



## 同花顺数据接口环境设置

修订时间：2023.10.11

---

浙江核新同花顺网络信息股份有限公司

地址：浙江省余杭区五常街道同顺街 18 号

邮编：310023

电话：952555

电子邮箱：[myhexin@myhexin.com](mailto:myhexin@myhexin.com)

官网：<http://quantapi.10jqka.com.cn/>

## 目录

1、接口使用介绍.....	3
1-1、使用流程介绍 .....	3
1-2、资料下载 .....	3
1-3、账号权限、数据量说明 .....	5
1-3-1、账号权限说明 .....	5
1-3-2、账号数据量说明 .....	5
1-4、超级命令客户端介绍 .....	7
2、SDK 接口环境修复说明 .....	9
2-1、SDK 接口 Windows 环境部署 .....	9
2-1-1、Python.....	9
2-1-2、MATLAB.....	11
2-1-3、Java .....	12
2-1-4、R.....	13
2-1-5、VBA .....	14
2-1-6、C#.....	16
2-1-7、C++.....	17
2-2、SDK 接口 linux 环境部署 .....	18
2-2-1、Python.....	19
2-2-2、Java .....	20
2-2-3、C++.....	21
3、HTTP 接口使用说明 .....	23
3-1、HTTP 接口说明.....	23
3-1-1、refresh token 与 access token 说明 .....	23
3-2、Postman 等工具使用 HTTP 接口 .....	24
3-2-1、获取 access token.....	24
3-2-2、获取数据 .....	25
3-3、Python 等语言环境使用 HTTP 接口.....	26
3-3-1、获取 access token.....	26
3-3-2、获取数据 .....	26

# 1、接口使用介绍

## 1-1、使用流程介绍

- 1) 阅读官网说明手册、示例及数据量计算等规则
- 2) 下载 Windows SDK 安装包登录超级命令并生成取数命令
- 3) 环境初始化设置
- 4) 开发环境使用数据接口账号登录
- 5) 开发环境通过取数命令执行提取数据

## 1-2、资料下载

- 1) SDK 接口安装包下载:

<http://quantapi.10jqka.com.cn/?page=home>



- 2) SDK 接口与 HTTP 接口说明文档下载:

<http://quantapi.10jqka.com.cn/?page=helpCenter>

产品手册

常见问题

文件下载

为了方便用户了解量化接口的安装及使用流程，提供相应文件供用户下载使用，具体如下：

1. 安装与使用手册下载

适用系统	编程语言	文件名称	版本号	升级时间	文件大小	文件下载
 Windows	全数据接口	同花顺数据接口用户手册-Windows	V2.0	2021-12-17	1.82MB	<a href="#">下载</a>
	Python	同花顺数据接口用户手册-Windows-Python	V3.0	2021-12-17	1.24MB	<a href="#">下载</a>
	Matlab	同花顺数据接口用户手册-Matlab	V3.0	2021-12-17	1.46MB	<a href="#">下载</a>
	HTTP	IFinD HTTP API 1.0用户手册	V1.1	2023-04-04	589.3KB	<a href="#">下载</a>
 Linux	全数据接口	同花顺数据接口用户手册-Linux	V2.0	2021-12-17	972.47KB	<a href="#">下载</a>
	Python	同花顺数据接口用户手册-Linux-Python	V3.0	2021-12-17	949.87KB	<a href="#">下载</a>

注：根据使用的编程语言下载对应说明，SDK 接口：Python、MATLAB 与 HTTP 接口有单

独文档，其他语言下载 v2.0 版本统一说明文档。

### 3) 接口代码示例下载:

<http://quantapi.10jqka.com.cn/?page=sample>

### 应用示例

[全部](#)[Python](#)[Matlab](#)[R](#)[C++](#)[C#](#)[Java](#)[VBA](#)



**数据接口基本函数应用(试用版正式版适用)**  
编程语言: python 上传日期: 2021-12-22  
文件大小: 6.04KB 下载次数: 18590次  
简介:  
本脚本为数据接口常用函数在通用场景的示例



**时点数据应用示例 (试用版正式版适用)**  
编程语言: python 上传日期: 2020-12-16  
文件大小: 3.84KB 下载次数: 5087次  
简介:  
演示如何提取财务指标的时点数据, 如何利用辅助功能更好的处理时点数据的应用场景, 如何在日期序列中根据自定义日期序列提取数据。



**python函数使用案例**  
编程语言: python 上传日期: 2019-12-09  
文件大小: 767B 下载次数: 5463次  
简介:  
演示python函数的使用, 直接对返回的JSON格式数据进行获取并保存



**大跌后指数表现情况案例**  
编程语言: python 上传日期: 2020-02-13  
文件大小: 2.6KB 下载次数: 2306次  
简介:  
分析股市大跌5%之后一个月, 一个季度和一年后的数据表现



**HTTP接口应用案例(python环境)**  
编程语言: python 上传日期: 2022-07-06  
文件大小: 8.48KB 下载次数: 2697次  
简介:  
演示在Python环境使用HTTP接口提取各债数数据

## 1-3、账号权限、数据量说明

### 1-3-1、账号权限说明

函数名称	免费账号	试用账号	正式账号
高序列函数	最近 1 年	最近 1 年	2010 年至今
实时行情函数	不限制	不限制	不限制
历史行情函数	最近 5 年	最近 5 年	证券上市至今
基础数据函数	最近 5 年	最近 5 年	证券上市至今
日期序列函数	最近 5 年	最近 5 年	证券上市至今
数据池函数	指定报表	不限制	不限制
专题报表函数	不限制	不限制	不限制
EDB 函数	不提供	最近 5 年	不限制
智能选股	不限制	不限制	不限制
组合管理	不限制	不限制	不限制
日内快照函数	最近 1 个月	最近 3 个月	2010 年至今
特色数据	不限制	不限制	不限制

注：日期序列中指数和板块起始日期和截止日期间隔不能超过 1 年，免费账号指拥有 ifind 产品账号的用户可以直接使用 ifind 账号登录接口。

### 1-3-2、账号数据量说明

#### 1) 免费账号

## 同花顺数据接口环境设置

免费版数据接口	
行情数据	高频序列：150万/月
	日内快照：200万/月
	历史行情：100万/月
	实时行情：300万/月
基本面数据	日期序列：60万/月
	基础数据：60万/月
	数据池：600次/月
特色数据	特色数据：2000/月
说明： 1) 1条数据指1个EXCEL单元格，比如同花顺在20180515的收盘价会统计为1条数据； 2) 数据统计会在每月1号的00:00进行数据清零； 3) 免费版账号仅需要ifind账号即可登录，不需要申请试用； 4) 日期序列与基础数据单次提取限制为1万单元格； 5) 高频序列单次提取限制为5万单元格； 6) 日内快照单次提取限制为10万单元格； 7) 日内快照只支持上海证券交易所、深圳证券交易所、上海期货交易所、大连商品交易所； 8) 历史行情单次提取限制为5万单元格； 9) 数据池仅能调取：板块成分、指数成分、每日交易龙虎榜、十大成交活跃股、沪深港通成交统计、沪深港通当日余额； 10) 特色数据目前包含：期股联动、形态预测、智能选股；其中提取限制为期股联动1000每月，形态预测1000每月，智能选股无限制； 11) 免费版公告函数上限为1万条	

