

TradeX Dev Team

# [TradeX 交易接口 开发手册 C++]

[多账户版]

[版本 1.4.0]

[2017-10-12]

# 目 录

1. 简述.....	4
2. 基础知识.....	5
2.1. 行情.....	5
2.1.1. 行情服务器.....	5
2.1.2. Level 2 行情.....	5
3. 交易 API（批量版/多账户版）.....	6
3.1. 函数.....	6
3.1.1. OpenTdx 打开通达信实例.....	6
3.1.2. CloseTdx 关闭通达信实例.....	7
3.1.3. Logon 交易账户登录.....	7
3.1.4. Logoff 交易账户退出登录.....	8
3.1.5. IsConnectOK 判断交易连接是否正确.....	8
3.1.6. QueryData 查询交易数据.....	9
3.1.7. SendOrder 委托下单.....	10
3.1.8. CancelOrder 撤销委托.....	11
3.1.9. GetQuote 获取证券的实时五档行情.....	11
3.1.10. GetTradableQuantity 查询当前可交易股票数量.....	12
3.1.11. Repay 融资融券账户直接还款.....	13
3.1.12. QueryHistoryData 查询历史交易数据.....	14
3.1.13. QueryDatas 单账户批量查询各类交易数据.....	14
3.1.14. SendOrders 单账户批量委托交易证券.....	15
3.1.15. CancelOrders 批量撤委托.....	16
3.1.16. GetQuotes.....	17
3.1.17. QueryMultiAccountsDatas.....	18
3.1.18. SendMultiAccountsOrders.....	19
3.1.19. CancelMultiAccountsOrders.....	20
3.1.20. GetMultiAccountsQuotes.....	20
4. 普通行情 API.....	21
4.1. 使用流程.....	21
4.2. 函数.....	22
4.2.1. TdxHq_Connect.....	22
4.2.2. TdxHq_Disconnect.....	22
4.2.3. TdxHq_GetSecurityCount.....	23
4.2.4. TdxHq_GetSecurityList.....	23
4.2.5. TdxHq_GetSecurityQuotes.....	24
4.2.6. TdxHq_GetSecurityBars.....	25
4.2.7. TdxHq_GetIndexBars.....	26
4.2.8. TdxHq_GetMinuteTimeData.....	27
4.2.9. TdxHq_GetHistoryMinuteTimeData.....	27
4.2.10. TdxHq_GetTransactionData.....	28
4.2.11. TdxHq_GetHistoryTransactionData.....	29

4.2.12. TdxHq_GetCompanyInfoCategory.....	30
4.2.13. TdxHq_GetCompanyInfoContent.....	30
4.2.14. TdxHq_GetXDXRInfo.....	31
4.2.15. TdxHq_GetFinanceInfo.....	32
<b>5. Level 2 行情 API.....</b>	<b>33</b>
5.1. 使用流程.....	33
5.2. 函数.....	33
5.2.1. TdxL2Hq_Connect.....	33
5.2.2. TdxL2Hq_Disconnect.....	34
5.2.3. TdxL2Hq_GetSecurityCount.....	34
5.2.4. TdxL2Hq_GetSecurityList.....	35
5.2.5. TdxL2Hq_GetSecurityQuotes.....	35
5.2.6. TdxL2Hq_GetSecurityBars.....	36
5.2.7. TdxL2Hq_GetIndexBars.....	37
5.2.8. TdxL2Hq_GetMinuteTimeData.....	38
5.2.9. TdxL2Hq_GetHistoryMinuteTimeData.....	39
5.2.10. TdxL2Hq_GetTransactionData.....	40
5.2.11. TdxL2Hq_GetHistoryTransactionData.....	40
5.2.12. TdxL2Hq_GetCompanyInfoCategory.....	41
5.2.13. TdxL2Hq_GetCompanyInfoContent.....	42
5.2.14. TdxL2Hq_GetXDXRInfo.....	43
5.2.15. TdxL2Hq_GetFinanceInfo.....	43
5.2.16. TdxL2Hq_GetSecurityQuotes10.....	44
5.2.17. TdxL2Hq_GetDetailTransactionData.....	45
5.2.18. TdxL2Hq_GetDetailOrderData.....	45
5.2.19. TdxL2Hq_GetBuySellQueue.....	46
<b>6. 扩展行情 API.....</b>	<b>47</b>
6.1. 函数.....	47
6.1.1. TdxExHq_Connect.....	47
6.1.2. TdxExHq_Disconnect.....	48
6.1.3. TdxExHq_GetMarkets.....	48
6.1.4. TdxExHq_GetInstrumentCount.....	48
6.1.5. TdxExHq_GetInstrumentInfo.....	49
6.1.6. TdxExHq_GetInstrumentQuote.....	50
6.1.7. TdxExHq_GetInstrumentBars.....	50
6.1.8. TdxExHq_GetMinuteTimeData.....	51
6.1.9. TdxExHq_GetHistoryMinuteTimeData.....	52
6.1.10. TdxExHq_GetTransactionData.....	52
6.1.11. TdxExHq_GetHistoryTransactionData.....	53
<b>7. 常见问题.....</b>	<b>54</b>
7.1.1. 常见问题.....	54



