod_code=UKOIL"
    );
        System.out.println(s);
        //k线数据
        String s=chart.sendGet("http://finance.api51.cn/chart", "candle_period=1&dat
a_count=8&prod_code=UKOIL");
        System.out.println(s);
        /*
        //发送 POST 请求
        String sr=HttpRequest.sendPost("http://localhost:6144/Home/RequestPostString
", "key=123&v=456");
        System.out.println(sr);
        */
    }
    public static String sendGet(String url, String param) {
        String result = "";
        BufferedReader in = null;
        try {
            String urlNameString = url + "?" + param;
            URL realUrl = new URL(urlNameString);
            // 打开和URL之间的连接
            URLConnection connection = realUrl.openConnection();
            // 设置通用的请求属性
            connection.setRequestProperty("accept", "*/*");
```

```

connection.setRequestProperty("connection", "Keep-Alive");
connection.setRequestProperty("user-agent",
    "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1;SV1)");
connection.setRequestProperty("Authorization", "APPCODE "+"您的appcode");

// 建立实际的连接
connection.connect();
// 获取所有响应头字段
Map<String, List<String>> map = connection.getHeaderFields();
// 遍历所有的响应头字段
for (String key : map.keySet()) {
    // System.out.println(key + "--->" + map.get(key));
}
// 定义 BufferedReader输入流来读取URL的响应
in = new BufferedReader(new InputStreamReader(
    connection.getInputStream()));
String line;
while ((line = in.readLine()) != null) {
    result += line;
}
} catch (Exception e) {
    System.out.println("发送GET请求出现异常!" + e);
    e.printStackTrace();
}
// 使用finally块来关闭输入流
finally {
    try {
        if (in != null) {
            in.close();
        }
    } catch (Exception e2) {
        e2.printStackTrace();
    }
}
return result;
}

/**
 * 向指定 URL 发送POST方法的请求
 *
 * @param url
 *         发送请求的 URL
 * @param param
 *         请求参数, 请求参数应该是 name1=value1&name2=value2 的形式。
 * @return 所代表远程资源的响应结果
 */
public static String sendPost(String url, String param) {
    PrintWriter out = null;
    BufferedReader in = null;
    String result = "";
    try {
        URL realUrl = new URL(url);

```



```
// 打开和URL之间的连接
URLConnection conn = realUrl.openConnection();
// 设置通用的请求属性
conn.setRequestProperty("accept", "*/*");
conn.setRequestProperty("connection", "Keep-Alive");
conn.setRequestProperty("user-agent",
    "Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1;SV1)");
// 发送POST请求必须设置如下两行
conn.setDoOutput(true);
conn.setDoInput(true);
// 获取URLConnection对象对应的输出流
out = new PrintWriter(conn.getOutputStream());
// 发送请求参数
out.print(param);
// flush输出流的缓冲
out.flush();
// 定义BufferedReader输入流来读取URL的响应
in = new BufferedReader(
    new InputStreamReader(conn.getInputStream()));
String line;
while ((line = in.readLine()) != null) {
    result += line;
}
} catch (Exception e) {
    System.out.println("发送 POST 请求出现异常!" + e);
    e.printStackTrace();
}
//使用finally块来关闭输出流、输入流
finally{
    try{
        if(out!=null){
            out.close();
        }
        if(in!=null){
            in.close();
        }
    }
    catch(IOException ex){
        ex.printStackTrace();
    }
}
return result;
}
}
```

金融数据返回结果说明

```
{
  "code": 200, 状态码
  "data": {
    "snapshot": {
      "000001": [ 产品代码
        "上证综指", 产品名称
        3217.96, 最新价格
        5.52, 涨跌幅
        0.17179999999999998, 涨跌额
        3219.52, 最高价格
        3195.29, 最低价格
        3212.44, 今日开盘价
        3212.44, 昨日收盘价
        1499412603 数据时间
      ],
      "fields": [
        "prod_name",
        "last_px",
        "px_change",
        "px_change_rate",
        "high_px",
        "low_px",
        "open_px",
        "preclose_px",
        "update_time"
      ]
    }
  }
}
```

金融数据图表代码下载

金融图表K线图demo下载

链接: <https://pan.baidu.com/s/1c2Ai4Pa> 密码: jufr

部署说明

默认环境：php5.3以上版本（使用其它后端程序，需要修改文件夹api内接口调用文件即可）

已成功购买阿里云API，获得APPCODE

本图表程序采用echarts图表组件开发

核心生成图表文件：电脑版本:js/index.js 手机版本:js/appindex.js

修改如图所示的接口调用地址，其中 api/r.php为实时数据接口 api/k.php为图表数据接口

```
var mkChartsMain = document.getElementById('mk-charts-main'),mkChartsData,candlePeriod,dataType;
var aList = document.getElementsByClassName('nav')[0].getElementsByTagName('a');
//均线&boll
var inputList = document.getElementsByClassName('main-indicator-selector')[0].getElementsByTagName('input');
//k线 或 折线
var shapeList = document.getElementsByClassName('shape');
//线条颜色
var maLineColor = ['#ec63a7', '#f5cc65', '#1e88e5'];
//涨跌样式
var upAndDownstyle = ['gt','lt'];
//数据刷新时间
var timer = 1000;

//产品类型 从地址栏获取code；默认使用EURUSD
var en_prod_code = getQueryStringHr("symbol") || "EURUSD"

/**
 * 获取url参数
 * @param name
 * @returns {*}
 */
function getQueryStringHr(name) {
    var reg = new RegExp("(^|&)" + name + "=[^&]*(&|$)");
    var r = window.location.search.substr(1).match(reg);
    if(r === null) {
        var reg2 = /^.*?[?]/;
        var r2 = window.location.hash.replace(reg2, "");
        r = r2.match(reg);
    }
    if(r != null) return decodeURI(decodeURI(r[2]));
    return null;
}

