**GlobalQuant数字货币量化API**

**说明文档**

目录

[1. 火币网操作说明 3](#_Toc512112909)

[1.1. 借贷操作 3](#_Toc512112910)

[1.2. 下单操作 3](#_Toc512112911)

[做多BTC（不加杠杆） 3](#_Toc512112912)

[做多BTC（加杠杆） 4](#_Toc512112913)

[做空BTC 4](#_Toc512112914)

[2. API使用说明 5](#_Toc512112915)

[导入API 5](#_Toc512112916)

[API 调用示例 5](#_Toc512112917)

[3. 基本对象 5](#_Toc512112918)

[Instrument 交易标的信息 5](#_Toc512112919)

[Tick 逐笔行情数据 6](#_Toc512112920)

[Bar K线数据 7](#_Toc512112921)

[Order 委托订单 7](#_Toc512112922)

[Position 持仓 8](#_Toc512112923)

[Indicator 绩效指标 8](#_Toc512112924)

[4. 接口函数 9](#_Toc512112925)

[4.1. 行情数据接口 10](#_Toc512112926)

[get\_last\_ticks 获取最新的实时tick数据 10](#_Toc512112927)

[get\_last\_bars 获取最新的实时K线数据 11](#_Toc512112928)

[get\_ticks 获取历史tick数据 12](#_Toc512112929)

[get\_bars 获取最新的实时K线数据 12](#_Toc512112930)

[4.2. 账户持仓接口 13](#_Toc512112931)

[get\_accounts 查看账户信息 13](#_Toc512112932)

[get\_positions 查看当前总持仓 14](#_Toc512112933)

[get\_positions\_by\_symbol 查看某标的持仓 15](#_Toc512112934)

[4.3. 借贷接口 15](#_Toc512112935)

[apply\_margins 借贷 15](#_Toc512112936)

[repay\_margins 还贷 16](#_Toc512112937)

[get\_margin\_orders 查询借贷订单 17](#_Toc512112938)

[4.4. 交易接口 18](#_Toc512112939)

[open\_long 开多仓 18](#_Toc512112940)

[close\_long 开多仓 19](#_Toc512112941)

[margincash\_open 融资开多仓 19](#_Toc512112942)

[magincash\_close融资平多仓 20](#_Toc512112943)

[marginsec\_open 融券开空仓 21](#_Toc512112944)

[marginsec\_close 融券平空仓 22](#_Toc512112945)

[cancel\_order 取消订单 22](#_Toc512112946)

[get\_order 查询某指定订单的详情 23](#_Toc512112947)

[get\_orders\_by\_symbol 查询某指定标的历史订单详情 23](#_Toc512112948)

# 火币网操作说明

因为交易需支持做空，也就是必须在融资融券账户（margin-api）中交易，所以现金（USDT）和数字货币都先转入到融资融券账户中。

## 借贷操作

借贷规则：

借USDT日息0.1%，借币（BCC、ETH、LTC、ETC、DASH、XRP、EOS、OMG、ZEC）日息0.02%。自申请杠杆时起算，不足24小时按24小时计息，归还杠杆时归还借贷资金和利息。

APP 上日息都是 0.1%，最大支持2倍杠杆借贷，即1 BTC （或者等值的BTC加USDT总和）最大可借 2 BTC。

BTC最小借贷单位：0.001

USDT最小借贷单位：100

火币Pro用户进入借贷管理，选择借贷账户申请借贷资产，用户可申请借贷的数字货币数量取决于账户本金和平台杠杆倍数，用户**最大可申请借贷 = 净资产折合(BTC)×（倍数-1）- 已借贷资产，**例如：平台支持最高杠杆倍数为3倍，则用户可借贷数字货币的数量为本金的2倍。

