

Mt4Api Dev Team

[Mt4Api 接口 开发手册 C++]

[多账户版]

[版本 1.0.0.1]

[2019-06-12]

广州汇酬信息科技有限公司

目 录

1	简述	5
2	基础知识	6
2.1	mt4	6
2.2	常用术语	6
3	API 使用流程	8
4	接口函数	8
4.1	实例创建销毁接口	8
4.1.1	MT4API_Create	8
4.1.2	MT4API_Destroy	8
4.2	错误获取接口	9
4.2.1	MT4API_GetLastError	9
4.2.2	MT4API_GetErrorDescription	9
4.2.3	MT4API_GetLastErrorDescription	9
4.3	模拟账号创建接口	10
4.3.1	MT4API_CreateDemoAccount	10
4.4	连接相关接口	10
4.4.1	MT4API_Init	10
4.4.2	MT4API_Connect	11
4.4.3	MT4API_SetDisconnectEventHandler	11
4.4.4	MT4API_DisConnect	11
4.4.5	MT4API_IsConnect	12
4.5	账户信息接口	12
4.5.1	MT4API_GetUser	12
4.5.2	MT4API_GetUserName	12
4.5.3	MT4API_GetUserGroup	13
4.5.4	MT4API_IsDemoAccount	13
4.5.5	MT4API_GetMoneyInfo	13
4.5.6	MT4API_GetFreeMargin	14
4.5.7	MT4API_GetMargin	14
4.5.8	MT4API_GetBalance	14
4.5.9	MT4API_GetCredit	14
4.5.10	MT4API_GetProfit	15
4.5.11	MT4API_GetEquity	15
4.5.12	MT4API_GetLeverage	15

4.6	服务器信息接口	16
4.6.1	MT4API_GetServerTime	16
4.6.2	MT4API_GetServerBuild	16
4.6.3	MT4API_GetSeverInfoCount	16
4.6.4	MT4API_GetSeverInfos	16
4.7	证券信息接口	17
4.7.1	MT4API_GetSymbolInfo	17
4.7.2	MT4API_GetSymbolsCount	17
4.7.3	MT4API_GetSymbolsInfo	17
4.7.4	MT4API_GetSymbolGroups	18
4.8	证券订阅接口	18
4.8.1	MT4API_IsSubscribe	18
4.8.2	MT4API_Subscribe	18
4.8.3	MT4API_UnSubscribe	19
4.9	行情接口	19
4.9.1	MT4API_DownloadQuoteHistory	19
4.9.2	MT4API_SetQuoteHistoryEventHandler	19
4.9.3	MT4API_GetQuote	20
4.9.4	MT4API_SetQuoteEventHandler	20
4.10	订单信息接口	20
4.10.1	MT4API_GetCloseOrder	20
4.10.2	MT4API_GetCloseOrdersCount	21
4.10.3	MT4API_GetCloseOrders	21
4.10.4	MT4API_GetOpenOrder	21
4.10.5	MT4API_GetOpenOrdersCount	22
4.10.6	MT4API_GetOpenOrders	22
4.10.7	MT4API_GetTradeHistoryCount	22
4.10.8	MT4API_DownloadTradeHistory	23
4.11	交易接口	23
4.11.1	MT4API_IsTradeAllowed	23
4.11.2	MT4API_SetOrderNotifyEventHandler	23
4.11.3	MT4API_SetOrderUpdateEventHandler	24
4.11.4	MT4API_OrderSend	24
4.11.5	MT4API_OrderSendAsync	24
4.11.6	MT4API_OrderClose	25
4.11.7	MT4API_OrderCloseAsync	25
4.11.8	MT4API_OrderCloseBy	26

4.11.9	MT4API_OrderCloseByAsync.....	26
4.11.10	MT4API_OrderMultipleCloseBy	27
4.11.11	MT4API_OrderMultipleCloseByAsync.....	27
4.11.12	MT4API_OrderDelete	27
4.11.13	MT4API_OrderDeleteAsync	27
4.11.14	MT4API_OrderModify	28
4.11.15	MT4API_OrderModifyAsync	28
5	相关结构体和宏定义附表	29
6	常见 FAQ	29
6.1	为什么集成 dll 之后跑不起来？	29
6.2	怎么获取 license 文件.....	29
7	联系我们	29

mt4api 接口, 是用于直接连接 mt4 服务器, 获取实时行情报价, 交易执行, 以及历史行情和历史订单的查询等功能的 dll 接口, 适合开发行情展示或者订单管理, 交易策略等, 如果可以, 你甚至可以用他开发一个 mt4 客户端! 该 api 比使用 mt4 终端进行报价和交易要快得多; 提供实时开仓、平仓等多种功能

[illegible]

-----> 版本更新 (2019/06/12) <-----

本手册为 C++ 语言描述，但支持 C#、Java、Delphi、VB、易语言，均可以参考本手册的参数说明；

2 基础知识

2.1 mt4

MT4 是市场行情接收软件，MT4 是 MetaQuotes Software Corp 缩写，由迈达克软件公司发布。它包括先前系统所有的特点，并且对这些功能和组成部分进行了进一步的介绍和重组。它适用于外汇，CFD 以及期货市场。MetaTrader 4 服务器明显在使用率，工作表现，和可信度方面要优于早先的系统