//查表头
function queryReal(){
    $.ajax({
        url: "api/r.php?en_prod_code="+en_prod_code,//接口调用地址
        dataType: "json",
        success: function(data){
            var data = data.data.snapshot[en_prod_code];
            var trade_status;
```

数据返回写入页面代码：

贵金属代码表

| 序号 | 产品名称 | 产品代码 | 演示地址 |
|----|-----------|-----------|----------------------|
| 1 | 黄金 | XAUUSD | 点击查看 |
| 2 | 白银 | XAGUSD | 点击查看 |
| 3 | 铜 | COPPER | 点击查看 |
| 4 | 铂金 | XPTUSD | 点击查看 |
| 5 | 钯金 | XPDUSD | 点击查看 |
| 6 | 上海黄金t+d | AUTD | 点击查看 |
| 7 | 上海白银t+d | AGTD | 点击查看 |
| 8 | 上海迷你黄金t+d | MAUTD | 点击查看 |
| 9 | 上海黄金9999 | AU9999 | 点击查看 |
| 10 | 上海黄金9995 | AU9995 | 点击查看 |
| 11 | 上海100克金条 | AU100G | 点击查看 |
| 12 | 上海铂金9995 | PT9995 | 点击查看 |
| 13 | 天津现货白银 | TTYXAGUSD | 点击查看 |
| 14 | 天津现货铜 | TTCU | 点击查看 |
| 15 | 天津现货钯金 | TTJPD | 点击查看 |
| 16 | 天津现货铂金 | TTJPT | 点击查看 |
| 17 | 天津现货镍 | TTNI | 点击查看 |
| 18 | 黄金ETF | GOLDETF | 点击查看 |
| 19 | 黄金CFTC | GOLDCFTC | 点击查看 |
| 20 | 纽约金 | COMEXGC | 点击查看 |
| 21 | 纽约银 | COMEXSL | 点击查看 |
| 22 | 纽约铜 | COPPER | 点击查看 |
| 23 | 伦敦期铅 | LEAD | 点击查看 |

| | | | |
|----|------|--------|----------------------|
| 24 | 伦敦期铝 | MAL | 点击查看 |
| 25 | 伦敦期铜 | MCU | 点击查看 |
| 26 | 伦敦期锌 | MZN | 点击查看 |
| 27 | 伦敦期镍 | NICKEL | 点击查看 |
| 28 | 伦敦期锡 | TIN | 点击查看 |

大宗商品代码表

| 产品名称 | 产品代码 | 演示地址 |
|-------|---------|----------------------|
| 布伦特原油 | UKOIL | 点击查看 |
| WTI原油 | USOIL | 点击查看 |
| 天然气 | NGAS | 点击查看 |
| 玉米 | CORN | 点击查看 |
| 小麦 | WHEAT | 点击查看 |
| 大豆 | SOYBEAN | 点击查看 |
| 糖 | SUGAR | 点击查看 |

外汇代码表

注意

本接口为延时数据，如需高频数据请查看[外汇高频数据接口](#)

本接口支持产品代码如下

| 产品名称 | 产品代码 | 演示地址 |
|---------|---------------|----------------------|
| 美元指数 | USDOLLARINDEX | 点击查看 |
| 离岸人民币 | USDCNH | 点击查看 |
| 在岸人民币 | USDCNY | 点击查看 |
| 欧元/美元 | EURUSD | 点击查看 |
| 英镑/美元 | GBPUSD | 点击查看 |
| 澳元/美元 | AUDUSD | 点击查看 |
| 新西兰元/美元 | NZDUSD | 点击查看 |
| 美元/日元 | USDJPY | 点击查看 |
| 美元/加元 | USDCAD | 点击查看 |
| 美元/瑞郎 | USDCHF | 点击查看 |
| 美元/港币 | USDHKD | 点击查看 |
| 比特币/美元 | BTCUSD | 点击查看 |
| 欧元/英镑 | EURGBP | 点击查看 |
| 欧元/澳元 | EURAUD | 点击查看 |
| 欧元/加元 | EURCAD | 点击查看 |
| 欧元/日元 | EURJPY | 点击查看 |
| 欧元/瑞郎 | EURCHF | 点击查看 |
| 欧元/新西兰元 | EURNZD | 点击查看 |
| 欧元/人民币 | EURCNY | 点击查看 |

| | | |
|----------|--------|----------------------|
| 欧元/捷克克朗 | EURCZK | 点击查看 |
| 欧元/丹麦克朗 | EURDKK | 点击查看 |
| 欧元/匈牙利福林 | EURHUF | 点击查看 |
| 欧元/波兰兹罗提 | EURPLN | 点击查看 |
| 欧元/瑞典克朗 | EURSEK | 点击查看 |
| 欧元/土耳其里拉 | EURTRY | 点击查看 |
| 欧元/挪威克朗 | EURNOK | 点击查看 |
| 英镑/澳元 | GBPAUD | 点击查看 |
| 英镑/加元 | GBPCAD | 点击查看 |
| 英镑/瑞郎 | GBPCHF | 点击查看 |
| 英镑/人民币 | GBPCNY | 点击查看 |
| 英镑/日元 | GBPJPY | 点击查看 |
| 英镑/新西兰元 | GBPNZD | 点击查看 |
| 澳元/人民币 | AUDCNY | 点击查看 |
| 澳元/加元 | AUDCAD | 点击查看 |
| 澳元/日元 | AUDJPY | 点击查看 |
| 澳元/新西兰元 | AUDNZD | 点击查看 |
| 澳元/瑞郎 | AUDCHF | 点击查看 |
| 加元/瑞郎 | CADCHF | 点击查看 |
| 加元/日元 | CADJPY | 点击查看 |
| 新西兰元/加元 | NZDCAD | 点击查看 |
| 新西兰元/日元 | NZDJPY | 点击查看 |
| 美元/巴西雷亚尔 | USDBRL | 点击查看 |
| 美元/捷克克朗 | USDCZK | 点击查看 |
| 美元/丹麦克朗 | USDDKK | 点击查看 |
| 美元/匈牙利福林 | USDHUF | 点击查看 |

| | | |
|-----------|--------|----------------------|
| 美元/挪威克朗 | USDNOK | 点击查看 |
| 美元/墨西哥比索 | USDMXN | 点击查看 |
| 美元/俄罗斯卢布 | USDRUB | 点击查看 |
| 美元/波兰兹罗提 | USDPLN | 点击查看 |
| 美元/新加坡元 | USDSGD | 点击查看 |
| 美元/土耳其里拉 | USDTRY | 点击查看 |
| 美元/南非兰特 | USDZAR | 点击查看 |
| 美元/瑞典克朗 | USDSEK | 点击查看 |
| 港元/人民币 | HKDCNY | 点击查看 |
| 港元/日元 | HKDJPY | 点击查看 |
| 南非兰特/日元 | ZARJPY | 点击查看 |
| 土耳其里拉/日元 | TRYJPY | 点击查看 |
| 新加坡元/日元 | SGDJPY | 点击查看 |
| 加拿大元/人民币 | CADCNY | 点击查看 |
| 斐济元/美元 | FJDUSD | 点击查看 |
| 加纳塞地/美元 | GHSUSD | 点击查看 |
| 以太币/美元 | ETHUSD | 点击查看 |
| 日元/人民币 | JPYCNY | 点击查看 |
| 美元/阿联酋迪拉姆 | USDAED | 点击查看 |
| 美元/以色列谢克尔 | USDILS | 点击查看 |
| 美元/印度卢比 | USDINR | 点击查看 |
| 美元/约旦第纳尔 | USDJOD | 点击查看 |
| 美元/科威特第纳尔 | USDKWD | 点击查看 |
| 美元/沙特里亚尔 | USDSAR | 点击查看 |
| 美元/泰铢 | USDTHB | 点击查看 |

数字货币数据

比特币代码说明

提供比特币以及其他热门虚拟货币的实时数据与走势图。

股指代码表

| 产品名称 | 产品代码 | 演示地址 |
|-----------|--------|----------------------|
| 比特币兑美元 | BTCUSD | 点击查看 |
| 比特币现金/比特币 | BCCBTC | 点击查看 |
| 比特币现金/美元 | BCCUSD | 点击查看 |
| 比特币现金/人民币 | BCCCNV | 点击查看 |
| 以太币/美元 | ETHUSD | 点击查看 |
| 莱特币/美元 | LTCUSD | 点击查看 |
| 达世币/美元 | DSHUSD | 点击查看 |
| 瑞波币/美元 | XRPUSD | 点击查看 |
| 比特币/人民币 | BTCCNY | 点击查看 |
| 莱特币/人民币 | LTCCNY | 点击查看 |
| 以太坊经典/美元 | ETCUSD | 点击查看 |
| 以太币/比特币 | ETHBTC | 点击查看 |
| 莱特币/比特币 | LTCBTC | 点击查看 |

股指代码表

股指代码表

| 序号 | 产品名称 | 产品代码 | 演示地址 |
|----|-------|--------|----------------------|
| 1 | 上证指数 | 000001 | 点击查看 |
| 2 | 深证成指 | 399001 | 点击查看 |
| 3 | 创业板指 | 399006 | 点击查看 |
| 4 | 沪深300 | 399300 | 点击查看 |
| 5 | 中小板指 | 399005 | 点击查看 |
| | | | 点 |

| | | | |
|----|-----------|--------------|----------------------|
| 6 | 富时中国A50指数 | CHINAA50 | 点击查看 |
| 7 | 恒生指数 | HKG33INDEX | 点击查看 |
| 8 | 标普500 | SPX500INDEX | 点击查看 |
| 9 | 道琼斯 | US30INDEX | 点击查看 |
| 10 | 纳斯达克 | NASINDEX | 点击查看 |
| 11 | 纳斯达克100 | NAS100INDEX | 点击查看 |
| 12 | 欧洲Stoxx50 | EUSTX50INDEX | 点击查看 |
| 13 | 日经225指数 | JPN225INDEX | 点击 |

| | | | |
|----|-------------|-------------|----------------------|
| 13 | 日经225指数 | JPN225INDEX | 查看 |
| 14 | 英国富时 100 | UK100INDEX | 点击查看 |
| 15 | 德国DAX | GER30INDEX | 点击查看 |
| 16 | 德国DAX | GER30INDEX | 点击查看 |
| 17 | 西班牙IBEX 35 | ESP35INDEX | 点击查看 |
| 18 | 澳大利亚ASX 200 | AUS200INDEX | 点击查看 |
| 19 | 韩国成份指数 | KOSPIINDEX | 点击查看 |
| 20 | 俄罗斯MICEX指 | MICEXINDEX | 点击 |

| | | | |
|----|----------------|----------------------|----------------------|
| | | | 看 |
| 21 | 波罗的海干散货指数 | BDIINDEX | 点击查看 |
| 22 | 印度孟买SENSEX指数 | BSESENSEX30INDEX | 点击查看 |
| 23 | 巴西BOVESPA股票指数 | BOVESPAINDEX | 点击查看 |
| 24 | 加拿大S&P/TSX综合指数 | SPTSXCOMPOSITEINDEX | 点击查看 |
| 25 | 希腊综合股指 | ATHENSGENERALINDEX | 点击查看 |
| 26 | 沙特阿拉伯综合指数 | TADAWULALLSHAREINDEX | 点击查看 |
| 27 | 墨西哥BOLSA指数 | IPCINDEX | 点击查看 |

| | | | |
|----|----------------|------------------------------------|----------------------|
| | | | 看 |
| 28 | 印尼雅加达综合指数 | JAKARTASTOCKEXCHANGECOMPOSITEINDEX | 点击查看 |
| 29 | 富时马来西亚KLCI综合指数 | FTSEMALAYSIAKLCIINDEX | 点击查看 |
| 30 | 菲律宾PSE综合股价指数 | PSEICOMPOSITEINDEX | 点击查看 |
| 31 | VIX波动率指数 | SP500VIXINDEX | 点击查看 |
| 32 | 欧洲Stoxx600 | STOXX600INDEX | 点击查看 |

债券代码表

债券代码表

| 产品名称 | 产品代码 | 演示地址 |
|------------|-------------------|----------------------|
| 美国10年期国债 | US10YEAR | 点击查看 |
| 美国1年期国债 | US1YEAR | 点击查看 |
| 美国2年期国债 | US2YEAR | 点击查看 |
| 美国3年期国债 | US3YEAR | 点击查看 |
| 美国5年期国债 | US5YEAR | 点击查看 |
| 美国7年期国债 | US7YEAR | 点击查看 |
| 美国30年期国债 | US30YEAR | 点击查看 |
| 中国10年期国债 | CHINA10YEAR | 点击查看 |
| 日本10年期国债 | JAPAN10YEAR | 点击查看 |
| 德国10年期国债 | GERMANY10YEA | 点击查看 |
| 英国10年期国债 | UK10YEAR | 点击查看 |
| 法国10年期国债 | FRANCE10YEAR | 点击查看 |
| 意大利10年期国债 | ITALY10YEAR | 点击查看 |
| 俄罗斯10年期国债 | RUSSIA10YEAR | 点击查看 |
| 印度10年期国债 | INDIA10YEAR | 点击查看 |
| 南非10年期国债 | SOUTHAFRICA10YEAR | 点击查看 |
| 巴西10年期国债 | BRAZIL10YEAR | 点击查看 |
| 阿根廷10年期国债 | ARGENTINA10YEAR | 点击查看 |
| 阿根廷30年期国债 | ARGENTINA30YEAR | 点击查看 |
| 澳大利亚10年期国债 | AUSTRALIA10YEAR | 点击查看 |
| 奥地利10年期国债 | AUSTRIA10YEAR | 点击查看 |
| | | |

| | | |
|-----------|------------------|----------------------|
| 加拿大10年期国债 | CANADA10YEAR | 点击查看 |
| 中国1年期国债 | CHINA1YEAR | 点击查看 |
| 中国2年期国债 | CHINA2YEAR | 点击查看 |
| 中国3年期国债 | CHINA3YEAR | 点击查看 |
| 中国5年期国债 | CHINA5YEAR | 点击查看 |
| 中国7年期国债 | CHINA7YEAR | 点击查看 |
| 希腊10年期国债 | GREECE10YEAR | 点击查看 |
| 印尼10年期国债 | INDONESIA10YEAR | 点击查看 |
| 日本3年期国债 | JP3YEAR | 点击查看 |
| 日本5年期国债 | JP5YEAR | 点击查看 |
| 葡萄牙10年期国债 | PORTUGAL10YEAR | 点击查看 |
| 韩国10年期国债 | SOUTHKOREA10YEAR | 点击查看 |
| 西班牙10年期国债 | SPAIN10YEAR | 点击查看 |

股指期货代码表

股指期货代码表

| 产品名称 | 产品代码 | 演示地址 |
|-------------|----------|----------------------|
| 富时中国A50股指期货 | CHINA50 | 点击查看 |
| 恒生指数期货 | HKG33 | 点击查看 |
| 标普500指数期货 | SPX500 | 点击查看 |
| 纳斯达克100指数期货 | NAS100 | 点击查看 |
| 道琼斯指数期货 | US30 | 点击查看 |
| 日经225指数期货 | JPN225 | 点击查看 |
| 欧洲50指数期货 | EUSTX50 | 点击查看 |
| 德国30指数期货 | GER30 | 点击查看 |
| 英国100指数期货 | UK100 | 点击查看 |
| 法国40指数期货 | FRA40 | 点击查看 |
| 意大利40指数期货 | ITA40 | 点击查看 |
| 西班牙35指数期货 | ESP35 | 点击查看 |
| 澳洲200指数期货 | AUS200 | 点击查看 |
| VIX波动率指数期货 | SP500VIX | 点击查看 |

ETF代码表

ETF代码表

| 产品名称 | 产品代码 | 演示地址 |
|---------------|------|----------------------|
| 安硕MSCI新兴市场指数 | EEM | 点击查看 |
| 安硕MSCI欧澳远东指数 | EFA | 点击查看 |
| Vectors黄金矿业基金 | GDX | 点击查看 |
| SPDR黄金信托基金 | GLD | 点击查看 |
| iShare 黄金信托基金 | IAU | 点击查看 |
| iShare 白银信托基金 | SLV | 点击查看 |
| SPDR标普500 | SPY | 点击查看 |
| 美国石油基金 | USO | 点击查看 |

快递查询接口

快递查询API

**** 购买地址以及价格表**** : [阿里云云市场](#)

**** 目前支持的快递公司**** : [点此查看](#)

示例代码

快递公司代码表

代码：jd 快递公司：京东 全称/别名：京东物流
代码：shunfeng 快递公司：顺丰 全称/别名：顺丰快递
代码：shentong 快递公司：申通 全称/别名：申通快递
代码：yunda 快递公司：韵达快运 全称/别名：韵达快运
代码：tiantian 快递公司：天天快递 全称/别名：天天快递
代码：yuantong 快递公司：圆通速递 全称/别名：圆通速递
代码：zhongtong 快递公司：中通速递 全称/别名：中通速递
代码：ems 快递公司：ems快递 全称/别名：ems快递
代码：huitongkuaidi 快递公司：汇通快运 全称/别名：汇通快运
代码：quanfengkuaidi 快递公司：全峰快递 全称/别名：全峰快递
代码：zhaijisong 快递公司：宅急送 全称/别名：宅急送
代码：aae 快递公司：aae全球专递 全称/别名：aae全球专递
代码：anjie 快递公司：安捷快递 全称/别名：安捷快递
代码：anxindakuaixi 快递公司：安信达快递 全称/别名：安信达快递
代码：biaojikuaidi 快递公司：彪记快递 全称/别名：彪记快递
代码：bht 快递公司：bht 全称/别名：bht
代码：baifudongfang 快递公司：百福东方国际物流 全称/别名：百福东方国际物流
代码：coe 快递公司：中国东方（COE） 全称/别名：中国东方（COE）
代码：changyuwuliu 快递公司：长宇物流 全称/别名：长宇物流
代码：datianwuliu 快递公司：大田物流 全称/别名：大田物流
代码：debangwuliu 快递公司：德邦物流 全称/别名：德邦物流
代码：dhl 快递公司：dhl 全称/别名：dhl
代码：dpex 快递公司：dpex 全称/别名：dpex
代码：dsukuaidi 快递公司：d速快递 全称/别名：d速快递
代码：disifang 快递公司：递四方 全称/别名：递四方
代码：fedex 快递公司：fedex（国外） 全称/别名：fedex（国外）
代码：feikangda 快递公司：飞康达物流 全称/别名：飞康达物流
代码：fenghuangkuaidi 快递公司：凤凰快递 全称/别名：凤凰快递
代码：feikuaida 快递公司：飞快达 全称/别名：飞快达
代码：guotongkuaidi 快递公司：国通快递 全称/别名：国通快递
代码：ganzhongnengda 快递公司：港中能达物流 全称/别名：港中能达物流
代码：guangdongyouzhengwuliu 快递公司：广东邮政物流 全称/别名：广东邮政物流
代码：gongsuda 快递公司：共速达 全称/别名：共速达
代码：hengluwuliu 快递公司：恒路物流 全称/别名：恒路物流
代码：huaxialongwuliu 快递公司：华夏龙物流 全称/别名：华夏龙物流
代码：haihongwangsong 快递公司：海红 全称/别名：海红

代码：haiwaihuanqiu 快递公司：海外环球 全称/别名：海外环球
代码：jiayiwuliu 快递公司：佳怡物流 全称/别名：佳怡物流
代码：jinguangsudikuaijian 快递公司：京广速递 全称/别名：京广速递
代码：jixianda 快递公司：急先达 全称/别名：急先达
代码：jjwl 快递公司：佳吉物流 全称/别名：佳吉物流
代码：jymwl 快递公司：加运美物流 全称/别名：加运美物流
代码：jindawuliu 快递公司：金大物流 全称/别名：金大物流
代码：jialidatong 快递公司：嘉里大通 全称/别名：嘉里大通
代码：jykd 快递公司：晋越快递 全称/别名：晋越快递
代码：kuaijiesudi 快递公司：快捷速递 全称/别名：快捷速递
代码：lianb 快递公司：联邦快递（国内） 全称/别名：联邦快递（国内）
代码：lianhaowuliu 快递公司：联昊通物流 全称/别名：联昊通物流
代码：longbanwuliu 快递公司：龙邦物流 全称/别名：龙邦物流
代码：lijisong 快递公司：立即送 全称/别名：立即送
代码：lejiedi 快递公司：乐捷递 全称/别名：乐捷递
代码：minghangkuaidi 快递公司：民航快递 全称/别名：民航快递
代码：meiguokuaidi 快递公司：美国快递 全称/别名：美国快递
代码：menduimen 快递公司：门对门 全称/别名：门对门
代码：ocs 快递公司：OCS 全称/别名：OCS
代码：peisihuoyunkuaidi 快递公司：配思货运 全称/别名：配思货运
代码：quanchenkuaidi 快递公司：全晨快递 全称/别名：全晨快递
代码：quanjitong 快递公司：全际通物流 全称/别名：全际通物流
代码：quanritongkuaidi 快递公司：全日通快递 全称/别名：全日通快递
代码：quanyikuaidi 快递公司：全一快递 全称/别名：全一快递
代码：rufengda 快递公司：如风达 全称/别名：如风达
代码：santaisudi 快递公司：三态速递 全称/别名：三态速递
代码：shenghuiwuliu 快递公司：盛辉物流 全称/别名：盛辉物流
代码：sue 快递公司：速尔物流 全称/别名：速尔物流
代码：shengfeng 快递公司：盛丰物流 全称/别名：盛丰物流
代码：saiaodi 快递公司：赛澳递 全称/别名：赛澳递
代码：tiandihuayu 快递公司：天地华宇 全称/别名：天地华宇
代码：tnt 快递公司：tnt 全称/别名：tnt
代码：ups 快递公司：ups 全称/别名：ups
代码：wanjiawuliu 快递公司：万家物流 全称/别名：万家物流
代码：wenjiesudi 快递公司：文捷航空速递 全称/别名：文捷航空速递
代码：wuyuan 快递公司：伍圆 全称/别名：伍圆
代码：wxwl 快递公司：万象物流 全称/别名：万象物流
代码：xinbangwuliu 快递公司：新邦物流 全称/别名：新邦物流
代码：xinfengwuliu 快递公司：信丰物流 全称/别名：信丰物流

代码：yafengsudi 快递公司：亚风速递 全称/别名：亚风速递

代码：yibangwuliu 快递公司：一邦速递 全称/别名：一邦速递

代码：youshuwuliu 快递公司：优速物流 全称/别名：优速物流

代码：youzhengguonei 快递公司：邮政包裹挂号信 全称/别名：邮政包裹挂号信

代码：youzhengguoji 快递公司：邮政国际包裹挂号信 全称/别名：邮政国际包裹挂号信

代码：yuanchengwuliu 快递公司：远成物流 全称/别名：远成物流

代码：yuanweifeng 快递公司：源伟丰快递 全称/别名：源伟丰快递

代码：yuanzhijiecheng 快递公司：元智捷诚快递 全称/别名：元智捷诚快递

代码：yuntongkuaidi 快递公司：运通快递 全称/别名：运通快递

代码：yuefengwuliu 快递公司：越丰物流 全称/别名：越丰物流

代码：yad 快递公司：源安达 全称/别名：源安达

代码：yinjiesudi 快递公司：银捷速递 全称/别名：银捷速递

代码：zhongtiekuaiyun 快递公司：中铁快运 全称/别名：中铁快运

代码：zhongyouwuliu 快递公司：中邮物流 全称/别名：中邮物流

代码：zhongxinda 快递公司：忠信达 全称/别名：忠信达

代码：zhimakaimen 快递公司：芝麻开门 全称/别名：芝麻开门

代码：annengwuliu 快递公司：安能物流 全称/别名：安能物流

快递查询代码(PHP)