风险率如何计算？

系统监听用户借贷账户的风险率=（总资产/借贷资产×100%）【注：总资产即借贷账户中的全部资产，折合成BTC计价】

## 下单操作

交易手续费 单边0.2%

BTC 最小买入单位 0.0001

USDT 最小买入单位 1

### 做多BTC（不加杠杆）

#### 开多仓

第一步：开多仓

假设此时1 BTC = 5000 USDT

使用5000 USDT买入1 BTC

手续费 = 5000 USDT \* 0.002 = 10 USDT

#### 平多仓

第一步：平多仓

假设此时 1 BTC = 6000 USDT，

卖出1 BTC，获得6000 USDT

手续费 = 1 BTC \* 0.002 = 0.002 BTC = 12 USDT

此笔交易收益 = 6000 - 5000 - 10 - 12 = 978 USDT

收益率 = 19.56%

### 做多BTC（加杠杆）

#### 开多仓

第一步：借入USDT

借入5000 USDT，此时欠平台5000 USDT

第二步：开多仓

假设此时1 BTC = 5000 USDT

使用10000 USDT买入2BTC

手续费 = 2\* 5000 USDT \* 0.002 = 20 USDT

#### 平多仓

第一步：平多仓

假设此时 1 BTC = 6000 USDT

卖出2 BTC，获得12000 USDT

手续费 = 2 BTC \* 0.002 = 0.004 BTC = 24 USDT

第二步：归还USDT

归还 5000 USDT

支付利息（持有1天）= 5000 USDT \* 0.001 = 5 USDT

此次交易收益 = 12000 - 5000 - 5000 - 20 - 24 - 5 = 1951 USDT

收益率 = 39.02%

如果持有10天

支付利息（持有10天）= 5000 USDT \* 0.001 \*10 = 50 USDT

此次交易收益 = 12000 - 5000 - 5000 - 20 - 24 - 50 = 1906 USDT

收益率 = 38.12%

### 做空BTC

做空BTC 必须通过融券卖出的方式。

#### 开空仓

第一步：借入BTC

借入1 BTC，此时欠平台1 BTC

第二步：卖出BTC

假设此时 1 BTC = 5000 USDT

卖出 2 BTC，获得 10000 USDT

手续费 = 2 BTC \* 0.002 = 0.004 BTC = 20 USDT

#### 平空仓

第一步：买入BTC

假设此时 1 BTC = 4000 USDT

花4000 USDT买入1 BTC 用于归还借贷

手续费 = 4000 USDT\*0.002 = 8 USDT

第二步：归还 BTC

归还平台1 BTC

支付利息（持有1日） = 1 BTC \* 0.002 = 0.002 BTC = 8 USDT

此笔交易收益 = 10000 - 5000 - 4000 - 20 - 8 - 8 = 964 USDT

收益率 = 19.28%

# 数据库

数据库访问地址

|  |  |  |
| --- | --- | --- |
| **信息字段** | **值** | **备注** |
| IP 地址 | 47.75.172.148 |  |
| 端口号 | 22 |  |
| 用户名 |  |  |
| 密码 |  |  |
| MongoDB版本 |  |  |

数据库清单

|  |  |  |  |
| --- | --- | --- | --- |
| **数据库database** | **数据表Collections** | **内容** | **备注** |
| btcmarket | btctick | Tick数据 |  |
| btcbar | Bar数据 |  |
| btctrade | 盘口竞价数据 |  |
| btcaccount | tradeorder | 交易订单记录 |  |
| loanorder | 借贷订单记录 |  |
| position | 持仓明细 |  |
| perform | 业绩表现 |  |
| btclog | syslog | 系统日志 |  |
| tradelog | 交易日志 |  |

## 行情数据库 dbbtcmarket

目前支持火币pro、币安（Binance）两个交易所。

|  |  |  |
| --- | --- | --- |
| **交易所** | **交易所代码** | **备注** |
| 火币Pro | hbp | 交易所数字货币代码格式为：usdt.hbp, btc.hbp |
| 币安 | bin | 交易所数字货币代码格式为：usdt.bin, btc.bin |

### Tick数据表 tbbtctick

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段** | **类型** | **含义** | **备注** | **示例** |
| sym | str | 数币代码 | 交易所数币代码.交易所代码 | usdt.hbp, btc.hbp, bch.bin, eth.bin |
| exg | str | 交易所代码 | hbp：火币pro  bin：币安 |  |
| symex | str | 交易所数币代码 |  | usdt, btc, bch, eth |
| cur | str | 计价数币 | 一般以USDT计价 |  |
| pre | float | 前收盘价 |  |  |
| utc | int | 时间戳 |  |  |
| time | str | 时间 |  |  |
| op | float | 开盘价 |  |  |
| hp | float | 最高价 |  |  |
| lp | float | 最低价 |  |  |
| cp | float | 收盘价 |  |  |
| vol | float | 成交量 |  |  |
| amt | float | 成交额 |  |  |

### Bar数据表tbbtcbar

数据表名：btcbar

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段** | **类型** | **含义** | **备注** | **示例** |
| sym | str | 数币代码 | 交易所数币代码.交易所代码 | usdt.hbp, btc.hbp, bch.bin, eth.bin |
| exg | str | 交易所代码 | hbp：火币pro  bin：币安 |  |
| symex | str | 交易所数币代码 |  | usdt, btc, bch, eth |
| type | str | BAR类型 | ‘1min’, ‘5min’, ‘15min’, ‘30min’, ‘60min’, ‘1d’ |  |
| utc | int | 时间戳 |  |  |
| time | str | 时间 |  |  |
| pre | float | 前收盘价 |  |  |
| op | float | 开盘价 |  |  |
| hp | float | 最高价 |  |  |
| lp | float | 最低价 |  |  |
| cp | float | 收盘价 |  |  |
| vol | float | 成交量 |  |  |
| amt | float | 成交额 | 一般以USDT计价 |  |
| adj | float | 复权因子 | 预留字段 |  |

### 买卖竞价明细表 tbbtcbid

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **字段** | | **类型** | **含义** | **备注** | **示例** |
| sym | | str | 数币代码 |  |  |
| exg | | str | 交易所 |  |  |
| symex | | str | 交易所数币代码 | ‘1min’, ‘5min’, ‘15min’, ‘30min’, ‘60min’, ‘1d’ |  |
| bar | | str | BAR类型 |  |  |
| utc | int | | 时间戳 |  |  |
| time | str | | 时间 |  |  |
| bid1 | | float | 买一价 |  |  |
| bid2 | | float | 买二价 |  |  |
| bid3 | | float | 买三价 |  |  |
| bid4 | | float | 买四价 |  |  |
| bid5 | | float | 买五价 |  |  |
| ask1 | | float | 卖一价 |  |  |
| ask2 | | float | 卖二价 |  |  |
| ask3 | | float | 卖三价 |  |  |
| ask4 | | float | 卖四价 |  |  |
| ask5 | | float | 卖五价 |  |  |