## 2. 基础知识

### 2.1. 行情

TradeX.dll 的行情接口均要求客户端主动查询请求行情数据，而非全推送行情；交易所推送的分笔数据为 3s 切片，逐笔为每 3s 间隔内的所有成交。

当查询股票时，如果存在停牌、退市等无效的股票代码，行情服务器会使用代码 600839 自动填充。这种情况建议用户自行维护一个停牌/退市的股票列表，在批量获取行情时对列表中股票代码予以剔除。

API 接口的空闲超时缺省值为 2000ms，如果连接空闲超时，API 接口会主动释放到行情服务器的连接，并返回提示：“接收数据超时，连接断开，请重连服务器”。

#### 2.1.1. 行情服务器

行情服务器一侧为保证系统资源的有效利用，会主动释放超时的连接；如果行情客户端要保持连接，应避免设置过长的空闲时间，可以每隔几秒请求一次行情数据，以保持保持心跳；否则超时无活动，服务器会主动断开连接。用户可以通过编写程序，每隔几秒调用一次任意一个 API 行情函数请求行情数据，以保持连接。

服务器侧同样有空闲超时机制，当查询连接超过阈值，服务器将主动释放连接，返回错误提示：“接收数据失败，服务器主动断开连接！请检查请求数据并重连服务器”。

行情服务器对客户端频繁的连接，也会主动释放断开。

当行情服务器接收到客户端传送的错误参数（比如收到已停牌股票的代码、或不存在的股票代码），也会主动释放连接，返回错误提示：“接收数据失败，服务器主动断开连接！请检查请求数据并重连服务器”。

#### 2.1.2. Level 2 行情

接口的 Level 2 行情服务需要用户自己购买通达信的 Level 2 金融终端账户。请到 <http://vip.tdx.com.cn> 购买专有 Level 2 行情账户后，开通 TradeX 授权。也可以通过客户号/资金账户号开通授权，待以后购买 Level 2 行情账户再另行绑定。

Level 2 服务器有自我保护机制（防止类 DDOS 服务攻击），针对过度频繁的访问可能视为异常连接予以并做短暂屏蔽后才重新开始接受访问。会返回错误提示：“行情链接失败！不能链接到服务器！无法验证用户”。遇到这种情况，用户应检查自己的程序是否过度频繁的重连，或者连续重连之间未设置空闲间隔或间隔太小；暂停一段，服务器恢复服务后，即可正常连接。

Level 2 行情服务器对连接的管理是后来者优先的逻辑；当用户 A 使用某行情账户建立连接，如果其他用户 B 使用相同行情账户登录，服务器检测到新的 Level 2 行情连接请求，会释放掉先前的连接并为用户 B 建立新的连接。