```
<?php
// +-----
//wangjikeji.com
// 快递数据调用示例代码 - 网极科技
error_reporting(0);
header('Content-Type: text/html; charset=UTF-8');
$url = "http://express.api51.cn/query?postid=402479420842&temp=123456&type=shunfeng";
//请求地址
/*
postid:快递单号
type:快递公司拼音简写
*/

$content = wangji_curl($url);
echo $content;

function wangji_curl($url,$params=false,$ispost=0){
    $httpInfo = array();
    $ch = curl_init();
    $appcode = "*****"; //购买后可获得 购买后在阿里云控制台 云市场 查看
    $headers = array();
    array_push($headers, "Authorization:APPCODE " . $appcode);
    curl_setopt( $ch, CURLOPT_HTTP_VERSION , CURL_HTTP_VERSION_1_1 );
    curl_setopt( $ch, CURLOPT_USERAGENT , 'JuheData' );
    curl_setopt( $ch, CURLOPT_CONNECTTIMEOUT , 60 );
    curl_setopt( $ch, CURLOPT_TIMEOUT , 60);
    curl_setopt( $ch, CURLOPT_RETURNTRANSFER , true );
    curl_setopt($ch, CURLOPT_FOLLOWLOCATION, true);
    if( $ispost )
    {
        curl_setopt( $ch , CURLOPT_POST , true );
        curl_setopt( $ch , CURLOPT_POSTFIELDS , $params );
        curl_setopt( $ch , CURLOPT_URL , $url );
    }
    else
    {
        if($params){
            curl_setopt( $ch , CURLOPT_URL , $url.'?'.$params );
        }else{
            curl_setopt( $ch , CURLOPT_URL , $url);
        }
        curl_setopt($ch, CURLOPT_HTTPHEADER, $headers);
    }
    $response = curl_exec( $ch );
    if ( $response === FALSE ) {
        //echo "CURL Error: " . curl_error($ch);
        return false;
    }
}
```