## 用户数据库 dbbtcaccount

### 交易订单表 tbtradeorder

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段** | **类型** | **含义** | **备注** | **示例** |
| usr | str | 用户ID | usr+创建时间戳+6位随机码 |  |
| sch | str | 策略ID | scheme+创建时间戳+6位随机码 |  |
| cur | str | 计价币种 | 一般以usdt或者btc计价 |  |
| exg | str | 交易所 | hbp：火币pro，bin：币安 |  |
| usrex | int | 交易所用户ID |  |  |
| ordid | int | 交易所订单ID |  |  |
| ordtime | str | 下单时间 |  |  |
| ordutc | int | 下单时间戳 |  |  |
| sym | str | 数币代码 |  |  |
| dir | int | 交易方向 | 1：多仓  0：平仓  -1：空仓 |  |
| ptype | int | 订单类型 | 0：市价单  1：限价单 |  |
| vol | float | 委托数量 |  |  |
| price | float | 委托价格 |  |  |
| amt | float | 委托金额 | =持仓数量\*Pirce(数币)/Price(计价币) |  |
| filltime | str | 成交时间 |  |  |
| fillvol | float | 成交数量 |  |  |
| fillprice | float | 成交价格 |  |  |
| fillamt | float | 成交金额 |  |  |
| fee | float | 手续费 |  |  |
| status | int | 订单状态 |  |  |
| mgid | int | 融资订单 |  |  |
| mgcur | str | 融资币种 |  |  |
| mgvol | float | 融资金额 | 如果是借币，金额为正  如果是还币，金额为负 |  |

### 借贷订单表 tbloadorder

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段** | **类型** | **含义** | **备注** | **示例** |
| usr | str | 用户ID |  |  |
| sch | str | 策略ID |  |  |
| cur | str | 计价币种 | 一般以usdt或者btc计价 |  |
| exg | str | 交易所 |  |  |
| usrex | int | 交易所用户ID |  |  |
| loanid | int | 交易所订单ID |  |  |
| loantime | str | 下单时间 |  |  |
| loanutc | int | 下单戳 |  |  |
| sym | str | 数币代码 |  |  |
| dir | int | 交易方向 | 1：借币  0：还币 |  |
| vol | float | 借贷数量 |  |  |
| amt | float | 借贷金额 | =持仓数量\*Price(数币)/Price(计价币) |  |
| interest | float | 借贷利息 | 每日更新 |  |
| status | int | 订单状态 |  |  |

### 持仓数据表 tbposition

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段** | **类型** | **含义** | **备注** | **示例** |
| usr | str | 用户ID |  |  |
| sch | str | 策略ID |  |  |
| cur | str | 计价币种 | 一般以usdt或者btc计价 |  |
| time | str | 时间 | 每次交易后，持仓自动更新，作为新一条记录 |  |
| utc | int | 时间戳 |  |  |
| sym | str | 持仓币种 | 格式为“交易所数币代码.交易所代码” |  |
| vol | float | 持仓数量 |  |  |
| amt | float | 持仓金额 | =持仓数量\*Pirce(数币)/Price(计价币) |  |
| price | float | 持仓成本 | 下单成交时价格 |  |
| loansym | str | 借贷币种 |  |  |
| loanamt | float | 借贷金额 |  |  |
| interest | float | 借贷利息 |  |  |

### 业绩表现表 tbperform

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段** | **类型** | **含义** | **备注** | **示例** |
| usr | str | 用户ID |  |  |
| sch | str | 策略ID |  |  |
| cur | str | 计价币种 | 一般以usdt或者btc计价 |  |
| time | str | 时间 | 每次交易后，持仓自动更新，作为新一条记录  每日00:00:00 统计一条新记录 |  |
| utc | int | 时间戳 |  |  |
| tot | float | 总计金额 | =总持仓金额-总借贷金额-借贷利息 |  |
| nav | float | 单位净值 |  |  |
| navacc | float | 累计净值 |  |  |
| ret | float | 累计收益率 |  |  |
| annret | float | 年化收益率 |  |  |
| std | float | 累计波动率 |  |  |
| maxdd | float | 最大回撤 |  |  |
| win | float | 胜率 |  |  |
| sharpe | float | 夏普率 |  |  |
| retvsdd | float | 收益回撤比 |  |  |

## 系统数据库tbbtclog

### 系统日志数据表 tbsyslog

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段** | **类型** | **含义** | **示例** | **备注** |
| usr | str | 用户ID |  |  |
| sch | str | 策略ID |  |  |
| time | str | 时间 |  |  |
| utc | int | 时间戳 |  |  |
| type | int | 日志类型 | 0：  1：  2： |  |
| msg | str | 日志内容 |  |  |

### 交易日志数据表 tbtradelog

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段** | **类型** | **含义** | **示例** | **备注** |
| usr | str | 用户ID |  |  |
| sch | str | 策略ID |  |  |
| time | str | 时间 |  |  |
| utc | int | 时间戳 |  |  |
| type | int | 日志类型 | 0：  1：  2： |  |
| msg | str | 日志内容 |  |  |