2.2 常用术语

对于投资新手，刚开始接触某一行，要学习的东西必然有很多的啦~在进行外汇投资时，其实有很多专业的基本术语还是需要了解，下面是一些有关外汇常用语：

建仓：建仓也叫开仓，是指交易者新买入或新卖出一定数量的黄金、白银，原油现货合约。

持仓：建仓之后尚没有平仓的合约，叫未平仓合约或者未平仓头寸，也叫持仓。

持仓量：持仓量指的是买入（或卖出）的头寸在未了结平仓前的价值总和。

平仓：买回已卖出合约，或卖出已买入合约的行为就叫平仓。

合约总价值：买卖产品的总价值。即原油或白银单价*手数（数量）。

做多：做多就是做多头，多头对市场判断是上涨，就会立即进行原油或白银买入，所以做多就是买入原油或白银。

做空：做空就是做空头，空头是对市场趋势判断是下跌行情，立即执行原油或白银的卖出行为看。

结算：是指将买入的黄金卖出，或将卖出的黄金购回。

占用保证金（预付款）：客户卖出或卖出黄金的金额价*预付款比例。

可用预付款：客户净存入资金总额，扣除客户买卖账户发生的全部手续费、库存费、易用预付款，并加上浮动盈亏后的资金余额。

买入价：客户向深石化买入原油的价格，即报价系统显示的较高的价格。

卖出价：客户向深石化卖出原油的价格，即报价系统显示的价低的价格。

佣金：经纪人从客户交易中收取的费用，出于自身利益考虑，经纪人会鼓励客户交易。

客户报表：记录客户买卖相关信息的列表单。

手续费：交易平台为客户提供买卖交易服务所收取的服务费用。

持仓风险率： $(\text{客户账户净值} / \text{持仓占用交易保证金}) * 100\%$

客户账户净值：账户余额+浮动盈亏。

盘整：盘整是指股价在一段时间内波动幅度小，无明显的上涨或下跌趋势，股价呈牛皮整理，给阶段的行情振幅小，方向不易把握，是投资者最迷惑的时候。

现手：已经成交的最新一笔买卖的手数。

仓位：给定货币的净持有量。仓位可以是空仓、多空持平、看多仓（买入仓位多于卖出仓位）或看空仓（卖出仓位多于买入仓位）。

满仓：不留资金，全部买或卖合约就是满仓，留一半叫半仓，跌了后再买叫补仓，从无到有的新买的叫建仓。

买入价：卖方某一时刻对某种货币获得的最高价格，卖出与买入价格的差额是价差。总之，报价由这两个价格组成，只是有一定价差。外汇中买卖价差用交易成本的百分比来表示。

买卖权平价：相同到期日的认购期权和认沽期权价格的平衡关系。

结算：结算是指遵照交易所有关规定和会员、用户的交易成果，对保证金、盈亏、手续费、延期费等款项进行的划拨较量争论。实行每日无负债结算形式，结算从此浮动盈亏被转化成结算盈亏，并产生资金的实际划转。

价格幅度：一个给定的交易时段中期货最高和最低价格的差距。

杠杆交易：顾名思义，就是利用小额的资金来进行数倍于原始金额的投资，以期望获取相对投资标的物波动的数倍收益率，抑或亏损。由于保证金（该笔小额资金）的增减不以标的资产的波动比例来运动，风险很高。

最高价：是指当日所成交的价格中的最高价位。

最低价：是指当日所成交的价格中的最低价位。

利空：促使价格下跌对空头有利的因素和消息。

利多：是刺激价格上涨对多头有利的因素和消息。

成交量：反映成交的数量。单位为手按买卖双边计算总和。

价位：喊价的升降单位。价位的高低随合约的不同而不同。

涨跌：以每天的收盘价与前一天的结算价相比较反映合约价格是涨还是跌。

3 API 使用流程

- 1) 应用程序先调用 `MT4API_Create()` 创建实例，获取实例 `ClientID`;
- 2) 通过调用 `MT4API_Init()` 初始化该实例的链接等相关参数;
- 3) 调用 `MT4API_Connect()` 进行连接登陆接服务器;
- 4) 退出前调用 `MT4API_Destroy()` 销毁实例;
- 5) 使用过程中，如果出现错误，可以调用 `MT4API_GetLastErr()`, `MT4API_GetErrorDescription()`, `MT4API_GetLastErrorDescription()` 等函数获取错误信息;

注释:

参数分为传入、传出和双向（传入&传出）参数，需要注意区分;

4 接口函数

4.1 实例创建销毁接口

4.1.1 MT4API_Create

```
int WINAPI MT4API_Create();
```

功能:

创建客户端通讯实例;

参数:

无;

返回值:

通讯实例 ID，成功时返回 大于 0 的值，失败返回-1;