```
$statusCode = curl_getinfo( $ch , CURLINFO_HTTP_CODE );  
$httpInfo = array_merge( $httpInfo , curl_getinfo( $ch ) );  
curl_close( $ch );  
return $response;  
}
```

快递查询返回结果

成功

```
{
  "message": "ok",
  "nu": "59152637163",
  "ischeck": "1",
  "condition": "F00",
  "com": "jd", 京东物流代码
  "status": "200",
  "state": "3",
  "data": [ 物流信息
    {
      "time": "2017-07-03 09:47:48",
      "ftime": "2017-07-03 09:47:48",
      "context": "货物已完成配送, 感谢您选择京东配送",
      "location": ""
    },
    {
      "time": "2017-07-03 08:03:39",
      "ftime": "2017-07-03 08:03:39",
      "context": "配送员开始配送, 请您准备收货, 配送员, 阙洪正, 手机号, 13883882813或153
40585575",
      "location": ""
    }
  ]
}
```

失败

订单号不存在或过期

```
{
  "message": "快递公司参数异常: 单号不存在或者已经过期",
  "nu": "",
  "ischeck": "0",
  "condition": "",
  "com": "",
  "status": "201",
  "state": "0",
  "data": []
}
```

快递公司代码错误或缺少必要参数

```
{  
  "message": "参数错误",  
  "nu": "",  
  "ischeck": "0",  
  "condition": "",  
  "com": "",  
  "status": "400",  
  "state": "0",  
  "data": []  
}
```

股票数据接口

[股票接口说明](#)

[股票代码说明](#)

[股票常见问题](#)

[交易状态](#)

[股票demo说明](#)

股票接口说明

演示地址

<http://demo.api51.cn>

购买地址

[阿里云API市场](#)

产品说明

本接口可获取包括A股、港股、美股、股指等数据，数据包括实时数据，与分时图K线图数据，以及股票关联信息，接口过多，请参考demo源码开发，Demo源码下载地址：<http://demo.api51.cn>。

提示：演示站已做流量限制，可能偶尔无法打开。建议下载demo程序本地搭建测试！

特别注意

由于股票数据专业性强，开发难度大，我们服务标准如下：

1、普通客户（按次数购买）

提供一次demo源码部署服务，不提供技术答疑

2、VIP客户（按年购买）

提供每个工作日1个小时的咨询时间

3、合作伙伴（委托开发）

提供每个工作日2个小时的咨询时间，提供定制开发服务。

声明

1. 本接口所引用资料均来源于互联网，版权归原作者所有，如有侵权，请联系我们，将立即改正。
2. 本接口数据仅供参考，不构成任何投资建议，用户据此操作造成任何风险及后果，均由用户自行负责。
3. 未经协议授权，任何人（包括媒体、网站、公司、其他组织、个人）不得转载、引用、摘编、盗链本接口资源，任何人不得复制或仿造本接口在非网极科技所属的服务器上建立镜像，以及利用其它方式使用接口数据等，否则我们将追究其法律责任。

实时数据

行情_报价

调用地址：http(s)://stock.api51.cn/real

请求方式：GET

返回类型：JSON

请求参数

| 参数名 | 类型 | 必填 | 最大长度 | 中文名 | 描述 |
|--------------|--------|----|------|-------|----------------|
| en_prod_code | String | 是 | 2000 | 产品代码集 | 一次最多允许查询100支代码 |
| fields | String | 否 | 255 | 字段集合 | 返回参数 |

返回参数

| 参数名 | 类型 | 最大长度 | 中文名 | 描述 |
|----------------------|---------|--------|--------|----|
| snapshot | Object | 行情对象 | | |
| prod_code | String | 32 | 产品代码 | |
| snapshot_detail_data | Object | 行情详细对象 | | |
| prod_name | String | 60 | 产品名称 | |
| data_timestamp | Integer | 10 | 最后更新时间 | |
| open_px | Integer | 10 | 开盘价 | |
| high_px | Integer | 10 | 最高价 | |
| low_px | Integer | 10 | 最低价 | |
| last_px | Integer | 10 | 最新价 | |
| preclose_px | Integer | 10 | 昨收价 | |

| | | | | |
|--------------------|---------|------|--------|--|
| business_amount | Integer | 10 | 成交数量 | |
| business_balance | Float | 16.2 | 成交金额 | |
| shares_per_hand | Integer | 10 | 每手股数 | |
| bid_grp | String | 255 | 委买档位 | |
| offer_grp | String | 255 | 委卖档位 | |
| w52_low_px | Integer | 10 | 52周最低价 | |
| w52_high_px | Integer | 10 | 52周最高价 | |
| px_change | Integer | 10 | 价格涨跌 | |
| px_change_rate | Integer | 10 | 涨跌幅 | |
| amount | Integer | 10 | 持仓量 | |
| entrust_diff | Integer | 16 | 委差 | |
| prev_amount | Float | 16.2 | 前余额 | |
| settlement | Integer | 10 | 今结算价 | |
| prev_settlement | Integer | 10 | 昨结算 | |
| market_value | Float | 16.2 | 证券市值 | |
| circulation_amount | Float | 16.2 | 流通股本 | |
| circulation_value | Integer | 16 | 流通市值 | |
| dyn_pb_rate | Integer | 8 | 动态市净率 | |
| pe_rate | Integer | 10 | 市盈率 | |
| entrust_rate | Float | 9.4 | 委比 | |
| turnover_ratio | Integer | 10 | 换手率 | |
| vol_ratio | Integer | 10 | 量比 | |
| amplitude | Integer | 10 | 振幅 | |

| | | | | |
|-----------------------------|---------|----------------|--------------|--|
| current_amount | Integer | 10 | 当前数量 | |
| money_type | String | 3 | 币种类别 | |
| trade_status | String | 16 | 交易状态 | |
| chi_spelling_grp | String | 255 | 拼音简称 | |
| hq_type_code | String | 32 | 类型代码 | |
| ipo_price | Integer | 10 | IPO价格 | |
| issue_date | Integer | 8 | 发行日期 | |
| up_px | Integer | 10 | 涨停价格 | |
| down_px | Integer | 10 | 跌停价格 | |
| business_amount_in | Integer | 16 | 内盘成交量 | |
| business_amount_out | Integer | 16 | 外盘成交量 | |
| eps | Float | 18.4 | 每股收益 (摊薄) | |
| bps | Integer | 10 | 每股净资产 | |
| iopv | Integer | 10 | 基金净值 | |
| exercise_date | Integer | 8 | 行权日期 | |
| expire_date | Integer | 8 | 到期日 | |
| exercise_price | Float | 12.6 | 行权价格 | |
| call_put | String | 32 | 认购认沽 | |
| min5_chgpct | Integer | 10 | 五分钟涨跌幅 | |
| stock_option_underlying_grp | Object | 个股期权标的 代码对象 | | |
| gamma_index | Integer | 10 | gamma指 | |

| | | | | |
|------------------|---------|----|---------|--|
| gamma_index | Integer | 10 | 标 | |
| delta_index | Integer | 10 | delta指标 | |
| real_leverage | Integer | 10 | 真实杠杆率 | |
| option_price | Integer | 10 | 期权定价 | |
| special_marker | String | 10 | 特殊标志 | |
| total_shares | Integer | 10 | 总股本 | |
| avg_px | Integer | 10 | 均价 | |
| wavg_px | Integer | 10 | 加权平均价 | |
| debt_fund_value | Integer | 10 | 国债基金净值 | |
| fund_t1_date | Integer | 10 | 基金T-1日期 | |
| fund_t1_value | Integer | 10 | 基金T-1净值 | |
| fund_t2_date | Integer | 10 | 基金T-2日期 | |
| fund_t2_value | Integer | 10 | 基金T-2净值 | |
| fund_start_date | Integer | 8 | 基金开始日期 | |
| fund_type | String | 10 | 基金类别 | |
| fund_buy_status | Integer | 10 | 基金买状态 | |
| fund_sell_status | Integer | 10 | 基金卖状态 | |

| | | | | |
|--------------------|---------|----|--------------|----------|
| fund_assess_change | Integer | 10 | 基金估值 涨跌幅 | 100 倍 |
| fund_real_change | Integer | 10 | 基金实时 涨跌幅(| 100 倍 |
| index_rise_trend | Integer | 10 | 指数上涨 趋势 | 多空 指标 |
| index_fall_trend | Integer | 10 | 指数下跌 趋势 | 强弱 度 |

正确返回示例

```
{
  "data": {
    "snapshot": {
      "fields": [
        "data_timestamp",
        "shares_per_hand",
        "open_px",
        "high_px",
        "low_px",
        "last_px",
        "business_amount",
        "business_balance",
        "offer_grp",
        "bid_grp"
      ],
      "600570.SS": [
        150305133,
        100,
        53980,
        56000,
        52510,
        54950,
        14465811,
        788995643,
        "54850,9887,0,54860,1500,0,54900,13300,0,54950,10000,0,54990,800,0,"
      ]
    }
  }
}
```