## 2) 试用账号

试用版数据接口		
行情数据：1.5亿/周		
高频序列函数	实时行情函数	历史行情函数
日内快照		
基本面数据：500万条/周		
基础数据函数	日期序列函数	数据池函数
宏观经济数据：5万/周		
EDB请求函数		
特色数据：1万/周		
1) 1条数据指1个EXCEL单元格，比如同花顺在20180515的收盘价会统计为1条数据； 2) 数据统计会在每周周一的00:00进行数据清零； 3) 通过高频序列函数、实时行情函数、历史行情函数调用的数据会统计到行情数据中，即： 高频序列函数 $\leq 1.5$ 亿/周； 实时行情函数 $\leq 1.5$ 亿/周； 历史行情函数 $\leq 1.5$ 亿/周； 日内快照 $\leq 1.5$ 亿/周； (高频序列函数 + 实时行情函数 + 历史行情函数 + 日内快照) $\leq 1.5$ 亿/周 4) 通过基础数据函数、日期序列函数、数据池函数调用的数据会统计到基本面数据中，即： 基础数据函数 $\leq 500$ 万/周； 日期序列函数 $\leq 500$ 万/周； 数据池函数 $\leq 500$ 万/周； (基础数据函数 + 日期序列函数 + 数据池函数) $\leq 500$ 万/周 5) 通过EDB请求函数调用的数据会统计到宏观经济数据中，即： EDB请求函数 $\leq 5$ 万/周		

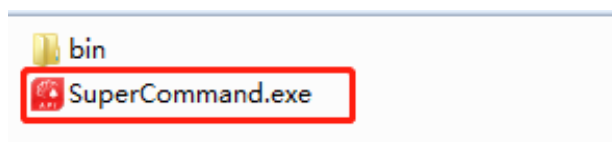
### 3) 正式账号

正式版数据接口		
行情数据：1.5亿/周		
高频序列函数	实时行情函数	历史行情函数
日内快照		
基本面数据：500万条/周		
基础数据函数	日期序列函数	数据池函数
宏观经济数据：5万/周		
EDB请求函数		
特色数据：1万/周		
1) 1条数据指1个EXCEL单元格，比如同花顺在20180515的收盘价会统计为1条数据； 2) 数据统计会在每周周一的00:00进行数据清零； 3) 通过高频序列函数、实时行情函数、历史行情函数调用的数据会统计到行情数据中，即： 高频序列函数 $\leq 1.5$ 亿/周； 实时行情函数 $\leq 1.5$ 亿/周； 历史行情函数 $\leq 1.5$ 亿/周； 日内快照 $\leq 1.5$ 亿/周； (高频序列函数 + 实时行情函数 + 历史行情函数 + 日内快照) $\leq 1.5$ 亿/周 4) 通过基础数据函数、日期序列函数、数据池函数调用的数据会统计到基本面数据中，即： 基础数据函数 $\leq 500$ 万/周； 日期序列函数 $\leq 500$ 万/周； 数据池函数 $\leq 500$ 万/周； (基础数据函数 + 日期序列函数 + 数据池函数) $\leq 500$ 万/周 5) 通过EDB请求函数调用的数据会统计到宏观经济数据中，即： EDB请求函数 $\leq 5$ 万/周		

## 1-4、超级命令客户端介绍

用户可以通过超级命令客户端查看接口支持的指标、辅助生成接口提取命令。

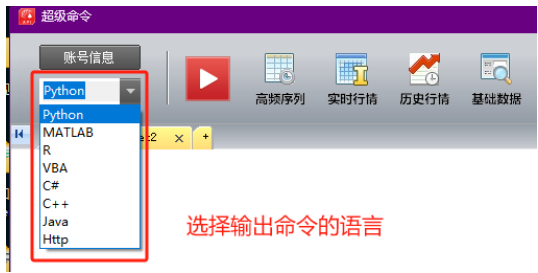
- 1) 下载 SDK 接口 Windows 安装包解压后双击 SuperCommand.exe 打开超级命令客户端；