所以不支持使用同一 Level 2 行情账户同时从多点连接，如果的确需要，您需要购买多个 Level 2 账户。

普通的五档实时行情、扩展行情没有任何 IP 限制，可以多点登录连接。

## 3. 交易 API（批量版/多账户版）

批量版/多账户版的函数相对于标准版有所优化，返回值有调整，多账户版有 20 个函数：

注： 有些券商对批量操作进行了限速，最大批量操作数目请咨询券商。

注释：

- 交易接口执行后，如果失败，则字符串 ErrInfo 保存了出错信息中文说明；
- 如果成功，则字符串 Result 保存了结果数据，形式为表格数据，行数据之间通过\n 字符分割，列数据之间通过\t 分隔。
- Result 是\n, \t 分隔的中文字符串，比如查询股票代码时返回的结果字符串就是

```
"股票代码\t 股东名称\t 帐号类别\t 保留信息\n0000064567\t\t0\t\nA000064567\t\t1\t\n2000064567\t\t2\t\nB000064567\t\t3\t"
```

查得此数据之后，通过分割字符串， 可以恢复为几行几列的表格形式的数据

### 3.1. 函数

#### 3.1.1. OpenTdx 打开通达信实例

```
int WINAPI OpenTdx(  
    short nClientType,  
    const char *pszClientVersion,  
    char nCliType,  
    char nVipTermFlag,  
    char *pszErrInfo);
```

功能：

打开通达信实例；

使用固定参数：`OpenTdx(14, "6.40", 12, 0)`

参数：

- |                               |   |
|-------------------------------|---|
| <code>nClientType</code>      | - 券商客户端类型；使用固定参数为 14                          |
| <code>pszClientVersion</code> | - 券商客户端的版本号；固定参数为 “6.40”                      |
| <code>nCliType</code>         | - 券商客户端指令接口类型；固定参数为 12                        |
| <code>nVipTermFlag</code>     | - 终端标志代码；固定参数为 0                              |
| <code>pszErrInfo</code>       | - 此 API 执行返回后，如果出错，保存了错误信息说明。一般要分配 256 字节的空间。 |

没出错时为空字符串。

返回值：

客户端 ID

负值 失败时返回-1

---

### 3.1.2. CloseTdx 关闭通达信实例

```
void WINAPI CloseTdx();
```

功能：

关闭通达信实例

参数： 无

---

### 3.1.3. Logon 交易账户登录

```
int WINAPI Logon(  
    int nQsid,  
    const char *pszHost,  
    short nPort,  
    const char *pszVersion,  
    short nYybId,  
    short nAccountType,  
    const char *pszAccountNo,  
    const char *pszTradeAccount,  
    const char *pszJyPassword,  
    const char *pszTxPassword,  
    char *pszErrInfo);
```

功能：

## 交易账户登录

参数:

nQsid	- 券商标识
pszHost	- 券商交易服务器 IP
nPort	- 券商交易服务器端口
pszVersion	- 设置通达信客户端的版本号
nYybId	- 营业部代码, 请到“TradeX.DLL 用户参考手册”。
nAccountType	- 登录账号类型
pszAccountNo	- 客户账户号 (Client Account), 即用户在券商客户端登陆时需要输入的账户号, 可以是客户号、资金账户号、股东代码等, 因登录类型不同而异。
pszTradeAccount	- 登录券商通达信软件, 查询股东列表, 从股东列表内的资金帐号一列获取;
pszJyPassword	- 交易密码
pszTxPassword	- 通讯密码
pszErrInfo	- 此 API 执行返回后, 如果出错, 保存了错误信息说明。一般要分配 256 字节的空间。没出错时为空字符串。

返回值:

成功则返回客户端 ID, 失败则返回-1

---

### 3.1.4. Logoff 交易账户退出登录

```
void WINAPI Logoff(int nClientID);
```

功能:

交易账户退出登录

参数:

nClientID - 客户端 ID

---

### 3.1.5. IsConnectOK 判断交易连接是否正确

```
bool WINAPI IsConnectOK(int nClientID);
```

功能:

判断交易连接是否正确

参数:

nClientID - 客户端 ID



返回值:

- 1 - 成功
- 0 - 失败

### 3.1.6. QueryData 查询交易数据

```
int WINAPI QueryData(  
    int nClientID,  
    int nCategory,  
    char *pszResult,  
    char *pszErrInfo);
```

功能:

查询各种交易数据

参数:

nClientID	- 客户端 ID
nCategory	- 查询信息的种类, 0 资金 1 股份 2 当日委托 3 当日成交 4 可撤单 5 股东代码 6 融资余额 7 融券余额 8 可融证券 9 10 11 12 可申购新股查询 13 新股申购额度查询 14 配号查询 15 中签查询
pszResult	- 此 API 执行返回后, Result 内保存了返回的查询数据, 形式为表格数据, 行数据之间通过\n 字符分割, 列数据之间通过\t 分隔。一般要分配 1024*1024 字节的空 间。  出错时为空字符串。
pszErrInfo	- 此 API 执行返回后, 如果出错, 保存了错误信息说明。一般要分配 256 字节的空间。没 出错时为空字符串。

返回值:

### 3.1.7. SendOrder 委托下单

```
int WINAPI SendOrder(  
    int nClientID,  
    int nCategory,  
    int PriceType,  
    const char *pszGddm,  
    const char *pszZqdm ,  
    float fPrice,  
    int nQuantity,  
    char *pszResult,  
    char *pszErrInfo);
```

功能:

委托下单

参数:

nClientID	- 客户端 ID
nCategory	- 委托的种类
	0 买入
	1 卖出
	2 融资买入
	3 融券卖出
	4 买券还券
	5 卖券还款
	6 现券还券
	7 担保品买入
	8 担保品卖出
nPriceType	- 报价方式
	0 限价委托: 上海限价委托 / 深圳限价委托
	1 市价委托(深圳对方最优价格)
	2 市价委托(深圳本方最优价格)
	3 市价委托(深圳即时成交剩余撤销)
	4 市价委托(上海五档即成剩撤 / 深圳五档即成剩撤)
	5 市价委托(深圳全额成交或撤销)
	6 市价委托(上海五档即成转限价)
pszGddm	- 股东代码; 交易上海股票填上海的股东代码, 交易深圳的股票填入深圳的股东代码
pszZqdm	- 证券代码
fPrice	- 委托价格
nQuantity	- 委托数量
pszResult	- 此 API 执行返回后, Result 内保存了返回的查询数据, 含有委托编号数据; 形式为表格

数据,

行数据之间通过\n 字符分割，列数据之间通过\t 分隔。一般要分配 1024\*1024 字节的空  
间。

出错时为空字符串。

pszErrInfo - 此 API 执行返回后，如果出错，保存了错误信息说明。一般要分配 256 字节的空间。没  
出错时为空字符串。

返回值：

---

### 3.1.8. CancelOrder 撤销委托

```
int WINAPI CancelOrder(  
    int nClientID,  
    char nMarket,  
    const char *pszOrderID,  
    char *pszResult,  
    char *pszErrInfo);
```

功能：

撤销委托

参数：

nClientID - 客户端 ID  
nMarket - 交易所 ID  
0 深圳 （招商证券，普通账户深圳是 2）  
1 上海

pszOrderID - 表示要撤的目标委托的编号

pszResult - 此 API 执行返回后，Result 内保存了返回的查询数据，形式为表格数据，  
行数据之间通过\n 字符分割，列数据之间通过\t 分隔。一般要分配 1024\*1024 字节的空

间。

出错时为空字符串。

pszErrInfo - 此 API 执行返回后，如果出错，保存了错误信息说明。一般要分配 256 字节的空间。没  
出错时为空字符串。

返回值：

---

### 3.1.9. GetQuote 获取证券的实时五档行情

```
int WINAPI GetQuote(  
    int nClientID,  
    const char *pszZqdm,
```

```
char *pszResult,  
char *pszErrInfo);
```

功能:

获取证券的实时五档行情

参数:

nClientID	- 客户端 ID
pszZqdm	- 证券代码
pszResult	- 此 API 执行返回后, Result 内保存了返回的查询数据, 形式为表格数据, 行数据之间通过\n 字符分割, 列数据之间通过\t 分隔。一般要分配 1024*1024 字节的空

间。

出错时为空字符串。

pszErrInfo	- 此 API 执行返回后, 如果出错, 保存了错误信息说明。一般要分配 256 字节的空间。没
------------	--

出错时为空字符串。

返回值:

---

### 3.1.10. GetTradableQuantity 查询当前可交易股票数量

```
int WINAPI GetTradableQuantity(  
    int nClientID,  
    char nCategory,  
    int nPriceType,  
    const char *pszGddm,  
    const char *pszZqdm,  
    float fPrice,  
    char *pszResult,  
    char *pszErrInfo);
```

功能:

查询可卖出股票数量 (T+1 交易, 当前可卖出的股票数量)

nClientID	- 客户端 ID
nCategory	- 委托的种类
	0 买入
	1 卖出
	2 融资买入
	3 融券卖出
	4 买券还券
	5 卖券还款
	6 现券还券
nPriceType	- 报价方式

0 限价委托； 上海限价委托 / 深圳限价委托  
 1 市价委托(深圳对方最优价格)  
 2 市价委托(深圳本方最优价格)  
 3 市价委托(深圳即时成交剩余撤销)  
 4 市价委托(上海五档即成剩撤 / 深圳五档即成剩撤)  
 5 市价委托(深圳全额成交或撤销)  
 6 市价委托(上海五档即成转限价)

pszGddm - 股东代码；交易上海股票填上海的股东代码，交易深圳的股票填入深圳的股东代码  
 pszZqdm - 证券代码  
 fPrice - 委托价格  
 pszResult - 此 API 执行返回后，Result 内保存了返回的查询数据，含有委托编号数据；形式为表格数据，  
 行数据之间通过\n 字符分割，列数据之间通过\t 分隔。一般要分配 1024\*1024 字节的空间。

出错时为空字符串。

pszErrInfo - 此 API 执行返回后，如果出错，保存了错误信息说明。一般要分配 256 字节的空间。没出错时为空字符串。

返回值：

成功时返回当前可卖股票数量，失败时返回负数

### 3.1.11. Repay 融资融券账户直接还款

```
int WINAPI Repay(
    int nClientID,
    const char *pszAmount,
    char *pszResult,
    char *pszErrInfo);
```

功能：

融资融券账户直接还款

参数：

nClientID - 客户端 ID  
 pszAmount - 还款金额  
 pszResult - 此 API 执行返回后，Result 内保存了返回的查询数据，形式为表格数据，  
 行数据之间通过\n 字符分割，列数据之间通过\t 分隔。一般要分配 1024\*1024 字节的空间。

出错时为空字符串。

pszErrInfo - 此 API 执行返回后，如果出错，保存了错误信息说明。一般要分配 256 字节的空间。没出错时为空字符串。

返回值：

---

### 3.1.12. QueryHistoryData 查询历史交易数据

```
int WINAPI QueryHistoryData(  
    int nClientID,  
    int nCategory,  
    const char *pszBeginDate,  
    const char *pszEndDate,  
    char *pszResult,  
    char *pszErrInfo);
```

功能:

查询历史数据

参数:

nClientID	- 客户端 ID
nCategory	- 查询信息的种类 0 历史委托 1 历史成交 2 资金流水 3 交割单
pszBeginDate	- 开始日期, 格式为 yyyyMMdd, 比如 2017 年 2 月 1 日为 20170201
pszEndDate	- 结束日期, 格式为 yyyyMMdd, 比如 2017 年 2 月 1 日为 20170201
pszResult	- 此 API 执行返回后, Result 内保存了返回的查询数据, 形式为表格数据, 行数据之间通过\n 字符分割, 列数据之间通过\t 分隔。一般要分配 1024*1024 字节的空