4.1.2 MT4API_Destroy

```
void WINAPI MT4API_Destroy(const int nID);
```

功能:

关闭通信实例

参数:

nID - 通讯实例 ID; (IN)

返回值：
空；

4.2 错误获取接口

4.2.1 MT4API_GetLastError

```
int WINAPI MT4API_GetLastError(const int nID, const int nPos);
```

功能：
获取错误号

参数：
nID - 通讯实例 ID；(IN)
nPos - 错误号位置, (0, 取最后一个错误号；-1, 倒数第二个错误号，依次类推)；(IN)

返回值：
错误号；

4.2.2 MT4API_GetErrorDescription

```
bool WINAPI MT4API_GetErrorDescription(const int nErrCode, const char *pszErrDesc, const int nSize);
```

功能：
获取指定错误号的错误描述

参数：
nErrCode - 错误号；(IN)
pszErrDesc - 错误描述，一般分配 256 个字符以上；(OUT)
nSize - pszErrDesc 分配的空间大小；(IN)

返回值：
成功：true， 失败：false；

4.2.3 MT4API_GetLastErrorDescription

```
bool WINAPI MT4API_GetLastErrorDescription(const int nID, const char *pszErrDesc, const int nSize, const int nPos);
```

功能：
获取指定位置的错误号的错误描述

参数：
nID - 通讯实例 ID；(IN)
nErrCode - 错误号；(IN)
pszErrDesc - 错误描述，一般分配 256 个字符以上；(OUT)
nSize - pszErrDesc 分配的空间大小；(IN)
nPos - 错误号位置, (0, 取最后一个错误号；-1, 倒数第二个错误号，依次类推)；(IN)

返回值：
成功：true， 失败：false；

4.3 模拟账号创建接口

4.3.1 MT4API_CreateDemoAccount

```
bool WINAPI MT4API_CreateDemoAccount(
    const char* pszHost,
    const int nPort,
    const int nLeverage,
    const double fBalance,
    const char *pszName,
    const char *pszAccountType,
    const char *pszCountry,
    const char *pszCity,
    const char *pszState,
    const char *pszZip,
    const char *pszAddress,
    const char *pszPhone,
    const char *pszEmail,
    const char *pszTerminalCompany,
    Mt4_stDemoAccount &Account);
```

功能:

创建模拟账号

参数:

pszHost	- 服务器 IP 或域名; (IN)
nPort	- 服务器端口号; (IN)
nLeverage	- 杠杆比例; (IN)
fBalance	- 初始金额; (IN)
pszName	- 账户名称; (IN)
pszAccountType	- 账户类型 (用户组) (IN);
pszCountry	- 所在国家; (IN)
pszCity	- 所在城市; (IN)
pszState	- 所在区域 (洲/省) (IN);
pszZip	- 邮政编码; (IN)
pszAddress	- 详细地址; (IN)
pszPhone	- 电话号码; (IN)
pszEmail	- 邮件地址; (IN)
pszTerminalCompany	- 公司名称; (IN)
Account	- 账户信息接收结构体; (OUT)

返回值:

成功: true, 失败: false;

注意 : 此函数如果失败, 获取错误时, 传入的实例 ID 需填 0。

4.4 连接相关接口

4.4.1 MT4API_Init

```
bool WINAPI MT4API_Init(
    const int nID,
    const char* pszBroker,
    const unsigned int nUser,
    const char *pszPassword,
```

```
const char *pszHost,
const int nPort,
const int nTradeHisTimeFrom,
const int nTradeHisTimeTo);
```

功能:

初始化登陆等相关信息;

参数:

nID	- 通讯实例 ID; (IN)
pszBroker	- 登陆的服务器名称(客户端 config 目录下的 srv 文件名); (IN)
nUser	- 登陆账号; (IN)
pszPassword	- 登陆密码; (IN)
pszHost	- 登陆服务器; (IN)
nPort	- 登陆端口; (IN)
nTradeHisTimeFrom	- 初始获取历史订单的开始时间
nTradeHisTimeTo	- 初始获取历史订单的结束时间

返回值:

成功: true, 失败: false;

注意: nTradeHisTimeFrom 和 nTradeHisTimeTo 设置为 0 则默认请求最近一个月的历史交易记录。

4.4.2 MT4API_Connect

```
bool WINAPI MT4API_Connect(const int nID);
```

功能:

连接登陆服务器;

参数:

nID	- 通讯实例 ID; (IN)
-----	-----------------

返回值:

成功: true, 失败: false;

4.4.3 MT4API_SetDisconnectEventHandler

```
bool WINAPI MT4API_SetDisconnectEventHandler(const int nID, LPVOID pHandler, LPVOID param);
```

功能:

设置连接断开回调接收处理函数;

参数:

nID	- 通讯实例 ID; (IN)
pHandler	- 回调接收函数指针; (IN)
param	- 回调函数回传参数; (IN)

返回值:

成功: true, 失败: false;

4.4.4 MT4API_DisConnect

```
bool WINAPI MT4API_DisConnect(const int nID);
```

功能:

断开连接;

参数:

nID - 通讯实例 ID; (IN)

返回值:

成功: true, 失败: false;

4.4.5 MT4API_IsConnect

```
bool WINAPI MT4API_IsConnect(const int nID);
```

功能:

是否已经连接登陆服务器;

参数:

nID - 通讯实例 ID; (IN)

返回值:

true:是, false:否;

4.5 账户信息接口

4.5.1 MT4API_GetUser

```
bool WINAPI MT4API_GetUser(const int nID, unsigned int &nUser);
```

功能:

获取当前登陆账号;

参数:

nID - 通讯实例 ID; (IN)
nUser - 账号; (OUT)

返回值:

成功: true, 失败: false;

4.5.2 MT4API_GetUserName

```
bool WINAPI MT4API_GetUserName(const int nID, const char * pUserName, const int nSize);
```

功能:

获取当前登陆账号名称;

参数:

nID - 通讯实例 ID; (IN)
pUserName - 账号名称; (OUT)
nSize - pUserName 分配空间大小(建议不低于 64); (IN)

返回值:

成功: true, 失败: false;

4.5.3 MT4API_GetUserGroup

```
bool WINAPI MT4API_GetUserGroup(const int nID, Mt4_stConGroup &group);
```

功能:

获取当前登陆账号所属用户组信息;

参数:

nID	- 通讯实例 ID; (IN)
group	- 用户组信息结构体; (OUT)

返回值:

成功: true, 失败: false;

4.5.4 MT4API_IsDemoAccount

```
bool WINAPI MT4API_IsDemoAccount(const int nID);
```

功能:

获取当前登陆账号是否密码账号;

参数:

nID	- 通讯实例 ID; (IN)
-----	-----------------

返回值:

true: 是, false: 否;

4.5.5 MT4API_GetMoneyInfo

```
bool WINAPI MT4API_GetMoneyInfo(
    const int nID,
    double &fBalance,
    double &fCredit,
    double &fMargin,
    double &fFreeMargin,
    double &fEquity,
    double &fProfit);
```

功能:

获取账号资金情况;

参数:

nID	- 通讯实例 ID; (IN)
fBalance	- 余额; (OUT)
fCredit	- 信用; (OUT)
fMargin	- 占用保证金; (OUT)
fFreeMargin	- 可用保证金; (OUT)
fEquity	- 净值; (OUT)
fProfit	- 浮动盈亏; (OUT)

返回值:

成功: true, 失败: false;

4.5.6 MT4API_GetFreeMargin

```
bool WINAPI MT4API_GetFreeMargin(const int nID, double &fFreeMargin);
```

功能:

获取可用保证金;

参数:

nID	- 通讯实例 ID; (IN)
fFreeMargin	- 可用保证金; (OUT)

返回值:

成功: true, 失败: false;

4.5.7 MT4API_GetMargin

```
bool WINAPI MT4API_GetMargin(const int nID, double &fMargin);
```

功能:

获取占用保证金;

参数:

nID	- 通讯实例 ID; (IN)
fMargin	- 占用保证金; (OUT)

返回值:

成功: true, 失败: false;

4.5.8 MT4API_GetBalance

```
bool WINAPI MT4API_GetBalance(const int nID, double &fBalance);
```

功能:

获取余额;

参数:

nID	- 通讯实例 ID; (IN)
fBalance	- 余额; (OUT)

返回值:

成功: true, 失败: false;

4.5.9 MT4API_GetCredit

```
bool WINAPI MT4API_GetCredit(const int nID, double &fCredit);
```

功能:

获取信用;

参数:

nID - 通讯实例 ID; (IN)
fCredit - 信用; (OUT)

返回值:

成功: true, 失败: false;

4.5.10 MT4API_GetProfit

```
bool WINAPI MT4API_GetCredit(const int nID, double &fProfit);
```

功能:

获取浮动盈亏;

参数:

nID - 通讯实例 ID; (IN)
fProfit - 浮动盈亏; (OUT)

返回值:

成功: true, 失败: false;

4.5.11 MT4API_GetEquity

```
bool WINAPI MT4API_GetCredit(const int nID, double &fEquity);
```

功能:

获取净值;

参数:

nID - 通讯实例 ID; (IN)
fEquity - 净值; (OUT)

返回值:

成功: true, 失败: false;

4.5.12 MT4API_GetLeverage

```
bool WINAPI MT4API_GetLeverage(const int nID, int &nLeverage);
```

功能:

获取杠杆比例;

参数:

nID - 通讯实例 ID; (IN)
nLeverage - 杠杆比例; (OUT)

返回值:

成功: true, 失败: false;

4.6 服务器信息接口

4.6.1 MT4API_GetServerTime

```
bool WINAPI MT4API_GetServerTime(const int nID, int &nTime);
```

功能:

获取服务器时间;

参数:

nID	- 通讯实例 ID; (IN)
nTime	- 服务器时间; (OUT)

返回值:

成功: true, 失败: false;