# API使用说明

### 导入API

import api.quant\_api as qapi

### API 调用示例

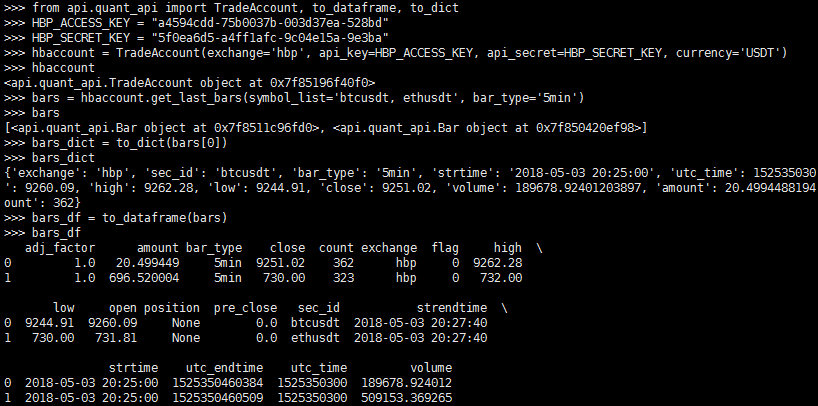
tick\_data = qapi.get\_last\_ticks(exchange='huobipro', symbol\_list=['btcusdt', 'ethusdt'])

# 基本对象

### TradeAccount 交易账户

交易账户类，支持火币（hbp）、币安（bnb）

**class** TradeAccount(object):  
 *'''  
 交易账户对象  
 '''* exchange = **''** *# hbp: 火币pro, bnb: Binance, okc: okcoin* api\_key = **''** api\_secret = **''** client = **None** currency = **''** *# 以哪种数币作为基准，比如USDT、BTC* **def** \_\_init\_\_(self, exchange, api\_key, api\_secret, currency=**'USDT'**):  
 *'''  
 指定交易所账户* **:param** *exchange:* **:param** *api\_key:* **:param** *api\_secret:* **:param** *currency:  
 '''* self.exchange = exchange  
 self.api\_key = api\_key  
 self.api\_secret = api\_secret  
 self.currency = currency



### Instrument 交易标的信息

交易代码数据类型

**class** **Instrument**(object):

**def** **\_\_init\_\_**(self):

self.exchange = '' *## 交易代码*

self.symbol = '' *## 交易代码*

self.sec\_type = 0 *## 代码类型*

self.sec\_name = '' *## 代码名称* self.base\_currency = **''** *## 基础币种* self.quote\_currency = '' *## string 计价币种* self.price\_precision = 0 *## string 价格精度位数（0为个位）* self.amount\_precision = 0  *## string 数量精度位数（0为个位）* self.symbol\_partition = '' *## string 交易区 main主区，innovation创新区，bifurcation分叉区*

### Tick 逐笔行情数据

逐笔行情数据。

**class** **Tick**(object):

**def** **\_\_init\_\_**(self):

self.exchange = '' *## 交易所代码*

self.sec\_id = '' *## 证券ID*

self.utc\_time = 0.0 *## 行情时间戳*

self.strtime = '' *## 可视化时间*

self.last\_price = 0.0 *## 最新价*

self.open = 0.0 *## 开盘价*

self.high = 0.0 *## 最高价*

self.low = 0.0 *## 最低价*

self.cum\_volume = 0.0 *## 成交总量/最新成交量,累计值*

self.cum\_amount = 0.0 *## 成交总金额/最新成交额,累计值*

self.cum\_position = 0.0 *## 合约持仓量(期),累计值*

self.last\_volume = 0 *## 瞬时成交量(中金所提供)*

self.last\_amount = 0.0 *## 瞬时成交额*

self.upper\_limit = 0.0 *## 涨停价*

self.lower\_limit = 0.0 *## 跌停价*

self.settle\_price = 0.0 *## 今日结算价*

self.trade\_type = 0 *## (保留)交易类型,对应多开,多平等类型 0:'上一个tick没有成交量', 1:'双开', 2: '双平', 3: '多开', 4:'空开', 5: '空平', 6:'多平', 7:'多换', 8:'空换'*

self.pre\_close = 0.0 *## 昨收价*

self.bids = [] *## [(price, volume), (price, volume), ...] ## 1-5档买价,量*

self.asks = [] *## [(price, volume), (price, volume), ...] ## 1-5档卖价,量*

### Bar K线数据

各种周期的Bar数据。

**class** **Bar**(object):

**def** **\_\_init\_\_**(self):

self.exchange = '' *## 交易所代码*

self.sec\_id = '' *## 证券ID*

self.bar\_type = 0 *## bar类型，’1min’,’5min’, '15min', '30min', '60min', '1day', '1mon', '1week', '1year'*