```
"54820,8000,0,54810,2100,0,54800,202900,0,54770,100,0,54720,1200,0,"  
],  
"000001.SZ": [  
153229327,  
100,  
15560,  
15830,  
15300,  
15480,  
170012067,  
2634796408,  
"15490,93700,0,15500,260609,0,15510,69996,0,15520,87008,0,15530,71400,0,"  
"15480,438292,0,15470,149000,0,15460,411400,0,15450,414573,0,15440,303733,0,"  
]  
}  
}  
}
```

K线数据

行情_K线

调用地址：[http\(s\)://stock.api51.cn/kline](http(s)://stock.api51.cn/kline)

请求参数

| 参数名 | 类型 | 必填 | 最大长度 | 默认值 | 中文名 | 描述 |
|---------------|---------|----|------|-----|------|---|
| get_type | String | 是 | 1 | | 查找类别 | offset 按偏移查找； range 按日期区间查找 必须输入其中一个值 |
| prod_code | String | 是 | 32 | | 产品代码 | 有且仅能有 1 个代码， 券代码包含交易所代码 后缀，作为该代码的唯一 标识。如：600570.SS |
| candle_period | Integer | 是 | 10 | | K线周期 | 取值可以是数字1-9， 示含义如下：1：1分钟 K线 2：5分钟K线 3：1 分钟K线 4：30分钟K线 60分钟K线 6：日K线 周K线 8：月K线 9：年 线 |
| candle_mode | Integer | 否 | 10 | 0 | K线模式 | 0：原始K线 1：前复权 K线 2：后复权K线 |
| | | | | | | 允许的字段：开盘价 open_px 最高价： high_px 最低价：low |

| | | | | | | |
|------------------|---------|---|-----|----|---------------|---|
| fields | String | 否 | 255 | | 字段集合 | 收盘价：close_px 成交量：business_amount 成交额： business_balance 如： 没有指定任何有效的字段，则返回所有字段 |
| search_direction | Integer | 否 | 10 | 1 | 搜索方向 | 1 表示向前查找（默认值），2 表示向后查找 仅在 get_type=offset 有效。 |
| date | Integer | 否 | 8 | | 日期 | 不输入默认为当前日期 请求日K线时，如果输入日期，不返回该日期的线 get_type=offset 时有效 |
| min_time | Integer | 否 | 10 | | 分时分钟时间 (HHMM) | 分钟 K 线的时间 HHMM 对于短周期 K 线类型使用(1min,5min 等)， 填写表示最新的市场时间，若填写必须同时填写 date 字段。 请求分钟时，如果输入该字段，返回输入分钟的K线 仅 get_type=offset 且 candle_period=1~5 钟 K 线) 时有效。 |
| data_count | Integer | 否 | 10 | 10 | 数据个数 | 需要取得的 K 线的根数 如果该字段不存在，取范围[1, 1000]，默认为 10 个。 仅在 |

| | | | | | | |
|------------|---------|---|---|--|------|--|
| | | | | | | get_type=offset 时有效。 |
| start_date | Integer | 否 | 8 | | 开始日期 | <p>1、 start_date 和 end_date 均不填， 返回距离当前日期的1000 K 线； 2、 仅填 start_date ， 当 start_date和最新日期之间的数据不足1000 根返回 start_date 和最末期之间的数据；如果数超过 1000 根 K 线， 返回距离当前日期的 1000 根 K线； 3、 仅 end_date ， 返回 end_date 之前存在的 最多1000 根 K 线； 4 start_date 和 end_da 均填充， 返回该日期区（闭区间）的数据，最 1000 根。 仅在 get_type=range 时有效。</p> |
| | | | | | | <p>默认为当前日期； 1、 start_date 和 end_da 均不填， 返回距离当前日期的1000 根 K 线； 2 仅填 start_date ， 当 start_date和最新日期之间的数据不足1000 根</p> |

| | | | | | |
|----------|---------|---|---|------|---|
| end_date | Integer | 否 | 8 | 截止日期 | 返回 start_date 和最 期之间的数据；如果类 超过 1000 根 K 线，！ 返回距离当前日期的 1000 根 K 线； 3、 仅 end_date ， 返回 end_date 之前存在的 最多1000 根 K 线； 4 start_date 和 end_da 均填充，返回该日期区 (闭区间) 的数据，最 1000 根。 仅在 get_type=range 时有 效。 |
|----------|---------|---|---|------|---|

返回参数

| 参数名 | 类型 | 最大长度 | 中文名 | 描述 |
|--------------------|---------|------------|------------------|----|
| candle | Object | K线对象 | | |
| prod_code | String | 32 | 产品代码 | |
| candle_detail_data | Object | K线详细 对象 | | |
| min_time | Integer | 10 | 分时分钟时间 (HHMM) | |
| open_px | Integer | 10 | 开盘价 | |
| high_px | Integer | 10 | 最高价 | |
| low_px | Integer | 10 | 最低价 | |
| close_px | Integer | 10 | 收盘价 | |
| business_amount | Integer | 10 | 成交数量 | |
| | | | | |

| | | | | |
|------------------|---------|------|------|--------------|
| business_balance | Float | 16.2 | 成交金额 | |
| turnover_ratio | Integer | 10 | 换手率 | 仅支持长周期 K线 |
| preclose_px | Integer | 10 | 前收价 | |

分时数据

行情_分时

调用地址：http(s)://stock.api51.cn/trend

请求方式：GET

返回类型：JSON

请求参数

| 参数名 | 类型 | 必填 | 最大长度 | 中文名 | 描述 |
|-----------|---------|----|------|---------|--|
| prod_code | String | 是 | 32 | 产品代码 | 有且仅有一只代码；证券代码包含交易所代码做后缀，作为该代码的唯一标识。如：000001.SS |
| fields | String | 否 | 255 | 字段集合 | 允许的字段：最新价：last_px；均价：avg_px；成交量：business_amount；成交额：business_balance；如果没有指定任何有效的字段，其效果等价于指定该字段：最新价，成交量 |
| crc | Integer | 否 | 10 | 循环冗余校验码 | 请求增量分时的情况下，必须带上此字段。该字段的值，由之前一次请求全量分时，服务器下发 |

| | | | | | |
|----------|---------|---|----|---------|---|
| date | Integer | 否 | 8 | 日期 | 只可指定一日，格式：YYYYmmdd；如果没有指定，取当前交易日 |
| min_time | Integer | 否 | 10 | 分时分分钟时间 | 请求增量分时时，格式：HHMM，可带上此字段，表示请求从该时间点开始的分时数据 不输入或输入时间是非交易时间时，默认为0，返回所有分时数据 |

返回示例

```
{
  "data": {
    "trend": {
      "fields": [
        "min_time",
        "last_px",
        "avg_px",
        "business_amount"
      ],
      "600001.SS": [
        [
          201501090930,
          54.98,
          54.98,
          28327
        ],
        [
          201501090931,
          54.63,
          54.829486,
          49700
        ]
      ]
    }
  }
}
```

5日分时

请求参数

| 参数名 | 类型 | 必填 | 最大长度 | 中文名 | 描述 |
|-----------|--------|----|------|------|--|
| prod_code | String | 是 | 32 | 产品代码 | 有且仅有一只代码；证券代码包含交易所代码做后缀，作为该代码的唯一标识。如：000001.SS |
| fields | String | 否 | 255 | 字段集合 | 允许的字段：最新价：last_px；均价：avg_px；成交量：business_amount；成交额：business_balance；如果没有指定任何有效的字段，其效果等价于指定该字段：最新价，成交量 |

返回参数

| 参数名 | 类型 | 最大长度 | 中文名 |
|-------------------|---------|------|--------------|
| trend_detail_data | Object | | 分时详细对象 |
| min_time | Integer | 10 | 分时分钟时间(HHMM) |
| last_px | Integer | 10 | 最新价 |
| avg_px | Integer | 10 | 均价 |
| business_amount | Integer | 10 | 成交数量 |
| business_balance | Float | 16.2 | 成交金额 |

返回示例

```
{
  "data": {
    "trend": {
      "fields": [
        "min_time",
        "last_px",
        "avg_px",
        "business_amount",
        "business_balance"
      ],
      "600570.SS": [
        [
          201504170931,
          124.07,
          124.83,
          387955,
          48426658
        ],
        [
          201504170932,
          123.01,
          124.59,
          572900,
          71378859
        ],
        [
          201504170933,
          121.89,
          123.94,
          834100,
          103378268
        ],
        [
          201504170934,
          122.44,
          123.74,
          941239,
          116465390
        ]
      ]
    }
  }
}
```

F10数据查询

F10数据查询

info参数说明

info=股票代码(不加后缀)_查询内容字母缩写

info=300731_gsgk 公司概况

info=300731_zdsx 重大事项

info=300731_jgsd 机构观点

info=000001_gdbd 股东变动

info=000001_cwfx 财务分析

info=000001_scfx 市场分析

info=000722_ggzd 个股诊断

返回数据示例

公司概况返回json示例

```

{
  "gsgk": [ 公司概况
    {
      "InnerCode": 3,
      "hy": "银行-银行",
      "ltgb": 16918007237,
      "gsmc": "平安银行股份有限公司",
      "zgb": 17170411366,
      "zyyw": "批发金融业务",
      "ssrq": "1991/04/03",
      "gsc1sj": "1987/12/22",
      "sjkzr": "深圳市人民政府国有资产监督管理委员会",
      "ssdq": "广东-深圳市"
    }
  ],
  "ssgn": [ 所属概念
    {
      "ConceptCode": 15031027,
      "InnerCode": 3,
      "indate": 1497974400000,
      "ID": 834,
      "LTwoName": "MSCI概念"
    }
  ],
  "zycpJY": [ 主营产品
    {

```

```

      "InnerCode": 3,
      "srzb": 55.02,
      "zycp": "批发金融业务"
    },
    {
      "InnerCode": 3,
      "srzb": 40.25,
      "zycp": "零售金融业务"
    },
    {
      "InnerCode": 3,
      "srzb": 4.71,
      "zycp": "其他业务"
    }
  ],
  "fhsz": [ 分红转送
    {
      "InnerCode": 3,
      "px": 1.58,
      "ExDiviDate": "2017-07-21",
      "bgq": "2016年报"
    },
    {
      "zz": 2,
      "InnerCode": 3,
      "px": 1.53,
      "ExDiviDate": "2016-06-16",
      "bgq": "2015年报"
    },
    {
      "zz": 2,
      "InnerCode": 3,
      "px": 1.74,
      "ExDiviDate": "2015-04-13",
      "bgq": "2014年报"
    }
  ],
  "glc": [ 管理层
    {
      "InnerCode": 3,
      "InDate": 1342540800000,
      "Background": " 谢永林,男,1968年出生,获得南京大学理学硕士、企业管理专业博士学位。1994年10月加入中国平安保险(集团)股份有限公司,历任江苏办事处国际业务部业务主任,平安产险南京分公司大厂支公司副经理(主持工作),平安寿险无锡支公司副经理(主持工作),平安寿险南京分公司团险部副经理、经理,平安寿险杭州分公司总经理助理、副总经理(主持工作),平安集团发展改革中心副主任,平安寿险市场营销部总经理、浙江分公司总经理等职。2006年3月加入平安银行,历任运营总监、人力资源总监,2007年7月至2012年6月任平安银行副行长,并自2007年6月至2012年6月任平安银行执行董事。2012年7月加入深圳发展银行。除上述简历披露的任职关系外,谢永林先生与深圳发展银行或其控股股东及实际控制人不存在其他关联关系,没有持有深圳发展银行股份,没有受过中国证监会的处罚和证券交易所惩戒。\\r\\n",
      "LeaderGender": 1,
      "Position": "董事长",
      "PositionSort": 1,

```



```
        "LeaderName": "谢永林",
        "LeaderDegree": 2,
        "PositionCode": 1,
        "BirthYMInfo": "1968"
    },
],
"addCode": "000001",
"addAbbr": "平安银行",
"addAbbrF": "华夏成长"
}
```

公司概况

info=300731_gsgk 公司概况

返回说明