4.6.2 MT4API_GetServerBuild

```
bool WINAPI MT4API_GetServerBuild(const int nID, unsigned short &nServerBuild);
```

功能:

获取服务端版本号;

参数:

nID	- 通讯实例 ID; (IN)
nServerBuild	- 服务端版本号; (OUT)

返回值:

成功: true, 失败: false;

4.6.3 MT4API_GetSeverInfoCount

```
bool WINAPI MT4API_GetSeverInfoCount(const int nID, int &nServerInfoCount);
```

功能:

获取服务器列表信息个数;

参数:

nID	- 通讯实例 ID; (IN)
nServerInfoCount	- 服务器列表信息个数; (OUT)

返回值:

成功: true, 失败: false;

4.6.4 MT4API_GetSeverInfos

```
bool WINAPI MT4API_GetSeverInfos(const int nID, const NS_APIX::Mt4_stConDataServer *pServerInfos, int &nCount);
```


功能:

获取服务器列表信息;

参数:

nID	- 通讯实例 ID; (IN)
pServerInfos	- 服务器列表信息; (OUT)
nCount	- 服务器列表信息个数; (IN)

返回值:

成功: true, 失败: false;

4.7 证券信息接口

4.7.1 MT4API_GetSymbolInfo

```
bool WINAPI MT4API_GetSymbolInfo(const int nID, const char *pszSymbolName, Mt4_stConSymbol &symbol);
```

功能:

获取证券信息;

参数:

nID	- 通讯实例 ID; (IN)
pszSymbolName	- 证券名称; (IN)
symbol	- 证券信息存储结构体; (OUT)

返回值:

成功: true, 失败: false;

4.7.2 MT4API_GetSymbolsCount

```
bool WINAPI MT4API_GetSymbolsCount(const int nID, int &nSymbolCount);
```

功能:

获取当前服务端支持的证券个数;

参数:

nID	- 通讯实例 ID; (IN)
nSymbolCount	- 证券个数; (OUT)

返回值:

成功: true, 失败: false;

4.7.3 MT4API_GetSymbolsInfo

```
bool WINAPI MT4API_GetSymbolsInfo(const int nID, const Mt4_stConSymbol * pSymbols, int &nCount);
```

功能:

获取当前服务端支持的所有证券信息;

参数:

nID	- 通讯实例 ID; (IN)
pSymbols	- 所有证券信息; (OUT)
nCount	- 证券个数; (IN, OUT)

返回值:

成功: true, 失败: false;

4.7.4 MT4API_GetSymbolGroups

```
bool CALL_TYPE MT4API_GetSymbolGroups(const int nID, const Mt4_stConSymbolGroup *pGroups, int &nCount);
```

功能:

获取当前服证券分组信息;

参数:

nID	- 通讯实例 ID; (IN)
pGroups	- 证券分组信息; (OUT)
nCount	- 证券分组个数 (请设置为 32 个); (IN, OUT)

返回值:

成功: true, 失败: false;

注意: nCount 需要设置为 32 个, 如果大于 32 个, 也只返回 32 个, 小于 32 个则返回设置个数!

4.8 证券订阅接口

4.8.1 MT4API_IsSubscribe

```
bool WINAPI MT4API_IsSubscribe(const int nID, const char *pszSymbolName);
```

功能:

证券是否订阅;

参数:

nID	- 通讯实例 ID; (IN)
pszSymbolName	- 证券名称; (IN)

返回值:

true: 已订阅, false: 未订阅;

4.8.2 MT4API_Subscribe

```
bool WINAPI MT4API_Subscribe(const int nID, const char *pszSymbolName);
```

功能:

订阅证券报价;

参数:

nID	- 通讯实例 ID; (IN)
pszSymbolName	- 证券名称; (IN)

返回值:

成功: true, 失败: false;

4.8.3 MT4API_UnSubscribe

```
bool WINAPI MT4API_UnSubscribe(const int nID, const char *pszSymbolName);
```

功能:

取消券报价订阅;

参数:

nID	- 通讯实例 ID; (IN)
pszSymbolName	- 证券名称; (IN)

返回值:

成功: true, 失败: false;

4.9 行情接口

4.9.1 MT4API_DownloadQuoteHistory

```
bool WINAPI MT4API_DownloadQuoteHistory(
    const int nID,
    const char *pszSymbolName,
    const int nPeroid,
    const int nTimeFrom,
    const short nCount);
```

功能:

获取历史行情报价;

参数:

nID	- 通讯实例 ID; (IN)
pszSymbolName	- 证券名称; (IN)
nPeroid	- 周期; (参见周期宏定义) (IN)
nTimeFrom	- 历史报价起始时间; (IN)
nCount	- 获取记录数(从最后一条开始计数); (IN)

返回值:

成功: true, 失败: false;

注意: 如果从 nTimeFrom 时间起, 记录数小于 nCount, 只获取实际记录数, 如果记录数大于 nCount, 则获取最新的 nCount 条记录。结果通过 MT4API_SetQuoteHistoryEventHandler 设置的函数回调。