self.strtime = '' *## Bar开始时间*

self.utc\_time = 0.0 *## Bar开始时间*

self.strendtime = '' *## Bar结束时间*

self.utc\_endtime = 0.0 *## Bar结束时间*

self.open = 0.0 *## 开盘价*

self.high = 0.0 *## 最高价*

self.low = 0.0 *## 最低价*

self.close = 0.0 *## 收盘价*

self.volume = 0.0 *## 成交量*

self.amount = 0.0 *## 成交额*

### Order 委托订单

委托订单。

**class** **Order**(object):

**def** **\_\_init\_\_**(self):

self.account\_id = '' *## 用户账号*

self.scheme\_id = '' *## 策略账号*

self.exchange = '' *## 交易所代码*

self.order\_id = '' *## 交易所订单ID*

self.sec\_id = '' *## 证券ID*

self.position\_effect = 0 *## 开平标志*

self.side = 0 *## 买卖方向*

self.order\_type = 0 *## 订单类型，*

self.order\_src = 0 *## 订单来源，’api’, ‘magin-api’*

self.status = 0 *## 订单状态*

self.ord\_rej\_reason = 0 *## 订单拒绝原因*

self.ord\_rej\_reason\_detail = '' *## 订单拒绝原因描述*

self.price = 0.0 *## 委托价*

self.stop\_price = 0.0; *## 止损价*

self.volume = 0.0 *## 委托量*

self.filled\_volume = 0.0 *## 已成交量*

self.filled\_vwap = 0.0 *## 已成交均价*

self.filled\_amount = 0.0 *## 已成交额*

self.sending\_time = 0.0 *## 委托下单时间*

self.transact\_time = 0.0 *## 最新一次成交时间*

### Position 持仓

持仓。

**class** **Position**(object):

**def** **\_\_init\_\_**(self):

self.account\_id = '' *## 账户id*

self.exchange = '' *## 交易所代码*

self.sec\_id = '' *## 证券ID*

self.side = 0 *## 买卖方向*

self.volume = 0.0 *## 总持仓量*

self.available = 0.0 *## 可使用量*

self.order\_frozen = 0.0 *## 被冻结量*

self.loan = 0.0 *## 借贷金额*

self.interest = 0.0 *## 借贷利息*

self.loan\_order\_id = ‘’ *## 借贷资产的订单号*

此处，多单（side=1）表示持有该类资产（包括usdt和数字货币）

空单（side=-1）表示持有该类资产的借贷

在火币网上，借贷资产偿还必须按照订单号偿还，因此需记录订单号。

### Indicator 绩效指标

绩效。

**class** **Indicator**(object):

**def** **\_\_init\_\_**(self):

self.strategy\_id = '' *## 策略ID*

self.account\_id = '' *## 账号ID*

self.nav = 0.0 *## 净值(cum\_inout + cum\_pnl + fpnl - cum\_commission)*

self.pnl = 0.0 *## 净收益(nav-cum\_inout)*

self.profit\_ratio = 0.0 *## 收益率(pnl/cum\_inout)*

self.profit\_ratio\_bench = 0.0 *## 基准收益率*

self.sharp\_ratio = 0.0 *## 夏普比率*

self.risk\_ratio = 0.0 *## 风险比率*

self.trade\_count = 0 *## 交易次数*

self.win\_count = 0 *## 盈利次数*

self.lose\_count = 0 *## 亏损次数*

self.win\_ratio = 0.0 *## 胜率*

self.max\_profit = 0.0 *## 最大收益*

self.min\_profit = 0.0 *## 最小收益*

self.max\_single\_trade\_profit = 0.0 *## 最大单次交易收益*

self.min\_single\_trade\_profit = 0.0 *## 最小单次交易收益*

self.daily\_max\_single\_trade\_profit = 0.0 *## 今日最大单次交易收益*

self.daily\_min\_single\_trade\_profit = 0.0 *## 今日最小单次交易收益*

self.max\_position\_value = 0.0 *## 最大持仓市值或权益*

self.min\_position\_value = 0.0 *## 最小持仓市值或权益*

self.max\_drawdown = 0.0 *## 最大回撤*

self.daily\_pnl = 0.0 *## 今日收益*

self.daily\_return = 0.0 *## 今日收益率*

self.annual\_return = 0.0 *## 年化收益率*

self.cum\_inout = 0.0 *## 累计出入金*

self.cum\_trade = 0.0 *## 累计交易额*

self.cum\_pnl = 0.0 *## 累计平仓收益(没扣除手续费)*

self.cum\_commission = 0.0 *## 累计手续费*

self.transact\_time = 0.0 *## 指标计算时间*

# 接口函数

## 行情数据接口

### get\_last\_ticks 获取最新的实时tick数据

* **函数格式**

get\_last\_ticks(exchange, symbol\_list)

* **输入参数**

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 类型 | 释义 | 备注 |
| exchange | str | 交易所名称 | 火币：’huobipro’ |
| symbol\_list | str | 交易代码，支持1个或多个交易代码，多个代码用逗号分隔 | 'btcusdt, ethusdt' |

* **输出结果**

Tick 对象列表

* **示例：**

data = qapi.get\_last\_ticks (exchange='huobipro', symbol\_list='btcusdt, ethusdt')

Out: [<api.quant\_api.Tick at 0x5c265e3198>, <api.quant\_api.Tick at 0x5c265e3668>]

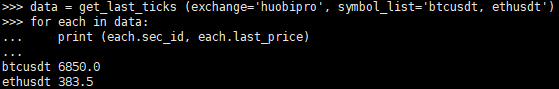
for each in data:

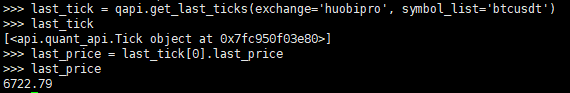
print (each.sec\_id, each.last\_price)

Out:

btcusdt 6835.37

ethusdt 393.03





### get\_last\_bars 获取最新的实时K线数据

* **函数格式**

get\_last\_bars (exchange, symbol\_list, bar\_type)

* **输入参数**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **类型** | **释义** | **备注** |
| exchange | str | 交易所名称 | 火币：’huobipro’ |
| symbol\_list | str | 交易代码，支持1个或多个交易代码 | ‘btcusdt’, ‘ethusdt’, ‘btcusdt’ |
| bar\_type | str | K线周期类型 | '1min', '5min', '15min', '30min', '60min', '1day', '1mon', '1week', '1year' |