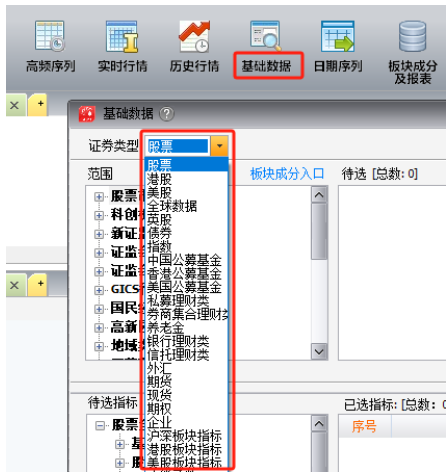
- 2) 指标查询与命令生成方式；

选择使用语言

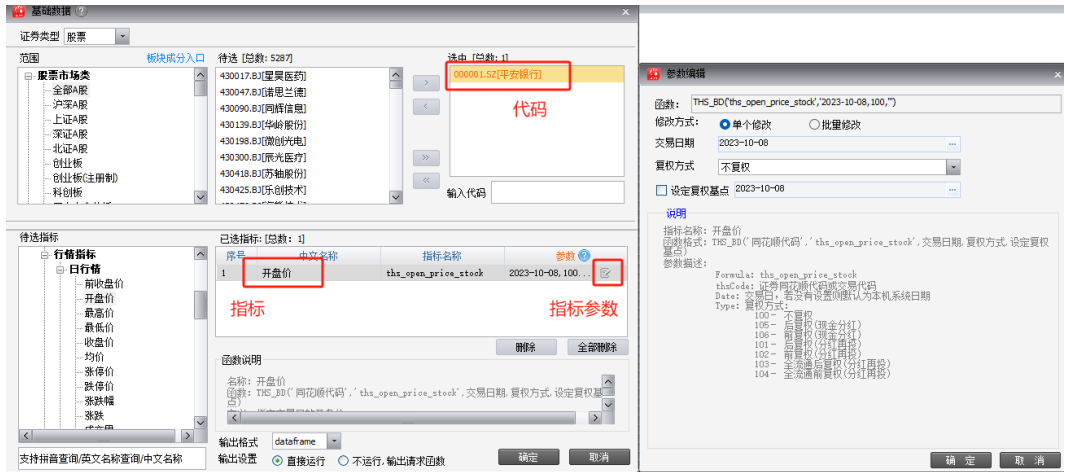
同花顺数据接口环境设置



选择函数、证券类型



选择代码、指标、指标参数生成命令



点击确定后生成接口命令





## 2、 SDK 接口环境修复说明

### 2-1、 SDK 接口 Windows 环境部署

下载 Windows SDK 安装包解压后，登录超级命令客户端-工具-环境设置进行环境修复



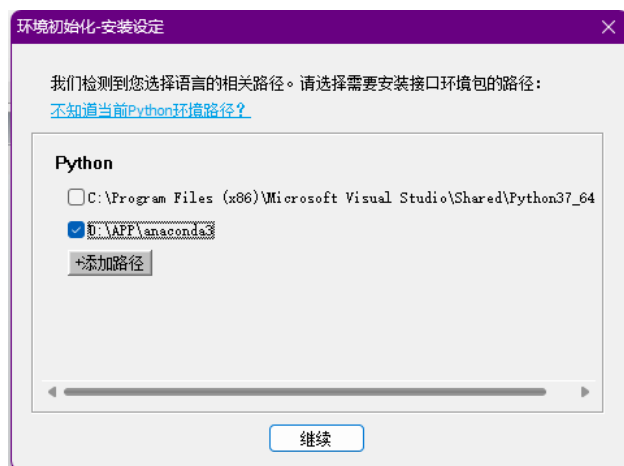
注：环境修复前用户需有对应语言的开发环境，此步骤不会帮助用户安装。

#### 2-1-1、Python

1) 选择 Python 语言点击确定



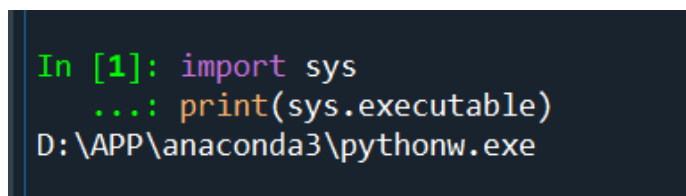
2) 选择使用的 Python 版本路径进行修复



如未识别出路径可以点击手动添加路径；

如不清楚目前所使用的 Python 路径，可以执行如下命令查看：

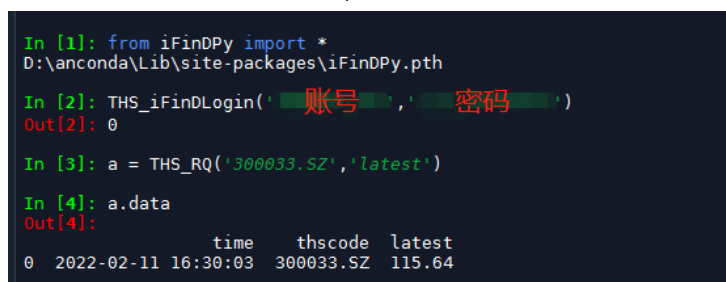
```
import sys
print(sys.executable)
```



- 3) 点击继续后程序会自动进行修复，点击显示桌面如出现如下提醒框则表示修复成功，点击确定完成修复



- 4) 修复成功后可以在开发软件中进行接口测试(建议修复后重启 Spyder、PyCharm 等开发工具，如不重启可能还会报错)

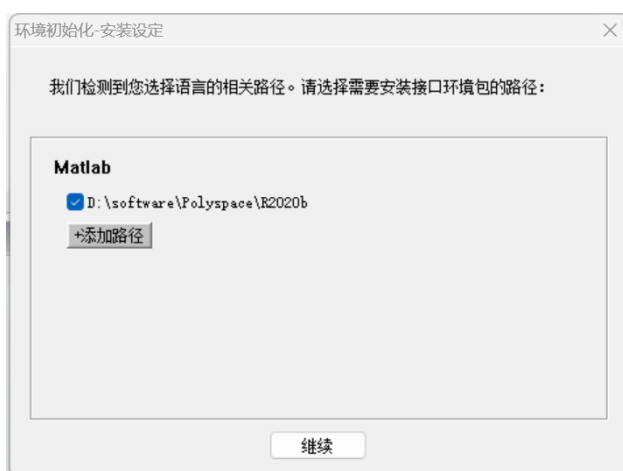


## 2-1-2、MATLAB

- 1) 选择 MATLAB 语言点击确定



- 2) 选择使用的 MATLAB 版本路径进行修复



如未识别出路径可以点击手动添加路径

- 3) 点击继续后程序会自动进行修复，点击显示桌面出现如下提醒框则表示修复成功，点击确定完成修复



- 4) 修复成功后可以在 MATLAB 中进行接口取数测试

```
>> THS_iFinDLogin("账号","密码")

ans =

    0

>> [data,errorcode,time,indicators,thscore,errmsg,dataVol,datatype,perf]=THS_RQ('300033.SZ','latest','','format:table')

data =

1×3 table

      time      thscore      latest
-----
{' 2022-03-04 16:30:03'} {' 300033.SZ'} 102.75
```

2-1-3、Java

1) 选择 Java 语言点击确定



2) 选择使用的 Java 版本路径进行修复



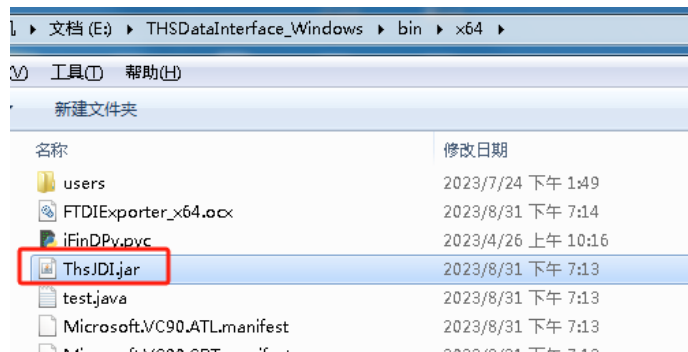
3) 点击继续后程序会自动进行修复，点击显示桌面出现如下提醒框则表示修复成功，点击确定完成修复



#### 4) Jar 包导入路径

64 位: THSDataInterface\_Windows\bin\x64\ThsJDI.jar

32 位: THSDataInterface\_Windows\bin\x86\ThsJDI.jar



#### 5) 修复成功后可以使用官网示例修改测试接口

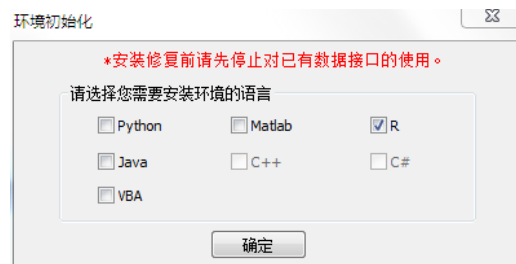
```
System.Load("E://THSDataInterface_Windows//bin//x64//iFinDJava_x64.dll");

//登入
JDIBridge client = new JDIBridge();
int ret = client.THS_iFinDLogin("账号", "密码");

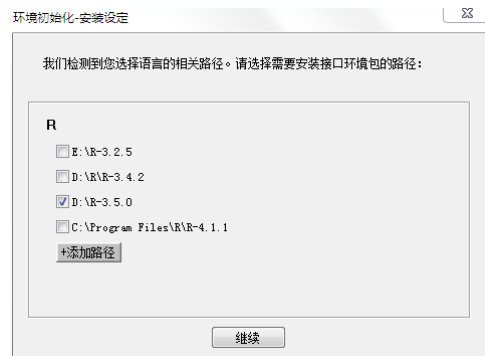
long startTime = System.currentTimeMillis(); //获取开始时间
String strResultDataSeries = JDIBridge.THS_RealtimeQuotes("300033.SZ", "latest");
```