4.9.2 MT4API_SetQuoteHistoryEventHandler

```
bool WINAPI MT4API_SetQuoteHistoryEventHandler(const int nID, LPVOID pHandler, LPVOID param);
```

功能:

设置历史行情报价数据回调函数指针;

参数:

nID	- 通讯实例 ID; (IN)
pHandler	- 史行情报价数据回调函数指针; (IN)
param	- 回传参数 (IN)

返回值:

成功: true, 失败: false;

4.9.3 MT4API_GetQuote

```
bool WINAPI MT4API_GetQuote(const int nID, const char *pszSymbolName, Mt4_stQuoteEventInfo &Info);
```

功能:

获取历史行情报价;

参数:

nID	- 通讯实例 ID; (IN)
pszSymbolName	- 证券名称; (IN)
Info	- 最新报价信息; (IN)

返回值:

成功: true, 失败: false;

4.9.4 MT4API_SetQuoteEventHandler

```
bool WINAPI MT4API_SetQuoteEventHandler(const int nID, LPVOID pHandler, LPVOID param);
```

功能:

设置证券行情报价数据回调函数指针;

参数:

nID	- 通讯实例 ID; (IN)
pHandler	- 最新情报价数据回调函数指针; (IN)
param	- 回传参数 (IN)

返回值:

成功: true, 失败: false;

4.10 订单信息接口

4.10.1 MT4API_GetCloseOrder

```
bool WINAPI MT4API_GetCloseOrder(const int nID, const int nOrder, Mt4_stTradeRecord &tr);
```

功能:

通过订单号获取 Init 时传入的时间范围内的交易订单信息;

参数:

nID	- 通讯实例 ID; (IN)
-----	-----------------

nOrder	- 订单 ID; (IN)
tr	- 订单信息; (OUT)

返回值:

成功: true, 失败: false;

注意: 此接口只能获取 MT4API_Init 时传入的时间段的历史订单信息!

4.10.2 MT4API_GetCloseOrdersCount

```
bool WINAPI MT4API_GetCloseOrdersCount(const int nID, int &nOrderCount);
```

功能:

获取 Init 时传入的时间范围内的交易订单记录个数;

参数:

nID	- 通讯实例 ID; (IN)
nOrderCount	- 订单个数; (OUT)

返回值:

成功: true, 失败: false;

注意: 此接口只能获取 MT4API_Init 时传入的时间段的历史订单个数!

4.10.3 MT4API_GetCloseOrders

```
bool WINAPI MT4API_GetCloseOrders(const int nID, const Mt4_stTradeRecord *pOrders, int &nCount);
```

功能:

获取 Init 时传入的时间范围内的交易订单记录;

参数:

nID	- 通讯实例 ID; (IN)
pOrders	- 订单信息; (OUT)
nCount	- 订单个数; (IN, OUT)

返回值:

成功: true, 失败: false;

注意: 此接口只能获取 MT4API_Init 时传入的时间段的历史订单!

4.10.4 MT4API_GetOpenOrder

```
bool WINAPI MT4API_GetOpenOrder(const int nID, const int nOrder, Mt4_stTradeRecord &tr);
```

功能:

通过订单号获取交易中的订单信息;

参数:

nID	- 通讯实例 ID; (IN)
nOrder	- 订单 ID; (IN)
tr	- 订单信息; (OUT)

返回值:

成功: true, 失败: false;

4.10.5 MT4API_GetOpenOrdersCount

```
bool WINAPI MT4API_GetOpenOrdersCount(const int nID, int &nOrderCount);
```

功能:

获取交易中的订单个数;

参数:

nID	- 通讯实例 ID; (IN)
nOrderCount	- 订单个数; (OUT)

返回值:

成功: true, 失败: false;

4.10.6 MT4API_GetOpenOrders

```
bool WINAPI MT4API_GetOpenOrders(const int nID, const Mt4_stTradeRecord *pOrders, int &nCount);
```

功能:

获取交易中的订单信息;

参数:

nID	- 通讯实例 ID; (IN)
pOrders	- 订单信息; (OUT)
nCount	- 订单个数; (IN, OUT)

返回值:

成功: true, 失败: false;

4.10.7 MT4API_GetTradeHistoryCount

```
bool WINAPI MT4API_GetTradeHistoryCount(const int nID, const int nTimeFrom, const int nTimeTo, int &nOrderCount);
```

功能:

获取历史交易订单个数;

参数:

nID	- 通讯实例 ID; (IN)
nTimeFrom	- 起始时间; (IN)
nTimeTo	- 结束时间; (IN)
nOrderCount	- 订单记录个数; (OUT)

返回值:

成功: true, 失败: false;

4.10.8 MT4API_DownloadTradeHistory

```
bool WINAPI MT4API_DownloadTradeHistory(const int nID, const Mt4_stTradeRecord *pOrders, int &nCount);
```

功能:

获取历史交易订单信息;

参数:

nID	- 通讯实例 ID; (IN)
pOrders	- 所有订单信息; (OUT)
nCount	- 订单个数; (IN, OUT)