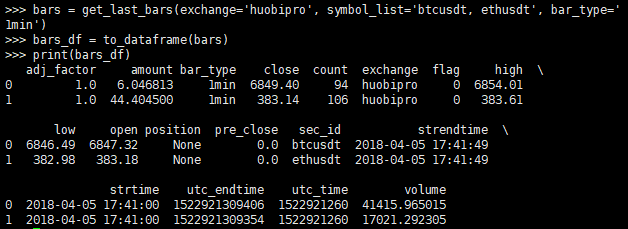
* **输出结果**

Bar 对象列表

* **示例：**

data = qapi.get\_last\_bars(exchange='huobipro', symbol\_list='btcusdt, ethusdt', bar\_type='1min')

Out: [<api.quant\_api.Bar at 0x5c265ec860>, <api.quant\_api.Bar at 0x5c265ecac8>]



### get\_ticks 获取历史tick数据

* **函数格式**

get\_ticks (exchange, symbol\_list, begin\_time=“”, end\_time=“”, size=0)

注意：目前只支持当前往前推历史2000个tick

* **输入参数**

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 类型 | 释义 | 备注 |
| exchange | str | 交易所名称 | 火币：’huobipro’ |
| symbol\_list | str | 交易代码列表，支持1个或多个交易代码 |  |
| begin\_time | str | 开始时间，如2017-10-30 09:30:00 |  |
| end\_time | str | 结束时间，如2017-10-30 15:00:00 |  |
| size | int | 从当前时间向前倒退size个tick | 可选 |

size 和 begin\_time/end\_time 为两种时间区间方式，二选一。

* **输出结果**

Tick对象列表

* **示例：**

data = qapi.get\_ticks(exchange='huobipro', symbol\_list='btcusdt', 'ethusdt', begin\_time=’’’2018-03-20 00:00:00’, end\_time=’’2018-03-20 02:00:00)

### get\_bars 获取最新的实时K线数据

* **函数格式**

get\_last\_bars(exchange, symbol\_list, bar\_type, begin\_time=“”, end\_time=“”, size=0)

注意：目前只支持当前往前推2000个bar

* **输入参数**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **类型** | **释义** | **备注** |
| exchange | str | 交易所名称 | 火币：’huobipro’ |
| symbol\_list | str | 交易代码列表，支持1个或多个交易代码 | ‘btcusdt’, ‘ethusdt’, ‘btcusdt’ |
| bar\_type | str | K线周期类型 | '1min', '5min', '15min', '30min', '60min', '1day', '1mon', '1week', '1year' |
| begin\_time | str | 开始时间，如2017-10-30 09:30:00 |  |
| end\_time | str | 结束时间，如2017-10-30 15:00:00 |  |
| size | int | 从当前时间向前倒退size个bar |  |

* **输出结果**

Bar 对象列表

* **示例：**

data = qapi.get\_bars(exchange='huobipro', symbol\_list='btcusdt, ethusdt', bar\_type= '1min', begin\_time=’’’2018-03-20 00:00:00’, end\_time=’’2018-03-20 02:00:00)

data = qapi.get\_bars(exchange='huobipro', symbol\_list='btcusdt, ethusdt', bar\_type= '1min', size=20)

## 账户持仓接口

### get\_accounts 查看账户信息

* **函数格式**

get\_accounts (exchange)

* **输入参数**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **类型** | **释义** | **备注** |
| exchange | str | 交易所名称 | 火币：’huobipro’ |

* **输出结果**

用户的账户信息，包括普通账户（spot）、融资融券账户（margin）和场外交易账户（otc）

* **示例：**

data = qapi.get\_accounts (exchange= 'huobipro')

{'margin': {'id': 3105263, 'state': 'working', 'subtype': 'btcusdt', 'type': 'margin'},

'otc': {'id': 3065073, 'state': 'working', 'subtype': '', 'type': 'otc'},

'spot': {'id': 2828146, 'state': 'working', 'subtype': '', 'type': 'spot'}}



### get\_positions 查看当前总持仓

* **函数格式**

get\_positions (exchange='huobipro', source='api')

* **输入参数**

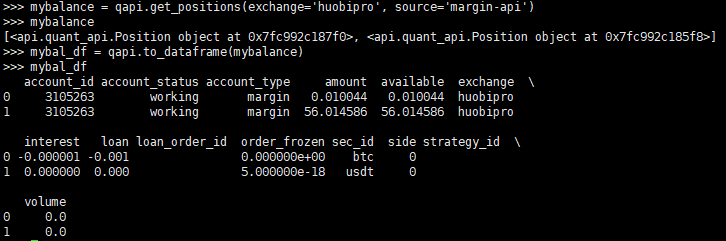
|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **类型** | **释义** | **备注** |
| exchange | str | 交易所名称 | 火币：’huobipro’ |
| source | str | 账户类型 | ‘api’：普通账户  ‘margin-api’：借贷账户 |

* **输出结果**

委托下单生成的Order对象

* **示例：**

data = qapi.get\_positions (exchange=’huobipro’, source='api')



### get\_positions\_by\_symbol 查看某标的持仓

* **函数格式**

get\_positions\_by\_symbol (exchange='huobipro', source='api', sec\_id='usdt')

* **输入参数**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **类型** | **释义** | **备注** |
| exchange | str | 交易所名称 | 火币：’huobipro’ |
| source | str | 账户类型 | ‘api’：普通账户  ‘margin-api’：借贷账户 |
| sec\_id | str | 交易代码 |  |