### 2-1-4、R

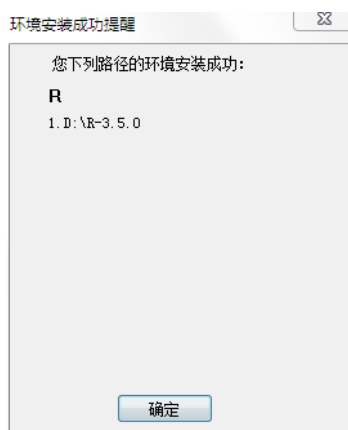
#### 1) 选择 R 语言点击确定



#### 2) 选择目前使用的 R 版本路径进行修复



#### 3) 选择继续后程序会自动进行修复，点击显示桌面出现如下提醒框则表示修复成功，点击确定完成修复



- 4) 修复成功后可以在 R 中进行接口测试

```
> library(iFinDR)
> library(RJSONIO)
> THS_iFinDLogin('账号', '密码')
[1] 0
> a = THS_RealtimeQuotes('300033.SZ', 'latest', '', TRUE)
> THS_Trans2DataFrame(a)
      time  thscode latest
1 2022-06-20 15:15:45 300033.SZ 96.01
```

## 2-1-5、VBA

- 1) 选择 VBA 语言点击确定



- 2) 选择 VBA 路径进行修复



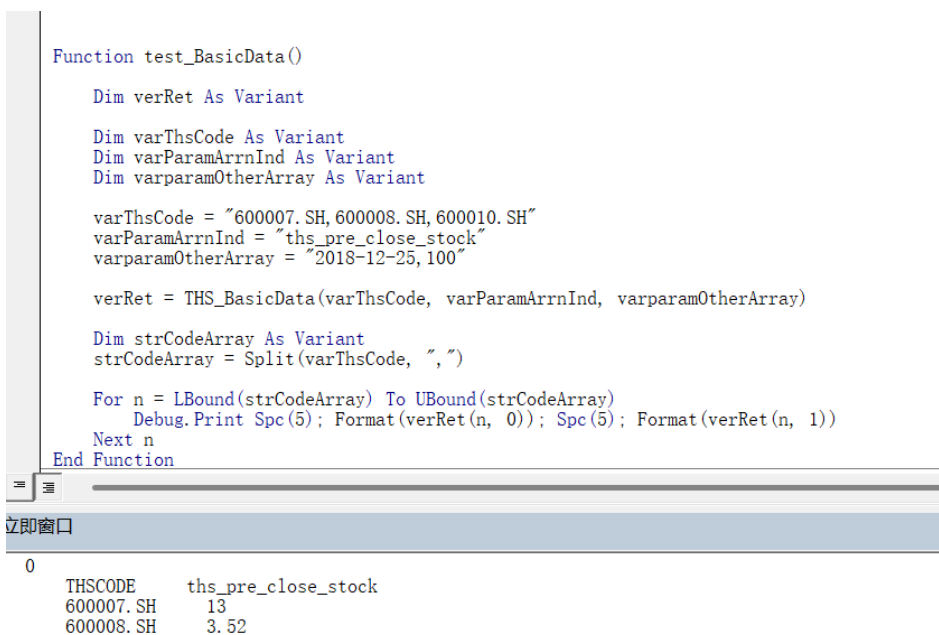
- 3) 选择继续后程序会自动进行修复，点击显示桌面出现如下提醒框则表示修复成功，点击确定完成修复



4) 修复成功后可以用官网 VBA 示例进行接口测试



```
Sub login()
    Dim verRet As Variant
    verRet = THS_iFindLogin("用户名", "密码")
    Debug.Print verRet
End Sub
```



## 2-1-6、C#

### 1) C#语言无需在终端中修复，直接下载官网示例测试

下载地址：<http://quantapi.10jqka.com.cn/?page=sample>



### 2) 修改接口安装包本地路径、填写账号、密码

```
//修改下面路径为本地数据接口安装包路径；如果是64位则目录为E:\\THSDDataInterface_Windows\\bin\\x64\\，文件名称为FTDataInterface_x64.dll
const string strPath = @"E:\\THSDDataInterface_Windows\\bin\\x86\\";
const string strDllFileName = "FTDataInterface.dll";
const string strDllFile = strPath + strDllFileName;

0 个引用
static void Main(string[] args)
{
    /*登录函数，填写账号、密码
    */
    string strUserName = "账号";
    string strPassWord = "密码";
    int nRet = THS_iFindLogin(strUserName, strPassWord);
    string strOut = "Login result:" + nRet;
    Console.WriteLine(strOut);
    Thread.Sleep(3000);

    _event = new AutoResetEvent(false);
    ThreadStart threadStart = new ThreadStart(testSample);
    Thread thread = new Thread(threadStart);
    thread.Start();
    _event.WaitOne();
}
```

### 3) 修改需要的函数命令执行

```
//实时行情函数
RealtimeQuotesSample();

static void RealtimeQuotesSample()
{
    string strOut;
    //请求实时数据
    /*实时行情函数的格式为THS_RealtimeQuotes('thsCodes','indicators','params')
    * thsCodes不可以为空，且支持多个输入，当有多个thsCodes则用英文半角逗号分隔，如thsCode1,thsCode2,thsCode3
    * indicators不可以为空，且支持多个输入，当有多个indicators则用英文半角分号分隔，如indicator1;indicator2;indicator3
    * params不可以为空
    * 老版本：THS_RealtimeQuotes('thsCode1,thsCode2,thsCode3','indicator1;indicator2;indicator3','param1,param2,param3')
    * 第三个参数 'pricetype:1' 为债券报价方式参数，超级命令生成其他证券类型的命令没有这个参数，但请不要填空，按照下面命令填写 'pricetype:1'
    */
    strOut = THS_RealtimeQuotes("300033.SZ", "latest", "pricetype:1");
    Console.WriteLine(strOut);
}

Login result:0
{"errorCode":0,"errmsg":"Success!","tables":[{"pricetype":1,"thscode":"300033.SZ","time":["2022-06-20 16:30:06"],"table":{"latest":[96.01]}}
```



## 2-1-7、C++

- 1) C++语言也无需在终端中修复，直接下载官网的示例测试



- 2) 修改接口安装包本地路径、填写账号、密码

```
int InitLong = sizeof(void *) / sizeof(int);
#ifdef _WIN32
//下面路径按照本地接口安装包路径修改
if (!InitialFunction("E:\\THSDataInterface_Windows\\bin\\x86\\ShellExport"))
{
    return 0;
}
//填写账号密码
int ret = THS_iFindLogin("账号", "密码");
std::cout << "Ret:" << ret << std::endl;
```