返回值:

成功: true, 失败: false;

用法: 先调用 MT4API_GetTradeHistoryCount 获取个数, 在调用此接口获取具体信息。

4.11 交易接口

4.11.1 MT4API_IsTradeAllowed

```
bool WINAPI MT4API_IsTradeAllowed(const int nID);
```

功能:

当前是否可进行交易操作;

参数:

nID	- 通讯实例 ID; (IN)
-----	-----------------

返回值:

True: 可以, false: 不可以;

4.11.2 MT4API_SetOrderNotifyEventHandler

```
bool WINAPI MT4API_SetOrderNotifyEventHandler (const int nID, LPVOID pHandler, LPVOID param);
```

功能:

设置订单处理结果回调函数;

参数:

nID	- 通讯实例 ID; (IN)
pHandler	- 订单结果回调函数指针; (IN)
param	- 回传参数 (IN)

返回值:

成功: true, 失败: false;

4.11.3 MT4API_SetOrderUpdateEventHandler

```
bool WINAPI MT4API_SetOrderUpdateEventHandler (const int nID, LPVOID pHandler, LPVOID param);
```

功能:

设置订单处理过程回调函数;

参数:

nID	- 通讯实例 ID; (IN)
pHandler	- 订单处理过程回调函数指针; (IN)
param	- 回传参数 (IN)

返回值:

成功: true, 失败: false;

4.11.4 MT4API_OrderSend

```
bool WINAPI MT4API_OrderSend(
    const int nID,
    const char *pszSymbol,
    unsigned char nTradeCmd,
    const double volume,
    const double price,
    const int ie_deviation,
    const double stoploss,
    const double takeprofit,
    const char *pszComment,
    const int magic,
    const int expiration,
    Mt4_stTradeRecord &TradeRecord);
```

功能:

下单(开单, 开仓);

参数:

nID	- 通讯实例 ID; (IN)
pszSymbol	- 证券名称; (IN)
nTradeCmd	- 交易指令; (OP_BUY, OP_SELL... 参见宏定义) (IN)
volume	- 量(手); (IN)
price	- 下单价格; (IN)
ie_deviation	- 点差; (IN)
stoploss	- 止损价; (IN)
takeprofit	- 止盈价; (IN)
pszComment	- 说明; (IN)
magic	- 魔法号(当前选定订单的标识号); (IN)
expiration	- 过期时间; (IN)
TradeRecord	- 下单结果; (OUT)

返回值:

成功: true, 失败: false;

4.11.5 MT4API_OrderSendAsync

```
int WINAPI MT4API_OrderSendAsync(
    const int nID,
    const char *pszSymbol,
    unsigned char nTradeCmd,
    const double volume,
    const double price,
    const int ie_deviation,
```



```

    const double stoploss,
    const double takeprofit,
    const char *pszComment,
    const int magic,
    const int expiration,
    Mt4_stTradeRecord &TradeRecord);

```

功能:

异步下单（开仓，开单）；

参数:

nID	- 通讯实例 ID; (IN)
pszSymbol	- 证券名称; (IN)
nTradeCmd	- 交易指令; (OP_BUY, OP_SELL... 参见宏定义) (IN)
volume	- 量 (手); (IN)
price	- 下单价格; (IN)
ie_deviation	- 点差; (IN)
stoploss	- 止损价; (IN)
takeprofit	- 止盈价; (IN)
pszComment	- 说明; (IN)
magic	- 魔法号 (当前选定订单的标识号); (IN)
expiration	- 过期时间; (IN)

返回值:

返回当前交易请求号，大于 0，则成功，等于 0，失败；

4.11.6 MT4API_OrderClose

```

bool WINAPI MT4API_OrderClose(
    const int nID,
    const char *pszSymbol,
    const int order,
    const double volume,
    const double price,
    const int ie_deviation,
    Mt4_stTradeRecord &TradeRecord);

```

功能:

关闭订单（平仓）；

参数:

nID	- 通讯实例 ID; (IN)
pszSymbol	- 证券名称; (IN)
order	- 订单号; (IN)
volume	- 量 (手); (IN)
price	- 下单价格; (IN)
ie_deviation	- 点差; (IN)
TradeRecord	- 下单结果; (OUT)

返回值:

成功: true, 失败: false;

4.11.7 MT4API_OrderCloseAsync

```

int WINAPI MT4API_OrderCloseAsync(
    const int nID,
    const char *pszSymbol,
    const int order,
    const double volume,
    const double price,
    const int ie_deviation);

```

功能:

异步关闭订单（平仓）；

参数:

nID	- 通讯实例 ID; (IN)
pszSymbol	- 证券名称; (IN)
order	- 订单号; (IN)
volume	- 量（手）; (IN)
price	- 下单价格; (IN)
ie_deviation	- 点差; (IN)

返回值:

返回当前交易请求号，大于 0，则成功，等于 0，失败；

4.11.8 MT4API_OrderCloseBy

```
bool WINAPI MT4API_OrderCloseBy(
    const int nID,
    const int order1,
    const int order2);
```