* **输出结果**

委托下单生成的Order对象

* **示例：**

data = qapi.get\_last\_ticks(exchange='huobipro', symbol\_list=['btcusdt', 'ethusdt'])

## 借贷接口

### apply\_margins 借贷

* **函数格式**

apply\_margin(exchange='huobipro', symbol='btcusdt', currency='btc', amount=0)

* **输入参数**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **类型** | **释义** | **备注** |
| exchange | str | 交易所名称 | 火币：’huobipro’ |
| symbol | str | 交易代码 |  |
| currency | str | 借贷资产类型 | 目前支持 btc 和 usdt |
| amount | float | 借入数量 | btc 最小 0.001  usdt 最小 100 |

* **输出结果**

借贷成功，返回订单号，6位整数

借贷失败，返回False

* **示例：**

result = qapi.apply\_margin(exchange=**'huobipro'**, symbol=**'btcusdt'**, currency=**'btc'**, amount=0.001)



必须先把旧订单还清后，才可以借新订单



### repay\_margins 还贷

* **函数格式**

repay\_margin(exchange='huobipro', order\_id=None, amount=0)

* **输入参数**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **类型** | **释义** | **备注** |
| exchange | str | 交易所名称 | 火币：’huobipro’ |
| order\_id | int | 订单号 |  |
| amount | float | 借入数量 | btc 最小 0.001  usdt 最小 100 |

* **输出结果**

还款成功：True

还款失败：False

* **示例：**

result = qapi.repay\_margin(exchange=**'huobipro'**, order\_id=676027, amount=0.15)



### get\_margin\_orders 查询借贷订单

* **函数格式**

get\_margin\_orders(exchange='huobipro', symbol='btcusdt', currency='btc', start\_date="", end\_date="", start="", direct="", size=0)

* **输入参数**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **类型** | **释义** | **备注** |
| exchange | str | 交易所名称 | 火币：’huobipro’ |
| symbol | int | 订单号 |  |
| currency | float | 借入数量 | btc 最小 0.001  usdt 最小 100 |
| start\_date | str | 查询开始日期 | 日期格式yyyy-mm-dd |
| end\_date | str | 查询结束日期 | 日期格式yyyy-mm-dd |
| start | str | 查询基准日期 | 日期格式yyyy-mm-dd，包括基准日期 |
| direct | str | 查询方向 | prev 向前，next 向后 |
| size | int | 查询记录大小 |  |

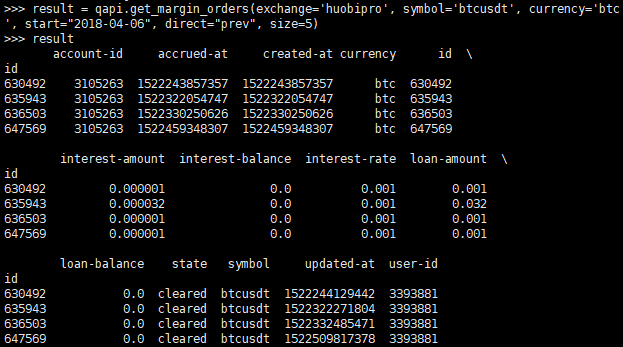
start\_date、end\_date 与 start、direct、size 两种查询方式二选一

* **输出结果**

订单详细信息，DataFrame 格式

* **示例：**

margin\_orders = qapi.get\_margin\_orders(exchange=**'huobipro'**, symbol=**'btcusdt'**, currency=**'btc'**, start=**"2018-04-06"**, direct=**"prev"**, size=5)



## 交易接口

### open\_long 开多仓

* **函数格式**

open\_long (exchange, sec\_id, price, volume)

* **输入参数**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **类型** | **释义** | **备注** |
| exchange | str | 交易所名称 | 火币：’huobipro’ |
| sec\_id | str | 交易代码 |  |
| price | float | 委托价格 | 如果price=0，为市价单  price>0，限价单 |
| volume | float | 委托量 | 下单的数字货币数量\* |

\* 注意此处和火币官网不同，火币市价买单是下单金额，被转换成下单数量，考虑到下单的滑点问题，实际下单数量可能略有偏差。

* **输出结果**

委托下单生成的Order对象

* **示例：**

myorder = qapi.open\_long(exchange=**'huobipro'**, source=**'margin-api'**, sec\_id=**'btcusdt'**, price=0, volume=0.001)



### close\_long 开多仓

* **函数格式**

close\_long (exchange, sec\_id, price, volume)

* **输入参数**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **类型** | **释义** | **备注** |
| exchange | str | 交易所名称 | 火币：’huobipro’ |
| sec\_id | str | 交易代码 |  |
| price | float | 委托价格 | 如果price=0，为市价单  price>0，限价单 |
| volume | float | 委托量 | 限价单表示下单数量，  市价买单时表示买多少钱，市价卖单时表示卖多少币 |

* **输出结果**

委托下单生成的Order对象

* **示例：**

myorder = qapi.close\_long(exchange=**'huobipro'**, source=**'margin-api'**, sec\_id=**'btcusdt'**, price=0, volume=0.002)



### margincash\_open 融资开多仓

* **函数格式**

margincash\_open (exchange='huobipro', sec\_id='btcusdt', price=0, volume=0, leverage=0)