- 3) 修改需要的函数命令执行

```
//实时行情
//第三个参数 'pricetype:1' 为债券报价方式参数，超级命令生成其他证券类型的命令没有这个参数，但请不要填空，按照下面命令填写 'pricetype:1'
int ret = THS_RealtimeQuotes("300033.SZ", "latest", "pricetype:1", &RetData);
```

```
Ret:0
result:<"errorCode":0,"errmsg":"Success!","tables":[{"pricetype":1,"thscode":"300033.SZ","time":["2022-06-20 16:30:06"],"table":{"latest":{"96.01}}}]
```

## 2-2、SDK 接口 linux 环境部署

在调用前必须在解压后的接口目录中使用 ldd libShellExport.so、ldd libFTDataInterface.so 和 ldd hqdatafeed 查看接口库所依赖的环境是否齐全，如果不齐全,请使用 yum 或者 apt-get 安装(以下说明命令以 Ubuntu 环境为例)，如下：

Centos: sudo yum install -y libgcc.i686 zlib.i686 glibc.i686 libstdc++-devel.i686

Ubuntu: sudo apt-get install libc6:i386 libncurses5:i386 libstdc++6:i386

1) 如出现以下‘not found’报错，按照缺少报错安装

```
an@litian-virtual-machine:~/ifind/bin64$ ldd libFTDataInterface.so
linux-vdso.so.1 (0x00007ffd41380000)
libhcrypto-3.1.so => ./libhcrypto-3.1.so (0x00007fbd63800000)
librt.so.1 => /lib/x86_64-linux-gnu/librt.so.1 (0x00007fbd6484d000)
libdl.so.2 => /lib/x86_64-linux-gnu/libdl.so.2 (0x00007fbd64848000)
libidn.so.11 => not found
libstdc++.so.6 => /lib/x86_64-linux-gnu/libstdc++.so.6 (0x00007fbd635d4000)
libm.so.6 => /lib/x86_64-linux-gnu/libm.so.6 (0x00007fbd6475f000)
libgcc_s.so.1 => /lib/x86_64-linux-gnu/libgcc_s.so.1 (0x00007fbd6473f000)
libc.so.6 => /lib/x86_64-linux-gnu/libc.so.6 (0x00007fbd633ac000)
libpthread.so.0 => /lib/x86_64-linux-gnu/libpthread.so.0 (0x00007fbd6473a000)
/lib64/ld-linux-x86-64.so.2 (0x00007fbd64863000)
```

2) 常见缺少 libidn.so.11 报错安装方式

可以先搜索一下 libidn 并安装：

查找：apt-cache search libidn

安装：sudo apt install libidn-dev

```
litian@litian-virtual-machine:~/ifind/bin64$ apt-cache search libidn
libidn12 - GNU Libidn library, implementation of IETF IDN specifications
libidn2-0 - Internationalized domain names (IDNA2008/TR46) library
libidn-dev - Development files for GNU Libidn, an IDN library
libidn11-dev - Transitional development package for GNU Libidn
libidn2-dev - Internationalized domain names (IDNA2008/TR46) development files
libidn2-doc - Internationalized domain names (IDNA2008/TR46) documentation
libnet-libidn-perl - Perl bindings for GNU Libidn
idn - Command line interface to GNU Libidn
idn2 - Internationalized domain names (IDNA2008/TR46) command line tool
libidna-punycode-perl - module to encode Unicode string in Punycode
```

安装后如 ldd libFTDataInterface.so 查看环境依赖还是缺少，可能是安装的位置有误

先搜索：sudo find / | grep libidn.so

如下图可以查看到 libidn.so.11 在目录：/snap/core18/2538/lib/x86\_64-linux-gnu/

将文件移动到 ldd 查询到的其他文件的目录：/lib/x86\_64-linux-gnu/

命令：sudo cp /snap/core18/2538/lib/x86\_64-linux-gnu/libidn.so.11 /lib/x86\_64-linux-gnu

```

litian@litian-virtual-machine:~/ifind/bin64$ sudo find / | grep libidn.so
/libidn.so.11
/home/litian/ifind/bin64/libidn.so.11
/home/litian/Downloads/qq-files/3409779239/file_recv/idn/libidn.so.11
/home/litian/Downloads/qq-files/3409779239/file_recv/idn/libidn.so
/home/litian/Downloads/qq-files/3409779239/file_recv/idn/libidn.so.11.6.1
/snap/core18/2538/lib/x86_64-linux-gnu/libidn.so.11
/snap/core18/2538/lib/x86_64-linux-gnu/libidn.so.11.6.16
/root/.local/share/Trash/files/libidn.so
/root/.local/share/Trash/files/libidn.so.11.6.1
/root/.local/share/Trash/info/libidn.so.trashinfo
/root/.local/share/Trash/info/libidn.so.11.6.1.trashinfo
/usr/lib/x86_64-linux-gnu/libidn.so.11
/usr/lib/x86_64-linux-gnu/libidn.so.12.6.3
/usr/lib/x86_64-linux-gnu/libidn.so
/usr/lib/x86_64-linux-gnu/libidn.so.12
/usr/lib64/libidn.so.11
find: '/run/user/1000/doc': 权限不够
find: '/run/user/1000/gvfs': 权限不够
litian@litian-virtual-machine:~/ifind/bin64$ ldd libShellExport.so
linux-vdso.so.1 (0x00007ffcb33ad000)
libidn.so.11 => /lib/x86_64-linux-gnu/libidn.so.11 (0x00007fb58c800000)
libpthread.so.0 => /lib/x86_64-linux-gnu/libpthread.so.0 (0x00007fb58ce7d000)
libdl.so.2 => /lib/x86_64-linux-gnu/libdl.so.2 (0x00007fb58ce78000)
libstdc++.so.6 => /lib/x86_64-linux-gnu/libstdc++.so.6 (0x00007fb58ce5d4000)
libm.so.6 => /lib/x86_64-linux-gnu/libm.so.6 (0x00007fb58cb19000)
libgcc_s.so.1 => /lib/x86_64-linux-gnu/libgcc_s.so.1 (0x00007fb58ce56000)
libc.so.6 => /lib/x86_64-linux-gnu/libc.so.6 (0x00007fb58c3ac000)
/lib64/ld-linux-x86-64.so.2 (0x00007fb58ce93000)

```

## 2-2-1、Python

1) 按照上述说明环境依赖查询已经齐全后，调用接口目录中 installiFinDPy.py 安装

例如：接口压缩包解压放在/lib 目录下

32 位调用接口 bin 目录中的 installiFinDPy.py 安装，输入参数为文件解压后的文件路径

例如:压缩包解压放在/lib 目录下

Sudo python /lib/bin/installiFinDPy.py /lib

64 位调用接口 bin64 目录中的 installiFinDPy.py 安装，输入参数为文件解压后的文件路径

例如:压缩包解压放在/lib 目录下

Sudo python /lib/bin64/installiFinDPy.py /lib

```

litian@litian-virtual-machine:~/ifind/bin64$ pwd
/home/litian/ifind/bin64
litian@litian-virtual-machine:~/ifind/bin64$ sudo python3 /home/litian/ifind/bin64/installiFinDPy.py /home/litian/ifind
/home/litian/ifind/bin64/installiFinDPy.py:1: DeprecationWarning: The distutils package is deprecated and slated for removal in
Python 3.12. Use setuptools or check PEP 632 for potential alternatives
  from distutils.core import setup
/home/litian/ifind/bin64/installiFinDPy.py:6: DeprecationWarning: The distutils.sysconfig module is deprecated, use sysconfig
instead
  from distutils.sysconfig import get_python_lib
Python is 64 bits
Installed into
/usr/lib/python3/dist-packages
OK!