功能:

同一价格关闭锁仓的两个订单（同时平掉一个多单和一个空单）；

参数:

nID	- 通讯实例 ID; (IN)
order1	- 订单号 1; (IN)
order2	- 反向订单号 2; (IN)

返回值:

成功: true, 失败: false;

注意: 此函数需要服务器设置支持才能调用成功。

4.11.9 MT4API_OrderCloseByAsync

```
bool WINAPI MT4API_OrderCloseByAsync(
    const int nID,
    const int order1,
    const int order2);
```

功能:

异步同一价格关闭锁仓的两个订单（同时平掉一个多单和一个空单）；

参数:

nID	- 通讯实例 ID; (IN)
pszSymbol	- 证券名称; (IN)
order1	- 订单号 1; (IN)
order2	- 反向订单号 2; (IN)

返回值:

返回当前交易请求号，大于 0，则成功，等于 0，失败；

注意: 此函数需要服务器设置支持才能调用成功。

4.11.10 MT4API_OrderMultipleCloseBy

```
bool WINAPI MT4API_OrderMultipleCloseBy(const int nID, const char *pszSymbol);
```

功能:

关闭指定证券的所有订单（对指定证券的所有订单进行平仓）；

参数:

nID	- 通讯实例 ID; (IN)
pszSymbol	- 证券名称; (IN)

返回值:

成功: true, 失败: false;

注意: 此函数需要服务器设置支持才能调用成功。

4.11.11 MT4API_OrderMultipleCloseByAsync

```
bool WINAPI MT4API_OrderMultipleCloseByAsync(const int nID, const char *pszSymbol);
```

功能:

异步关闭指定证券的所有订单（对指定证券的所有订单进行平仓）；

参数:

nID	- 通讯实例 ID; (IN)
pszSymbol	- 证券名称; (IN)

返回值:

返回当前交易请求号, 大于 0, 则成功, 等于 0, 失败;

注意: 此函数需要服务器设置支持才能调用成功。

4.11.12 MT4API_OrderDelete

```
bool WINAPI MT4API_OrderDelete(const int nID, const int order);
```

功能:

删除挂单;

参数:

nID	- 通讯实例 ID; (IN)
order	- 挂单号; (IN)

返回值:

成功: true, 失败: false;

4.11.13 MT4API_OrderDeleteAsync

```
bool WINAPI MT4API_OrderDeleteAsync(const int nID, const int order);
```

功能:

异步删除挂单;

参数:

nID	- 通讯实例 ID; (IN)
-----	-----------------

order - 挂单号; (IN)

返回值:

返回当前交易请求号, 大于 0, 则成功, 等于 0, 失败;

4.11.14 MT4API_OrderModify

```
bool WINAPI MT4API_OrderModify(
    const int nID,
    unsigned char nTradeCmd,
    const int order,
    const double price,
    const double stoploss,
    const double takeprofit,
    const int expiration,
    Mt4_stTradeRecord &TradeRecord);
```

功能:

修改挂单;

参数:

nID	- 通讯实例 ID; (IN)
pszSymbol	- 证券名称; (IN)
nTradeCmd	- 交易指令; (OP_BUY, OP_SELL... 参见宏定义) (IN)
order	- 订单号; (IN)
price	- 下单价格; (IN)
ie_deviation	- 点差; (IN)
stoploss	- 止损价; (IN)
takeprofit	- 止盈价; (IN)
expiration	- 过期时间; (IN)
TradeRecord	- 下单结果; (OUT)

返回值:

成功: true, 失败: false;

4.11.15 MT4API_OrderModifyAsync

```
bool WINAPI MT4API_OrderModifyAsync(
    const int nID,
    unsigned char nTradeCmd,
    const int order,
    const double price,
    const double stoploss,
    const double takeprofit,
    const int expiration,
    Mt4_stTradeRecord &TradeRecord);
```

功能:

异步修改挂单;

参数:

nID	- 通讯实例 ID; (IN)
pszSymbol	- 证券名称; (IN)
nTradeCmd	- 交易指令; (OP_BUY, OP_SELL... 参见宏定义) (IN)
order	- 订单号; (IN)
price	- 下单价格; (IN)
ie_deviation	- 点差; (IN)
stoploss	- 止损价; (IN)
takeprofit	- 止盈价; (IN)
expiration	- 过期时间; (IN)

返回值:

返回当前交易请求号，大于 0，则成功，等于 0，失败；

5 相关结构体和宏定义附表

详情请参考 mt4api.h 头文件。

6 常见 FAQ

6.1 为什么集成 dll 之后跑不起来？

请确保已经有 license 文件，且 mt4api.dll, LoginX.dll, LoginId.dll, mt4api.lic, mt4api.ini 在你的程序目录下。

6.2 怎么获取 license 文件

可以通过开发包中带的 License 生成器生成或直接联系我们！

7 联系我们

外汇 MT4 技术交流群： 712830563

Email: support@gzhuichou.com

官方首页: <https://mt4api.gzhuichou.com>