```
{
  "gsgk": [{ //公司概况 基本内容
    "InnerCode": 3,
    "hy": "银行-银行", //所属行业
    "ltgb": 16918007237, //流通股本
    "gsmc": "平安银行股份有限公司", //公司名称
    "zgb": 17170411366, //总股本
    "zyyw": "批发金融业务", //主营业务
    "ssrq": "1991/04/03", //上市日期
    "gscslsj": "1987/12/22", //成立日期
    "sjkzr": "深圳市人民政府国有资产监督管理委员会", //实际控制人
    "ssdq": "广东-深圳市" //所属地区
  }],
  "ssgn": [{ // 所属概念
    "rankk": 1, //排序顺序
    "LTwoName": "供应链金融" //概念名称
  }],
  "zycp": [{ //主营产品
    "InnerCode": 3,
    "srzb": 132.50,
    "zycp": "利息收入"
  }],
  "zycpJY": [{ //主营产品经营
    "InnerCode": 3,
    "srzb": 55.02, //收入占比
    "zycp": "批发金融业务" //主营产品
  }],
  "fhsz": [{ //分红转送
    "zz": 2.0, //转增
    "InnerCode": 3,
    "px": 1.53, //派息
    "ExDiviDate": "2016-06-16", //除权付息日
    "bgq": "2015年报" //报告期
  }],
  "glc": [{ //管理层
    "InnerCode": 3,
    "InDate": 1342540800000,
    "Background": " 谢永林,男,1968年出生,获得南京大学理学硕士、企业管理专业博士学位。1994年10月加入中国平安保险(集团)股份有限公司,历任江苏办事处国际业务部业务主任,平安产险南京分公司大厂支公司副经理(主持工作),平安寿险无锡支公司副经理(主持工作),平安寿险南京分公司团险部副经理、经理,平安寿险杭州分公司总经理助理、副总经理(主持工作),平安集团发展改革中心副主任,平安寿险市场营销部总经理、浙
```

江分公司总经理等职。2006年3月加入平安银行, 历任运营总监、人力资源总监, 2007年7月至2012年6月任平安银行副行长, 并自2007年6月至2012年6月任平安银行执行董事。2012年7月加入深圳发展银行。除上述简历披露的任职关系外, 谢永林先生与深圳发展银行或其控股股东及实际控制人不存在其他关联关系, 没有持有深圳发展银行股份, 没有受过中国证监会的处罚和证券交易所惩戒。 \r\n",

```
    "LeaderGender": 1,
    "Position": "董事长",
    "PositionSort": 1,
    "LeaderName": "谢永林",
    "LeaderDegree": 2,
    "PositionCode": 1,
    "BirthYMInfo": "1968"
  }],
  "addCode": "000001",
  "addAbbr": "平安银行",
  "addAbbrF": "华夏成长"
}
```

行情_排序

行情_排序

调用地址：http(s)://stock.api51.cn/sort

请求方式：GET

返回类型：JSON

字段说明

en_hq_type_code 类型代码集 SS.ESA上证,SZ.ESA深证,XBHS.HY沪深板块

sort_field_name排序字段 如：turnover_ratio 换手率 px_change_rate涨跌率 参考实时数据返回字段

sort_type 排序方式 0：表示升序； 1：表示降序(默认)

可查询内容

1、沪深股票排行

所需参数：

```
en_hq_type_code:SS.ESA,SZ.ESA 选择沪深股市  
sort_field_name:px_change_rate 按涨跌率排序  
start_pos:0 起始位置  
data_count:15 所需数据个数  
sort_type:1 正序排序
```

2、沪深板块排行

所需参数：

```
en_hq_type_code:XBHS.HY 选择沪深板块  
sort_field_name:px_change_rate 按涨跌率排序  
start_pos:0 起始位置  
data_count:15 所需数据个数  
sort_type:1 正序排序
```

板块返回示例：

```
{
```

```

"data": {
  "sort": {
    "fields": [
      "prod_name", //板块名称
      "rise_first_grp", //领涨
      "px_change_rate" //板块涨跌率
    ],
    "430100.XBHS": [
      "房地产开发",
      "000517.SZ, 荣安地产, 3.81, 10.12, 000014.SZ, 沙河股份, 12.17, 10.04, 000718.SZ
, 苏宁环球, 4.95, 10.00, 000732.SZ, 泰禾集团, 27.71, 10.00, 000514.SZ, 渝 开 发, 6.97, 9.94, 600823.
SS, 世茂股份, 5.59, 9.61, 000667.SZ, 美好置业, 3.38, 9.39, 000042.SZ, 中洲控股, 18.06, 8.21, 600325.
SS, 华发股份, 8.34, 8.17, 000036.SZ, 华联控股, 9.93, 7.93, ",
      3.88
    ],
  },

```

客户效果图



沪深股票列表

调用地址：http(s)://stock.api51.cn/sort

请求方式：GET

返回类型：JSON

字段说明

en_hq_type_code 类型代码集 SS.ESA上证,SZ.ESA深证,XBHS.HY沪深板块

sort_field_name排序字段 如：turnover_ratio 换手率 px_change_rate涨跌率 参考实时数据返回字段

sort_type 排序方式 0：表示升序；1：表示降序(默认)

可查询内容

1、沪深股票排行

所需参数：

```
en_hq_type_code:SS.ESA,SZ.ESA 选择沪深股市
sort_field_name:px_change_rate 按涨跌率排序
start_pos:0 起始位置
data_count:15 所需数据个数
sort_type:1 正序排序
```

香港股票列表

调用地址：[http\(s\)://stock.api51.cn/sort](http(s)://stock.api51.cn/sort)

请求方式：GET

返回类型：JSON

字段说明

en_hq_type_code 类型代码集 [HKM.ES](#) 选择香港股市

sort_field_name 排序字段 如：turnover_ratio 换手率 px_change_rate 涨跌率 参考实时数据返回字段

sort_type 排序方式 0：表示升序；1：表示降序(默认)

可查询内容

香港股票排行

所需参数：

```
en_hq_type_code:HKM.ES 选择香港股市
sort_field_name:px_change_rate 按涨跌率排序
start_pos:0 起始位置
data_count:15 所需数据个数
sort_type:1 正序排序
```

美国股票列表

调用地址：http(s)://stock.api51.cn/sort

请求方式：GET

返回类型：JSON

字段说明

en_hq_type_code 类型代码集 O.ES纳斯达克, N.ES纽交所

sort_field_name排序字段 如：turnover_ratio 换手率 px_change_rate涨跌率 参考实时数据返回字段

sort_type 排序方式 0：表示升序； 1：表示降序(默认)

可查询内容

美国股票排行

所需参数：