```

## 2) 环境部署好后测试接口的使用

```
litian@litian-virtual-machine:~/ifind/bin64$ python3
Python 3.10.12 (main, Jun 11 2023, 05:26:28) [GCC 11.4.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> from iFinDPy import *
/usr/lib/python3/dist-packages/iFinDPy.pth
>>> THS_iFinDLogin('litian000','1234qwerws')
0
>>> THS_RQ('000001.SZ,600000.SH','open;high;low;latest;amount;volume')
errorcode=0
errmsg=Success!
data=
```

	time	thscore	open	high	low	latest	amount	volume
0	2023-10-09 15:28:30	000001.SZ	11.16	11.17	11.06	11.11	772959350.0	695828.0
1	2023-10-09 15:28:38	600000.SH	7.08	7.08	7.00	7.03	195214390.0	277631.0

### 2-2-2、Java

- 按照上述说明环境依赖查询已经齐全后，在编译使用前,请添加当前目录到系统环境变量 LD\_LIBRARY\_PATH（由于本程序使用到了配置文件,请勿复制到系统目录,否则会导致数据异常）

如解压之后的 bin64 所在目录为 /root/Linux

export LD\_LIBRARY\_PATH=\$LD\_LIBRARY\_PATH:/root/Linux/bin64

## 2) 修改示例代码登录函数、提取数据命令函数测试

```
import Ths.JDIBridge;

public class Demo6 {
    public static void main(String[] args) {
        System.out.println("=====JDI_demo6=====");

        System.loadLibrary("FTDataInterface");
        int ret = -1;

        //登入
        JDIBridge client = new JDIBridge();
        ret = client.THS_iFinDLogin("账号", "密码");//登录函数
        if(ret==0){
            System.out.println("login Successful!");
        }else{
            System.out.println("login Failed!");
        }

        //查询
        String dateString = client.THS_HistoryQuotes("000001.SZ", "open", "", "2023-09-09", "2023-10-09");//提取数据函数
        System.out.println(dateString);
    }
}
```

## 3) 引入 jar 包，使用示例代码测试

例如解压之后的 java 所在目录为/root/Linux

命令如下(注意命令间空格):

javac -cp /root/Linux/java/Ths/output/ThsJDI.jar test.java

java -cp /root/Linux/java/Ths/output/ThsJDI.jar: test

```
litian@litian-virtual-machine:~/ifind/java/Ths/output$ javac -cp /home/litian/ifind/java/Ths/output/ThsJDI.jar Demo6.java
litian@litian-virtual-machine:~/ifind/java/Ths/output$ java -cp /home/litian/ifind/java/Ths/output/ThsJDI.jar: Demo6
=====JDI_demo6=====
Login Successful!
{"errorCode":0,"errmsg":"","tables":[{"thscode":"000001.SZ","time":["2023-09-11","2023-09-12","2023-09-13","2023-09-14","2023-09-15","2023-09-18","2023-09-19","2023-09-20","2023-09-21","2023-09-22","2023-09-25","2023-09-26","2023-09-27","2023-09-28","2023-10-09"],"table":{"open":[11.3,11.35,11.29,11.35,11.22,11.22,11.2,11.14,11.03,11.24,11.22,11.15,11.19,11.16],"high":[11.4,11.35,11.32,11.33,11.4,11.24,11.23,11.22,11.14,11.25,11.32,11.26,11.24,11.17],"low":[11.27,11.25,11.21,11.25,11.2,11.14,11.16,11.15,11.04,11.03,11.19,11.14,11.15,11.18,11.06],"close":[11.34,11.28,11.25,11.29,11.23,11.21,11.15,11.05,11.24,11.22,11.16,11.17,11.2,11.11],"volume":[65448558,49548506,49021797,55165017,89047613,50132501,50366432,55887241,843936]}
```

## 2-2-3、C++

按照上述说明环境依赖查询已经齐全后，下载官网示例修改测试

下载地址：<http://quantapi.10jqka.com.cn/?page=sample>



### C++接口函数使用案例

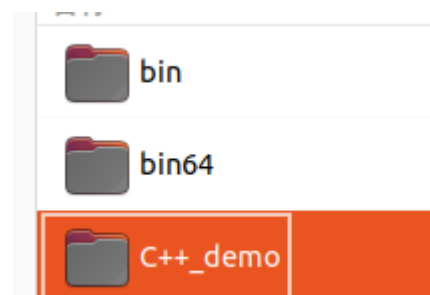
编程语言：C++ 上传日期：2022-02-12

文件大小：8.08MB 下载次数：2847次

简介：

演示C++各接口函数的使用，包括底层动态库函数定义调用及对返回的JSON格式数据进行获取并保存

解压 linux 安装包与 C++示例



修改示例、填写账号、密码

```
33
34 int main(int argc, char *argv[])
35 {
36     int nBitLong = sizeof(void *);
37 #ifdef _WIN32
38     //下面路径按照本地接口安装包路径修改
39     if (!InitialFunction("E://THSDataInterface_Windows//bin//x86//ShellExport"))
40     {
41         return 0;
42     }
43 #else
44     if (nBitLong == 8)
45     {
46         if (!InitialFunction("libFTDataInterface.so"))
47         {
48             return 0;
49         }
50     }
51     else
52     {
53         if (!InitialFunction("libFTDataInterface.so"))
54         {
55             return 0;
56         }
57     }
58 }
```

添加接口目录到系统环境变量(以 bin64 目录为例)

```
litian@litian-virtual-machine:~/ifind/C++_demo$ pwd
/home/litian/ifind/C++_demo
litian@litian-virtual-machine:~/ifind/C++_demo$ export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/home/litian/ifind/bin64
```