间。

出错时为空字符串。

pszErrInfo	- 此 API 执行返回后, 如果出错, 保存了错误信息说明。一般要分配 256 字节的空间。没
------------	--

出错时为空字符串。

返回值:

---

### 3.1.13. QueryDatas 单账户批量查询各类交易数据

```
int WINAPI QueryDatas(  
    int nClientID,  
    int nCategory[],  
    int nCount,  
    char *pszResultOK[],  
    char *pszResultFail[],
```

```
char *pszErrInfo[]);
```

功能:

单账户批量查询各类交易数据; 用数组传入每个委托的参数, 数组第 i 个元素表示第 i 个查询的相应参数

参数:

nClientID	- 客户端 ID
nCategory	- 信息种类的数组, 第 i 个元素表示第 i 个查询的信息种类
	0 资金
	1 股份
	2 当日委托
	3 当日成交
	4 可撤单
	5 股东代码
	6 融资余额
	7 融券余额
	8 可融证券
	9
	10
	11
	12 可申购新股查询
	13 新股申购额度查询
	14 配号查询
	15 中签查询
nCount	- 查询的个数, 即数组的长度
pszResultOK[i]	- 返回数据的数组, 第 i 个元素表示第 i 个委托的返回信息。
pszResultFail[i]	- 返回错误信息的数组, 第 i 个元素表示第 i 个委托的返回信息。
pszErrInfo	- 错误信息的数组, 返回全局性的错误信息。

返回值:

---

### 3.1.14. SendOrders 单账户批量委托交易证券

```
int WINAPI SendOrders(  
    int nClientID,  
    int nCategory[],  
    int nPriceType[],  
    const char *pszGddm[],  
    const char *pszZqdm[],  
    float fPrice[],  
    int nQuantity[],  
    int nCount,  
    char *pszResult[],
```

```
char *pszErrInfo[]);
```

功能:

单账户批量委托交易证券; 用数组传入每个委托的参数, 数组第  $i$  个元素表示第  $i$  个委托的相应参数

参数:

nClientID	- 客户端 ID
nCategory	- 委托种类的数组, 第 $i$ 个元素表示第 $i$ 个委托的种类 0 买入 1 卖出 2 融资买入 3 融券卖出 4 买券还券 5 卖券还款 6 现券还券 7 担保品买入 8 担保品卖出
nPriceType	- 报价方式的数组, 第 $i$ 个元素表示第 $i$ 个委托的报价方式 0 限价委托: 上海限价委托 / 深圳限价委托 1 市价委托(深圳对方最优价格) 2 市价委托(深圳本方最优价格) 3 市价委托(深圳即时成交剩余撤销) 4 市价委托(上海五档即成剩撤 / 深圳五档即成剩撤) 5 市价委托(深圳全额成交或撤销) 6 市价委托(上海五档即成转限价)
pszGddm	- 股东代码数组, 第 $i$ 个元素表示第 $i$ 个委托的股东代码; 交易上海股票填上海的股东代码, 交易深圳的股票填入深圳的股东代码
pszZqdm	- 证券代码数组, 第 $i$ 个元素表示第 $i$ 个委托的证券代码
fPrice	- 委托价格数组, 第 $i$ 个元素表示第 $i$ 个委托的委托价格
nQuantity	- 委托数量数组, 第 $i$ 个元素表示第 $i$ 个委托的委托数量
nCount	- 委托的个数, 即数组的长度
pszResult	- 返回数据的数组, 第 $i$ 个元素表示第 $i$ 个委托的返回信息. 此 API 执行返回后, Result[i]

含义同上。

pszErrInfo	- 错误信息的数组, 第 $i$ 个元素表示第 $i$ 个委托的错误信息. 此 API 执行返回后, ErrInfo[i]
------------	---

含义同上。

返回值:

### 3.1.15. CancelOrders 批量撤委托

```
int WINAPI CancelOrders(
    int nClientID,
    char nMarket[],
```



```
const char *pszOrderID[],
int nCount,
char *pszResult[],
char *pszErrInfo[]);
```