```
en_hq_type_code:0.ES或者 N.ES
sort_field_name:px_change_rate 按涨跌率排序
start_pos:0 起始位置
data_count:15 所需数据个数
sort_type:1 正序排序
```


板块查询说明

板块说明

1、所有板块

调用接口：板块_关系

调用地址：http(s)://stock.api51.cn/block/info

所需参数：

| 参数 | 说明 | 举例 | 解释 |
|-------------|-------|------|----------|
| finance_mic | 类型代码集 | XBHS | (沪深板块) |

2、板块合集查询

调用接口：行情_排序

调用地址：http(s)://stock.api51.cn/sort

所需参数：

| 参数 | 说明 | 举例 | 解释 |
|-----------------|-------|---|---------------|
| en_hq_type_code | 类型代码集 | XBHS.GN | (概念板块) |
| sort_field_name | 排序字段 | px_change_rate | 以涨跌率排序 |
| fields | 字段合集 | prod_name, rise_first_grp, px_change_rate | 板块名称, 板块股票排序, |

| | | | |
|--|--|--|---------------|
| | | | 板块 涨跌 率 |
|--|--|--|---------------|

板块合集代码对照表

| 板块代码 | 板块说明 |
|-------------------------|-----------|
| XBHS.GN | 沪深概念板块 |
| XBHS.HY | 沪深行业板块 |
| XBHS.DY | 沪深地域板块 |
| XBHS.ZJHHY | 沪深证监会行业板块 |

3、查询股票所属板块

调用接口：板块_所属

调用地址：http(s)://stock.api51.cn/block/query

| 参数 | 说明 | 举例 |
|-----------|------|---------------------------|
| prod_code | 股票代码 | 000001.SZ |

返回该只股票所属的所有板块，H5demo个股展示所显示的板块是返回值中的行业板块（XBHS.HY）信息

4、查询板块内股票涨跌

调用接口：行情_排序

调用地址：http(s)://stock.api51.cn/sort

所需参数：

| 参数 | 说明 | 举例 | 解释 |
|-----------------|------|----------------|--------|
| en_prod_code | 板块代码 | 610300.XBHS | （其他建材） |
| sort_field_name | 排序字段 | px_change_rate | 以涨跌率排序 |

5、查询板块详细信息

调用接口：行情_报价

调用地址：http(s)://stock.api51.cn/real

所需参数：

| 参数 | 说明 | 举例 | 解释 |
|--------------|------|--|--------------------------------|
| en_prod_code | 板块代码 | 610300.XBHS | (其他建材) |
| fields | 查询字段 | prod_name,member_count,rise_count,fall_count | 板块名称,成员个数,上涨个数,下跌个数,其他参数参考实时数据 |

期货列表查询

期货列表排行

调用接口：行情_排序

调用地址：[http\(s\):// stock.api51.cn/sort](http(s)://stock.api51.cn/sort)

所需参数：

en_hq_type_code:选值见下方表
 sort_field_name:px_change_rate 按涨跌率排序
 start_pos:0 起始位置
 data_count:15 所需数据个数
 sort_type:1 正序排序

| en_hq_type_code代码 | 解释 |
|----------------------------|--------|
| CCFX.FF.INDEX.HS300 | 股指IF |
| CCFX.FF.INDEX.ZZ500 | 中证IC |
| CCFX.FF.INDEX.SZ50 | 上证IH |
| CCFX.FF.DEBT | 国债T |
| XSGE.FC.REBAR | 螺纹钢RB |
| XSGE.FC.NiSn | 沪镍NI |
| XSGE.FC.METAL | 金属ZN |
| XSGE.FC.SILVER | 沪银AG |
| XSGE.FC.RUBBER | 橡胶RU |
| XDCE.FC.INRU | 大宗工业I |
| DCE.FC.BEAN | 豆粕M |
| XDCE.FC.PALM | 棕榈油P |
| XDCE.FC.PETR | 石化期货PP |
| XDCE.FC.BOIL | 豆油Y |
| XZCE.FC.MA | 甲醇MA |

| | |
|---------------|--------|
| XZCE.FC.PTA | PTA TA |
| XZCE.FC.SUGAR | 白糖SR |

股票代码说明

股票代码说明

产品代码为股票6位数字代码+英文点+交易所识别码

交易所识别码

上证(SZ) 如 平安银行：[000001.SZ](#)

深证(SS) 如 新南洋：600661.SS

港股(HKM) 如 腾讯控股：00700.HKM

纳斯达克(O) 如 FacebookK：FB.O

纽交所(N) 如 趣店：QD.N

如：

平安银行：[000001.SZ](#)

新南洋：600661.SS

腾讯控股：00700.HKM

FacebookK：FB.O

趣店：QD.N

股票数据演示



股指代码表

| 产品名称 | 产品代码 | 具体说明 |
|------|-----------|------|
| 上证指数 | 000001.SS | |
| | | |

| | | |
|-------|-------------|------------------|
| 深证成指 | 399001.SZ | |
| 创业板指 | 399006.SZ | |
| 沪深300 | 399300.SZ | |
| 中小板指 | 399005.SZ | |
| 恒生指数 | HKG33INDEX | 本接口不包含，请在金融数据中查询 |
| 标普500 | SPX500INDEX | 本接口不包含，请在金融数据中查询 |
| 道琼斯 | US30INDEX | 本接口不包含，请在金融数据中查询 |
| 纳斯达克 | NASINDEX | 本接口不包含，请在金融数据中查询 |

股票常见问题

计费模式

统一采用阿里云API内置的按次数计费方式，用户需提前购买次数，次数使用完毕则无法调用。多次购买可以累加次数。

阿里云要求每笔订单有效期为一年。

demo源码如何部署

demo源码为php+highcharts组件开发，封装自选股功能，需要部署在支持php的环境上。

我们推荐部署环境为：

Linux+nginx+php5.6

- 1、阿里云6元虚拟主机部署，[点击购买](#)（选择linux系统，上传代码，修改appcode即可使用）
- 2、阿里云独享主机或ECS服务器可达到更好效果，[点击领取优惠券](#)

APP如何接入？

demo源码提供php版本的调用组件，可根据开发方式修改api内接口文件为您的程序语言。

也可部署好demo后直接引入，app和小程序可用web-view一键引入，其它语言可用Iframe等引入

交易状态

| 状态代码 | 状态说明 |
|----------|------------------------------|
| BREAK | 休市 |
| ENDTR | 交易结束 |
| PRETR | 盘前 |
| HALT | 暂停，除了自有订单和交易的查询之外，任何交易活动都被禁止 |
| START | 市场启动（初始化之后，集合竞价前） |
| STOPT | 长期停盘，停盘n天, $n \geq 1$ |
| OCALL | 开始集合竞价 |
| SUSP | 停盘，与HALT的区别在于SUSP时可以撤单 |
| TRADE | 交易（连续撮合） |
| POSIT | 盘后 |
| DELISTED | 退市 |

股票demo说明

下载地址

链接: <https://pan.baidu.com/s/1pKCqzbX> 密码: jsd9

部署说明

php5.3以上版本

开发说明

修改api目录中appcode.php文件

程序核心js/app.js文件，其中定义接口访问路径

如需其他语言，可依据api目录中接口文件进行修改，并修改app.js中接口请求路径

外汇高频数据接口

购买地址

[阿里云市场](#)

产品优势

直连欧洲交易中心，数据及时稳定
数据更新到小数点后五位，行业领先
支持241种产品，全面满足您的需要

查看演示

[点击查看](#)

支持产品

| 产品代码 | 产品名称 |
|--------|----------|
| XAUUSD | 黄金/美元 |
| XAGUSD | 白银/美元 |
| EURUSD | 欧元/美元 |
| USDJPY | 美元/日元 |
| GBPUSD | 英镑/美元 |
| USDCHF | 美元/瑞士法郎 |
| EURCHF | 欧元/瑞士法郎 |
| AUDUSD | 澳大利亚元/美元 |
| USDCAD | 美元/加拿大元 |
| NZDUSD | 新西兰元/美元 |
| EURGBP | 欧元/英镑 |
| EURJPY | 欧元/日元 |
| GBPJPY | 英镑/日元 |
| CHFJPY | 瑞士法郎/日元 |
| GBPCHF | 英镑/瑞士法郎 |
| EURAUD | 欧元/澳元 |
| EURCAD | 欧元/加元 |
| AUDCAD | 澳元/加元 |
| AUDJPY | 澳元/日元 |
| CADJPY | 加元/日元 |

NZDJPY 新西兰元/日元
GBPCAD 英镑/加元
GBPNZD 英镑/西兰元
GBPAUD 英镑/澳元
AUDNZD 澳元/新西兰元
USDSEK 美元/瑞典克朗
EURSEK 欧元/瑞典克朗
EURNOK 欧元/挪威克朗
USDNOK 美元/挪威克朗
USDMXN 美元/墨西哥元
AUDCHF 澳元/瑞士法郎
EURNZD 欧元/新西兰元
USDZAR 美元/南非兰特
ZARJPY 南非兰特/日元
USDTRY 美元/新土耳其里拉
EURTRY 欧元/新土耳其里拉
NZDCHF 新西兰元/瑞士法郎
CADCHF 加元/瑞士法郎
NZDCAD 新西兰元/加元
TRYJPY 新土耳其里拉/日元
USDCNH 美元/离岸人民币
USDEUR 美元/欧元
USDGBP 美元/英镑
USDAUD 美元/澳元
USDNZD 美元/新西兰元
USDXAU 美元/黄金
USDXAG 美元/白银
EURMXN 欧元/墨西哥元
EURZAR 欧元/南非兰特
EURCNH 欧元/离岸人民币
EURXAU 欧元/黄金
EURXAG 欧元/白银
JPYUSD 日元/美元
JPYEUR 日元/欧元
JPYGBP 日元/英镑
JPYCHF 日元/瑞士法郎
JPYAUD 日元/澳元
JPYCAD 日元/加元
JPYNZD 日元/新西兰元

JPYSEK 日元/瑞典克朗
JPYNOK 日元/挪威克朗
JPYMXN 日元/墨西哥元
JPYZAR 日元/南非兰特
JPYTRY 日元/新土耳其里拉
JPYCNH 日元/离岸人民币
JPYXAU 日元/黄金
JPYXAG 日元/白银
GBPEUR 英镑/欧元
GBPSEK 英镑/瑞典克朗
GBPNOK 英镑/挪威克朗
GBPMXN 英镑/墨西哥元
GBPZAR 英镑/南非兰特
GBPTRY 英镑/新土耳其里拉
GBPCNH 英镑/离岸人民币
GBPXAU 英镑/黄金
GBPXAG 英镑/白银
CHFUSD 瑞士法郎/美元
CHFEUR 瑞士法郎/欧元
CHFGBP 瑞士法郎/英镑
CHFAUD 瑞士法郎/澳元
CHFCAD 瑞士法郎/加元
CHFNZD 瑞士法郎/新西兰元
CHFSEK 瑞士法郎/瑞典克朗
CHFNOK 瑞士法郎/挪威克朗
CHFMXN 瑞士法郎/墨西哥元
CHFZAR 瑞士法郎/南非兰特
CHFTRY 瑞士法郎/新土耳其里拉
CHFCNH 瑞士法郎/离岸人民币
CHFXAU 瑞士法郎/黄金
CHFXAG 瑞士法郎/白银
AUDEUR 澳元/欧元
AUDGBP 澳元/英镑
AUDSEK 澳元/瑞典克朗
AUDNOK 澳元/挪威克朗
AUDMXN 澳元/墨西哥元
AUDZAR 澳元/南非兰特
AUDTRY 澳元/新土耳其里拉
AUDCNH 澳元/离岸人民币

AUDXAU 澳元/黄金
AUDXAG 澳元/白银
CADUSD 加元/美元
CADEUR 加元/欧元
CADGBP 加元/英镑
CADAUD 加元/澳元
CADNZD 加元/新西兰元
CADSEK 加元/瑞典克朗
CADNOK 加元/挪威克朗
CADMXN 加元/墨西哥元
CADZAR 加元/南非兰特
CADTRY 加元/新土耳其里拉
CADCNH 加元/离岸人民币
CADXAU 加元/黄金
CADXAG 加元/白银
NZDEUR 新西兰元/欧元
NZDGBP 新西兰元/英镑
NZDAUD 新西兰元/澳元
NZDSEK 新西兰元/瑞典克朗
NZDNOK 新西兰元/挪威克朗
NZDMXN 新西兰元/墨西哥元
NZDZAR 新西兰元/南非兰特
NZDTRY 新西兰元/新土耳其里拉
NZDCNH 新西兰元/离岸人民币
NZDXAU 新西兰元/黄金
NZDXAG 新西兰元/白银
SEKUSD 瑞典克朗/美元
SEKEUR 瑞典克朗/欧元
SEKJPY 瑞典克朗/日元
SEKGBP 瑞典克朗/英镑
SEKCHF 瑞典克朗/瑞士法郎
SEKAUD 瑞典克朗/澳元
SEKCAD 瑞典克朗/加元
SEKNZD 瑞典克朗/新西兰元
SEKNOK 瑞典克朗/挪威克朗
SEKMXN 瑞典克朗/墨西哥元
SEKZAR 瑞典克朗/南非兰特
SEKTRY 瑞典克朗/新土耳其里拉
SEKCNH 瑞典克朗/离岸人民币

SEKXAU 瑞典克朗/黄金
SEKXAG 瑞典克朗/白银
NOKUSD 挪威克朗/美元
NOKEUR 挪威克朗/欧元
NOKJPY 挪威克朗/日元
NOKGBP 挪威克朗/英镑
NOKCHF 挪威克朗/瑞士法郎
NOKAUD 挪威克朗/澳元
NOKCAD 挪威克朗/加元
NOKNZD 挪威克朗/新西兰元
NOKSEK 挪威克朗/瑞典克朗
NOKMXN 挪威克朗/墨西哥元
NOKZAR 挪威克朗/南非兰特
NOKTRY 挪威克朗/新土耳其里拉
NOKCNH 挪威克朗/离岸人民币
NOKXAU 挪威克朗/黄金
NOKXAG 挪威克朗/白银
MXNUSD 墨西哥元/美元
MXNEUR 墨西哥元/欧元
MXNJPY 墨西哥元/日元
MXNGBP 墨西哥元/英镑
MXNCHF 墨西哥元/瑞士法郎
MXNAUD 墨西哥元/澳元
MXNCAD 墨西哥元/加元
MXNNZD 墨西哥元/新西兰元
MXNSEK 墨西哥元/瑞典克朗
MXNNOK 墨西哥元/挪威克朗
MXNZAR 墨西哥元/南非兰特
MXNTRY 墨西哥元/新土耳其里拉
MXNCNH 墨西哥元/离岸人民币
MXNXAU 墨西哥元/黄金
MXNXAG 墨西哥元/白银
ZARUSD 南非兰特/美元
ZAREUR 南非兰特/欧元
ZARGBP 南非兰特/英镑
ZARCHF 南非兰特/瑞士法郎
ZARAUD 南非兰特/澳元
ZARCAD 南非兰特/加元
ZARNZD 南非兰特/新西兰元

ZARSEK 南非兰特/瑞典克朗
ZARNOK 南非兰特/挪威克朗
ZARMXN 南非兰特/墨西哥元
ZARTRY 南非兰特/新土耳其里拉
ZARCNH 南非兰特/离岸人民币
ZARXAU 南非兰特/黄金
ZARXAG 南非兰特/白银
TRYUSD 新土耳其里拉/美元
TRYEUR 新土耳其里拉/欧元
TRYGBP 新土耳其里拉/英镑
TRYCHF 新土耳其里拉/瑞士法郎
TRYAUD 新土耳其里拉/澳元
TRXCAD 新土耳其里拉/加元
TRYNZD 新土耳其里拉/新西兰元
TRYSEK 新土耳其里拉/瑞典克朗
TRYNOK 新土耳其里拉/挪威克朗
TRYMXN 新土耳其里拉/墨西哥元
TRYZAR 新土耳其里拉/南非兰特
TRXCNH 新土耳其里拉/离岸人民币
TRYXAU 新土耳其里拉/黄金
TRYXAG 新土耳其里拉/白银
CNHUSD 离岸人民币/美元
CNHEUR 离岸人民币/欧元
CNHJPY 离岸人民币/日元
CNHGBP 离岸人民币/英镑
CNHCHF 离岸人民币/瑞士法郎
CNHAUD 离岸人民币/澳元
CNHCAD 离岸人民币/加元
CNHNZD 离岸人民币/新西兰元
CNHSEK 离岸人民币/瑞典克朗
CNHNOK 离岸人民币/挪威克朗
CNHMXN 离岸人民币/墨西哥元
CNHZAR 离岸人民币/南非兰特
CNHTRY 离岸人民币/新土耳其里拉
CNHXAU 离岸人民币/黄金
CNHXAG 离岸人民币/白银
XAUEUR 黄金/欧元
XAUJPY 黄金/日元
XAUGBP 黄金/英镑

XAUCHF 黄金/瑞士法郎
XAUAUD 黄金/澳元
XAUCAD 黄金/加元
XAUNZD 黄金/新西兰元
XAUSEK 黄金/瑞典克朗
XAUNOK 黄金/挪威克朗
XAUMXN 黄金/墨西哥元
XAUZAR 黄金/南非兰特
XAUTRY 黄金/新土耳其里拉
XAUCNH 黄金/离岸人民币
XAUXAG 黄金/白银
XAGEUR 白银/欧元
XAGJPY 白银/日元
XAGGBP 白银/英镑
XAGCHF 白银/瑞士法郎
XAGAUD 白银/澳元
XAGCAD 白银/加元
XAGNZD 白银/新西兰元
XAGSEK 白银/瑞典克朗
XAGNOK 白银/挪威克朗
XAGMXN 白银/墨西哥元
XAGZAR 白银/南非兰特
XAGTRY 白银/新土耳其里拉
XAGCNH 白银/离岸人民币
XAGXAU 白银/黄金

外汇高频数据接口

| 产品代码 | 产品名称 |
|--------|-----------|
| XAUUSD | 黄金/美元 |
| XAGUSD | 白银/美元 |
| EURUSD | 欧元/美元 |
| USDJPY | 美元/日元 |
| GBPUSD | 英镑/美元 |
| USDCHF | 美元/瑞士法郎 |
| EURCHF | 欧元/瑞士法郎 |
| AUDUSD | 澳大利亚元/美元 |
| USDCAD | 美元/加拿大元 |
| NZDUSD | 新西兰元/美元 |
| EURGBP | 欧元/英镑 |
| EURJPY | 欧元/日元 |
| GBPJPY | 英镑/日元 |
| CHFJPY | 瑞士法郎/日元 |
| GBPCHF | 英镑/瑞士法郎 |
| EURAUD | 欧元/澳元 |
| EURCAD | 欧元/加元 |
| AUDCAD | 澳元/加元 |
| AUDJPY | 澳元/日元 |
| CADJPY | 加元/日元 |
| NZDJPY | 新西兰元/日元 |
| GBPCAD | 英镑/加元 |
| GBPZD | 英镑/西兰元 |
| GBPAUD | 英镑/澳元 |
| AUDNZD | 澳元/新西兰元 |
| USDSEK | 美元/瑞典克朗 |
| EURSEK | 欧元/瑞典克朗 |
| EURNOK | 欧元/挪威克朗 |
| USDNOK | 美元/挪威克朗 |
| USDMXN | 美元/墨西哥元 |
| AUDCHF | 澳元/瑞士法郎 |
| EURNZD | 欧元/新西兰元 |
| USDZAR | 美元/南非兰特 |
| ZARJPY | 南非兰特/日元 |
| USDTRY | 美元/新土耳其里拉 |
| EURTRY | 欧元/新土耳其里拉 |
| NZDCHF | 新西兰元/瑞士法郎 |
| CADCHF | 加元/瑞士法郎 |
| NZDCAD | 新西兰元/加元 |
| TRYJPY | 新土耳其里拉/日元 |
| USDCNH | 美元/离岸人民币 |
| USDEUR | 美元/欧元 |
| USDGBP | 美元/英镑 |
| USDAUD | 美元/澳元 |
| USDNZD | 美元/新西兰元 |
| USDXAU | 美元/黄金 |
| USDXAG | 美元/白银 |
| EURMXN | 欧元/墨西哥元 |
| EURZAR | 欧元/南非兰特 |
| EURCNH | 欧元/离岸人民币 |
| EURXAU | 欧元/黄金 |
| EURXAG | 欧元/白银 |
| JPYUSD | 日元/美元 |
| JPYEUR | 日元/欧元 |
| JPYGBP | 日元/英镑 |
| JPYCHF | 日元/瑞士法郎 |
| JPYAUD | 日元/澳元 |
| JPYCAD | 日元/加元 |
| JPYNZD | 日元/新西兰元 |
| JPYSEK | 日元/瑞典克朗 |
| JPYNOK | 日元/挪威克朗 |
| JPYMXN | 日元/墨西哥元 |
| JPYZAR | 日元/南非兰特 |
| JPYTRY | 日元/新土耳其里拉 |
| JPYCNH | 日元/离岸人民币 |
| JPYXAU | 日元/黄金 |
| JPYXAG | 日元/白银 |
| GBPEUR | 英镑/欧元 |
| GBPSEK | 英镑/瑞典克朗 |
| GBPNOK | 英镑/挪威克朗 |

图形验证码识别接口

图形验证码识别接口

由于阿里云内部风控，图形验证码已下架，需要的客户可直接在API无忧平台充值 www.api51.cn

POST/GET请求

地址：<http://www.api51.cn/api/CaptchaApi/getres>

如需https请修改为 <https://www.api51.cn>

价格

统一识别成功一次0.025元，识别失败不计费。直接登录www.api51.cn充值即可使用

参数说明

| 参数 | 说明 | 备注 |
|-------|-------------|---|
| token | 识别用户的唯一令牌 | 在 www.api51.cn 注册后可获得 |
| pic | 图片base64编码值 | 不支持 html inline base64，需要直接 文件内容 base64 |

返回示例：

识别成功