* **输入参数**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **类型** | **释义** | **备注** |
| exchange | str | 交易所名称 | 火币：’huobipro’ |
| sec\_id | str | 交易代码 |  |
| price | float | 委托价格 | 如果price=0，为市价单  price>0，限价单 |
| volume | float | 委托量 |  |
| leverage | float | 融资比例 | 取值范围0~2。  比如需要融资本金的1.5倍，leverage=1.5 |

* **输出结果**

委托下单生成的Order对象

* **示例：**

myorder = qapi.margincash\_open(exchange=**'huobipro'**, sec\_id=**'btcusdt'**, price=0, volume=50, leverage=2)



### magincash\_close融资平多仓

* **函数格式**

close\_short (exchange, sec\_id, price, volume)

* **输入参数**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **类型** | **释义** | **备注** |
| exchange | str | 交易所名称 | 火币：’huobipro’ |
| sec\_id | str | 交易代码 |  |
| price | float | 委托价格 | 如果price=0，为市价单  price>0，限价单 |
| volume | float | 委托量 |  |

* **输出结果**

委托下单生成的Order对象

* **示例：**

data = qapi.get\_last\_ticks(exchange='huobipro', symbol\_list=['btcusdt', 'ethusdt'])

### marginsec\_open 融券开空仓

* **函数格式**

marginsec\_open (exchange='huobipro', sec\_id='btcusdt', price=0, volume=0)

* **输入参数**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **类型** | **释义** | **备注** |
| exchange | str | 交易所名称 | 火币：’huobipro’ |
| sec\_id | str | 交易代码 |  |
| price | float | 委托价格 | 如果price=0，为市价单  price>0，限价单 |
| volume | float | 委托量 | 数字货币数量 |

* **输出结果**

委托下单生成的Order对象

* **示例：**

myorder = qapi.marginsec\_open(exchange=**'huobipro'**, sec\_id=**'btcusdt'**, price=0, volume=0.001)



### marginsec\_close 融券平空仓

* **函数格式**

marginsec\_close (exchange='huobipro', sec\_id='btcusdt', price=0, volume=0, margin\_order\_id='')

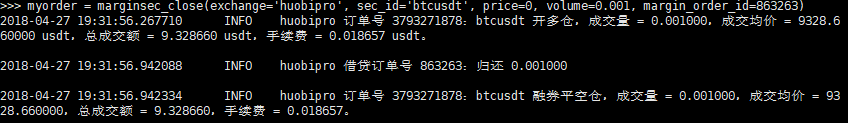
* **输入参数**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **类型** | **释义** | **备注** |
| exchange | str | 交易所名称 | 火币：’huobipro’ |
| sec\_id | str | 交易代码 |  |
| price | float | 委托价格 | 如果price=0，为市价单  price>0，限价单 |
| volume | float | 委托量 |  |

* **输出结果**

委托下单生成的Order对象

* **示例：**



### cancel\_order 取消订单

* **函数格式**

cancel\_order (exchange, cl\_ord\_id=None)

* **输入参数**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **类型** | **释义** | **备注** |
| exchange | str | 交易所名称 | 火币：’huobipro’ |
| cl\_ord\_id | int | 交易所订单ID |  |

* **输出结果**

查询返回的Order对象

* **示例：**

data = qapi.get\_last\_ticks(exchange='huobipro', symbol\_list=['btcusdt', 'ethusdt'])

### get\_order 查询某指定订单的详情

* **函数格式**

get\_order (exchange, cl\_ord\_id)

* **输入参数**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **类型** | **释义** | **备注** |
| exchange | str | 交易所名称 | 火币：’huobipro’ |
| cl\_ord\_id | int | 交易所订单ID |  |

* **输出结果**

查询返回的Order对象

* **示例：**

data = qapi.get\_last\_ticks(exchange='huobipro', symbol\_list=['btcusdt', 'ethusdt'])

### get\_orders\_by\_symbol 查询某指定标的历史订单详情

* **函数格式**

get\_orders\_by\_symbol (exchange, sec\_id, start\_time, end\_time, states='filled', types=None)

* **输入参数**

|  |  |  |  |
| --- | --- | --- | --- |
| **参数** | **类型** | **释义** | **备注** |
| exchange | str | 交易所名称 | 火币：**'huobipro'** |
| cl\_ord\_id | int | 交易所订单ID |  |
| start\_time | str | 开始时间 | **'%Y-%m-%d %H:%M:%S'** |
| end\_time | str | 结束时间 | **'%Y-%m-%d %H:%M:%S'** |
| states | str | 订单状态 | **'pre-submitted'** 准备提交  **'submitted'** 已提交  partial-filled 部分成交  partial-canceled 部分成交撤销  filled 完全成交  canceled 已撤销 |
| types | str | 订单类型 | buy-market：市价买  sell-market：市价卖  buy-limit：限价买  sell-limit：限价卖 |

* **输出结果**

查询返回的Order对象

* **示例：**

data = qapi.get\_last\_ticks(exchange='huobipro', symbol\_list=['btcusdt', 'ethusdt'])

## 数据库函数

### dbconn 连接数据库

### tick2db tick数据入库

### bar2db bar数据入库

### bid2db 竞价数据入库

### order2db 交易记录入库

### position2db 持仓数据入库

### perform2db 业绩表现数据入库

### log2db 日志数据入库

## 格式转换函数

### trans\_sym 代码格式转换

### obj2dict 对象转成字典

### obj2df 对象转成DataFrame