编译、测试接口示例

```
litian@litian-virtual-machine:~/ifind/C++_demo$ make -f makefile 编译
g++ test.o ExportFunction.o -o ./test -fPIC -lidn -L../bin -ldl -lpthread
litian@litian-virtual-machine:~/ifind/C++_demo$ ./test
result:{"errorCode":0,"errmsg":"","tables":[{"thscode":"300033.SZ","table":{"ths_stock_short_name_stock":77F},"ths_the_sw_industry_stock":["\u8BA1\u7B97\u673A"]}],{"thscode":"600000.SH","table":{"ths_stock_short_name_stock":["\u4E3B\u677F"],"ths_the_sw_industry_stock":["\u94F6\u884C"]}],datatype":[{"itemid":"ths_stock_short_name_stock","type":"DT_STRING"},{"itemid":"ths_the_sw_industry_stock","type":"DT_STRING"}],"input":{"name":"THSCODE","system":"false","value":""},{"name":"FDIR","system":"false","value":""}],{"function":"FDIR","system":"false","value":""}],{"function":"ths_listedsector_stock","id":"09235","params":[{"name":"ths_the_sw_industry_stock","id":"00444","params":[{"name":"THSCODE","system":"false","value":""},{"name":"","system":"false","value":""}]}]}],"dataVol":8,"perf":25}
```

## 3、 HTTP 接口使用说明

### 3-1、 HTTP 接口说明

#### 3-1-1、 refresh token 与 access token 说明

接口鉴权方案分为长期的 refresh\_token 和短期的 access\_token

##### 1) refresh token

refresh\_token 有效期与获取时账号有效期一致(后续账号有续期、权限变更必须重新获取来更新 refresh token 的有效期)。

refresh\_token 只用来请求当前有效的 access\_token 或者获取一个新的 access\_token。

refresh\_token 只能通过超级命令客户端“工具-refresh\_token 查询”获取。



注：超级命令使用介绍见本说明‘1-4、超级命令客户端介绍’

refresh token 更新后，所有环境过去的的 refresh\_token、access\_token 均会失效，更新 refresh\_token 相当于更改 HTTP 接口的账号密码。

##### 2) access token

access\_token 可以利用 refresh token 使用接口函数获取，在初次生成的七天后失效，如失效前三天重新获取 access\_token 会刷新失效时间。

access\_token 用来直接向同花顺服务器请求数据。

单个 access\_token 最多支持 20 个 IP，如报错 ‘Device exceed limit’ 可以更新 access token 重置绑定。

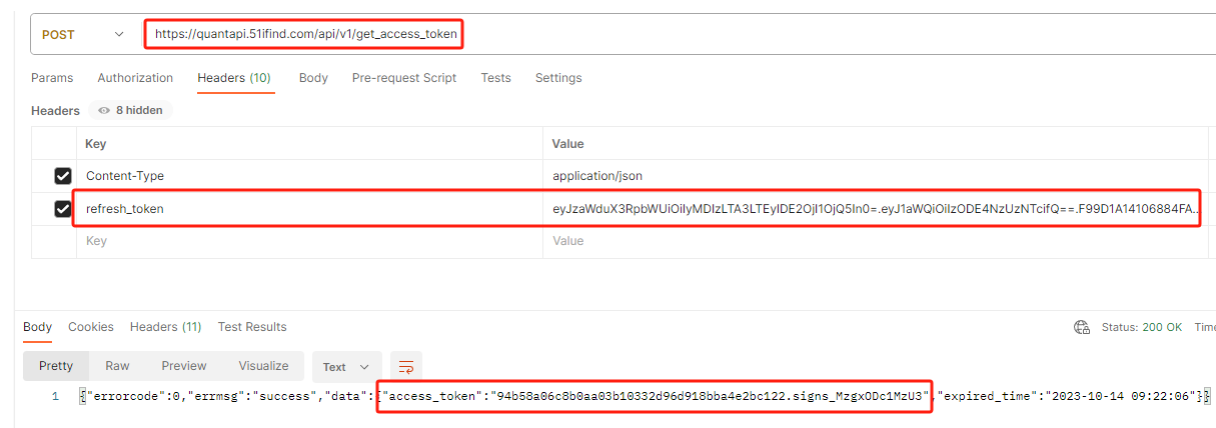
### 3-2、Postman 等工具使用 HTTP 接口

### 3-2-1、获取 access token

### 超级命令选择 HTTP 接口生成 get\_access\_token 函数的命令



超级命令获取到 refresh token 后利用接口获取最新的 access token





3-2-2、获取数据

超级命令查看取数函数命令

账号信息

Http

高频序列

实时行情

历史行情

基础数据

日期序列

板块成分及报表

经济数据库

sheet1

sheet2

+

实时行情-股票-开盘价:最高价:最低价:最新价:现额:现量-iFind数据接口

requestMethod:POST

requestURL:https://quantapi.51ifind.com/api/v1/real\_time\_quotation

requestHeaders:{"Content-Type":"application/json","access\_token":"94b58a06c8b0aa03b10332d96d918bba4e2b122signs\_MzgxODc1MzU3"}

formData:{"codes":"000001.SZ,600000.SH","indicators":"open,high,low,latest,latestAmount,latestVolume"}

sheet1

sheet2

+

number	time	thsoode	open	high	low	latest	latestAmount	latestVolume
1	2023-10-10 16:00:42	000001.SZ	11.1900	11.2000	11.0200	11.0200	18670084.0000	1694200.0000
2	2023-10-10 15:59:41	600000.SH	7.1300	7.1400	7.0100	7.0100	6517898.0000	929800.0000

填写参数获取数据

Params

Authorization

Headers (11)

Body

Pre-request Script

Tests

Settings

Headers

9 hidden

Key	Value
<input checked="" type="checkbox"/> Content-Type	application/json
<input checked="" type="checkbox"/> access_token	94b58a06c8b0aa03b10332d96d918bba4e2b122signs_MzgxODc1MzU3

POST

https://quantapi.51ifind.com/api/v1/real\_time\_quotation

Params

Authorization

Headers (11)

Body

Pre-request Script

Tests

Settings

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

JSON

1

2

3

4

1

2

3

4

Body

Cookies

Headers (11)