```
{
  "code": 0,
  "data": {
    "captchaId": "20171103:0000000000010654423692",
    "recognition": "AX23" //识别出来的
  },
  "message": "",
  "status": 1
}
```

识别失败

```
{
  "code": 10216003,
  "message": "上传的验证码图片数据有误 错误的图片"
}
```

代码示例

```
<html>
<body>

请输入Token :
<input type='text' id='token' value='' /><br>
请输入图片base64转码后的值 :
<input type='text' id='pic' value='' /><br>

<button onclick='send()'>识别</button>

<div id='content'>

</div>

<script src="https://cdn.bootcss.com/jquery/1.11.1/jquery.min.js"></script>

<script>

function send(){
var pic = $('#pic').val();
var token = $('#token').val();
$('#content').html("");
  $('#content').html("正在识别中....");

$.ajax({
  type: 'POST',
  url: 'http://www.api51.cn/api/CaptchaApi/getres',
  data: {
    'pic': pic,
    'token': token
  },
  dataType: "json",
  success: function(return_data){
    $('#content').html("");
    console.log(return_data);
    if(return_data.code=='0'){
      $('#content').html("识别结果为："+return_data.data.recognition);
    }else{
      $('#content').html(return_data.message);
    }
  }
});
}
```


关于调用次数的说明

什么是调用次数

阿里云上的价格表是XX元/1000次 这个次数是什么？

首先，API接口购买后会有一个Appcode 在哪？[点击这里查看](#)

然后，每个产品对应一个或者多个接口，如下图：

API接口

行情_报价

行情_K线

行情_分时

行情_分笔

行情_排序

旧_实时报价

行情_报价

调用地址：[http\(s\)://stock.api51.cn/real](http(s)://stock.api51.cn/real)

请求方式：GET

返回类型：JSON

API 调用：[API 简单身份认证调用方法 \(APPCODE \)](#) 展开▼

调试工具：[去调试](#)

▶ 请求参数 (Headers)

如上图所示，一个产品分出来好几个接口，每个接口有独立的请求地址与参数，对应返回不同的结果。那么使用Appcode调用某个接口一次 就算一次的次数。

如何预估需要购买多少次

一般我们会把API接口封装到我们的网站或者APP中，那么有多少人在使用我们的网站或者APP呢？这个大家肯定有一定的统计。

假如我们的网站每天有一千个人访问，每个人访问10分钟，我们需要同时调用2个接口，而且这2个接口每秒更新，那么计算公式如下：

1000人 乘以 10分钟 乘以 60秒 乘以 2个=120W次

那么我们每天就需要120万次的调用次数 一个月就是3600W次

一年大约是4亿次