Test Results

Pretty

Raw

Preview

Visualize

Text

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

61

62

63

64

65

66

67

68

69

70

71

72

73

74

75

76

77

78

79

80

81

82

83

84

85

86

87

88

89

90

91

92

93

94

95

96

97

98

99

100

101

102

103

104

105

106

107

108

109

110

111

112

113

114

115

116

117

118

119

120

121

122

123

124

125

126

127

128

129

130

131

132

133

134

135

136

137

138

139

140

141

142

143

144

145

146

147

148

149

150

151

152

153

154

155

156

157

158

159

160

161

162

163

164

165

166

167

168

169

170

171

172

173

174

175

176

177

178

179

180

181

182

183

184

185

186

187

188

189

190

191

192

193

194

195

196

197

198

199

200

201

202

203

204

205

206

207

208

209

210

211

212

213

214

215

216

217

218

219

220

221

222

223

224

225

226

227

228

229

230

231

232

233

234

235

236

237

238

239

240

241

242

243

244

245

246

247

248

249

250

251

252

253

254

255

256

257

258

259

260

261

262

263

264

265

266

267

268

269

270

271

272

273

274

275

276

277

278

279

280

281

282

283

284

285

286

287

288

289

290

291

292

293

294

295

296

297

298

299

300

301

302

303

304

305

306

307

308

309

310

311

312

313

314

315

316

317

318

319

320

321

322

323

324

325

326

327

328

329

330

331

332

333

334

335

336

337

338

339

340

341

342

343

344

345

346

347

348

349

350

351

352

353

354

355

356

357

358

359

360

361

362

363

364

365

366

367

368

369

370

371

372

373

374

375

376

377

378

379

380

381

382

383

384

385

386

387

388

389

390

391

392

393

394

395

396

397

398

399

400

401

402

403

404

405

406

407

408

409

410

411

412

413

414

415

416

417

418

419

420

421

422

423

424

425

426

427

428

429

430

431

432

433

434

435

436

437

438

439

440

441

442

443

444

445

446

447

448

449

450

451

452

453

454

455

456

457

458

459

460

461

462

463

464

465

466

467

468

469

470

471

472

473

474

475

476

477

478

479

480

481

482

483

484

485

486

487

488

489

490

491

492

493

494

495

496

497

498

499

500

501

502

503

504

505

506

507

508

509

510

511

512

513

514

515

516

517

518

519

520

521

522

523

524

525

526

527

528

529

530

531

532

533

534

535

536

537

538

539

540

541

542

543

544

545

546

547

548

549

550

551

552

553

554

555

556

557

558

559

560

561

562

563

564

565

566

567

568

569

570

571

572

573

574

575

576

577

578

579

580

581

582

583

584

585

586

587

588

589

590

591

592

593

594

595

596

597

598

599

600

601

602

603

604

605

606

607

608

609

610

611

612

613

614

615

616

617

618

619

620

621

622

623

624

625

626

627

628

629

630

631

632

633

634

635

636

637

638

639

640

641

642

643

644

645

646

647

648

649

650

651

652

653

654

655

656

657

658

659

660

661

662

663

664

665

666

667

668

669

670

671

672

673

674

675

676

677

678

679

680

681

682

683

684

685

686

687

688

689

690

691

692

693

694

695

696

697

698

699

700

701

702

703

704

705

706

707

708

709

710

711

712

713

714

715

716

717

718

719

720

721

722

723

724

725

726

727

728

729

730

731

732

733

734

735

736

737

738

739

740

741

742

743

744

745

746

747

748

749

750

751

752

753

754

755

756

757

758

759

760

761

762

763

764

765

766

767

768

769

770

771

772

773

774

775

776

777

778

779

780

781

782

783

784

785

786

787

788

789

790

791

792

793

794

795

796

797

798

799

800

801

802

803

804

805

806

807

808

809

810

811

812

813

814

815

816

817

818

819

820

821

822

823

824

825

826

827

828

829

830

831

832

833

834

835

836

837

838

839

840

841

842

843

844

845

846

847

848

849

850

851

852

853

854

855

856

857

858

859

860

861

862

863

864

865

866

867

868

869

870

871

872

873

874

875

876

877

878

879

880

881

882

883

884

885

886

887

888

889

890

891

892

893

894

895

896

897

898

899

900

901

902

903

904

905

906

907

908

909

910

911

912

913

914

915

916

917

918

919

920

921

922

923

924

925

926

927

928

929

930

931

932

933

934

935

936

937

938

939

940

941

942

943

944

945

946

947

948

949

950

951

952

953

954

955

956

957

958

959

960

961

962

963

964

965

966

967

968

969

970

971

972

973

974

975

976

977

978

979

980

981

982

983

984

985

986

987

988

989

990

991

992

993

994

995

996

997

998

999

1000

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

61

62

63

64

65

66

67

68

69

70

71

72

73

74

75

76

77

78

79

80

81

82

83

84

85

86

87

88

89

90

91

92

93

94

95

96

97

98

99

100

101

102

103

104

105

106

107

108

109

110

111

112

113

114

115

116

117

118

119

120

121

122

123

124

125

126

127

128

129

130

131

132

133

134

135

136

137

138

139

140

141

142

143

144

145

146

147

148

149

150

151

152

153

154

155

156

157

158

159

160

161

162

163

164

165

166

167

168

169

170

171

172

173

174

175

176

177

178

179

180

181

182

183

184

185

186

187

188

189

190

191

192

193

194

195

196

197

198

199

200

201

202

203

204

205

206

207

208

209

210

211

212

213

214

215

216

217

218

219

220

221

222

223

224

225

226

227

228

229

230

231

232

233

234

235

236

237

238

239

240

241

242

243

244

245

246

247

248

249

250

251

252

253

254

255

256

257

258

259

260

261

262

263

264

265

266

267

268

269

270

271

272

273

274

275

276

277

278

279

280

281

282

283

284

285

286

287

288

289

290

291

292

293

294

295

296

297

298

299

300

301

302

303

304

305

306

307

308

309

310

311

312

313

314

315

316

317

318

319

320

321

322

323

324

325

326

327

328

329

330

331

332

333

334

335

336

337

338

339

340

341

342

343

344

345

346

347

348

349

350

351

352

353

354

355

356

357

358

359

360

361

362

363

364

365

366

367

368

369

370

371

372

373

374

375

376

377

378

379

380

381

382

383

384

385

386

387

388

389

390

391

392

393

394

395

396

397

398

399

400

401

402

403

404

405

406

407

408

409

410

411

412

413

### 3-3、Python 等语言环境使用 HTTP 接口

用户可以参考官网示例测试，下载地址：<http://quantapi.10jqka.com.cn/?page=sample>



#### HTTP接口应用案例(python环境)

编程语言：python

上传日期：2022-07-06

文件大小：8.48KB

下载次数：2715次

简介：

演示在Python环境使用HTTP接口提取各函数数据

#### 3-3-1、获取 access token

```
# Token accessToken 及权限校验机制
getAccessTokenUrl = 'https://quantapi.51ifind.com/api/v1/get_access_token'
# 获取refresh_token需下载Windows版本接口包解压，打开超级命令-工具-refresh_token查询
refreshToken = 'eyJzaWduX3RpbWUiOiIyMDIzLTA3LTEyIDE2OjI1OjQ5In0=.eyJ1aWQiOiIzODE4NzUzNTcifQ==.F94'
getAccessTokenHeader = {"Content-Type": "application/json", "refresh_token": refreshToken}
getAccessTokenResponse = requests.post(url=getAccessTokenUrl, headers=getAccessTokenHeader)
accessToken = json.loads(getAccessTokenResponse.content)['data']['access_token']
```

#### 3-3-2、获取数据

```
# 基础数据：获取证券基本信息、财务指标、盈利预测、日频行情等数据
def basic_data():
    thsUrl = 'https://quantapi.51ifind.com/api/v1/basic_data_service'
    thsPara = {"codes": "601318.SH", "indipara": [{"indicator": "ths_corp_cn_name_stock"}]}
    thsResponse = requests.post(url=thsUrl, json=thsPara, headers=thsHeaders)
    data = json.loads(thsResponse.content)
```