**功能:**

批量撤委托: 用数组传入每个委托的参数, 数组第 i 个元素表示第 i 个撤委托的相应参数

**参数:**

nClientID	- 客户端 ID
nMarket	- 交易所 ID 0 深圳 (招商证券, 普通账户深圳是 2) 1 上海
pszOrderID	- 要撤的目标委托的编号
nCount	- 要撤委托的个数, 即数组的长度
pszResult	- 返回数据的数组, 第 i 个元素表示第 i 个委托的返回信息. 此 API 执行返回后, Result[i]

含义同上。

pszErrInfo	- 错误信息的数组, 第 i 个元素表示第 i 个委托的错误信息. 此 API 执行返回后, ErrInfo[i]
------------	---

含义同上。

**返回值:**

xxx, 失败时返回负数

---

### 3.1.16. GetQuotes

```
int WINAPI GetQuotes(
    int nClientID,
    const char *pszZqdm[],
    int nCount,
    char *pszResult[],
    char *pszErrInfo[]);
```

**功能:**

单账户批量获取证券的实时五档报价

**参数:**

nClientID	- 客户端 ID
pszZqdm	- 证券代码
nCount	- 待查询证券的数量, 即证券代码数组的长度
pszResult	- 返回数据的数组, 第 i 个元素表示第 i 个委托的返回信息. 此 API 执行返回后, Result[i]

含义同上。

pszErrInfo	- 错误信息的数组, 第 i 个元素表示第 i 个委托的错误信息. 此 API 执行返回后, ErrInfo[i]
------------	---

含义同上。

返回值:

xxx, 失败时返回负数

---

### 3.1.17. QueryMultiAccountsDatas

```
int WINAPI QueryMultiAccountsDatas(  
    int nClientID[],  
    int nCategory[],  
    int nCount,  
    char *pszResult[],  
    char *pszErrInfo[]);
```

功能:

批量向不同账户查询各类交易数据; 用数组传入每个委托的参数, 数组第 i 个元素表示第 i 个查询的相应参数

参数:

- |            |   |
|------------|---|
| nClientID  | - 客户端 ID  |
| nCategory  | - 信息种类的数组, 第 i 个元素表示第 i 个查询的信息种类                          |
|            | 0 资金  |
|            | 1 股份  |
|            | 2 当日委托  |
|            | 3 当日成交  |
|            | 4 可撤单   |
|            | 5 股东代码  |
|            | 6 融资余额  |
|            | 7 融券余额  |
|            | 8 可融证券  |
|            | 9   |
|            | 10  |
|            | 11  |
|            | 12 可申购新股查询  |
|            | 13 新股申购额度查询   |
|            | 14 配号查询   |
|            | 15 中签查询   |
| nCount     | - 查询的个数, 即数组的长度   |
| pszResult  | - 返回数据的数组, 第 i 个元素表示第 i 个委托的返回信息. 此 API 执行返回后, Result[i]  |
| pszErrInfo | - 错误信息的数组, 第 i 个元素表示第 i 个委托的错误信息. 此 API 执行返回后, ErrInfo[i] |

含义同上。

含义同上。

返回值:

xxx, 失败时返回负数

### 3.1.18. SendMultiAccountsOrders

```
int WINAPI SendMultiAccountsOrders(
    int nClientID[],
    int nCategory[],
    int nPriceType[],
    const char *pszGddm[],
    const char *pszZqdm[],
    float fPrice[],
    int nQuantity[],
    int nCount,
    char *pszResult[],
    char *pszErrInfo[]);
```

功能:

向不同账户批量下委托单; 用数组传入每个委托的参数, 数组第 i 个元素表示第 i 个委托的相应参数

参数:

- |            |                                      |
|------------|--------------------------------------|
| nClientID  | - 客户端 ID 数组, 第 i 个元素表示第 i 个委托的客户端 ID |
| nCategory  | - 委托种类的数组, 第 i 个元素表示第 i 个委托的种类       |
|            | 0 买入                                 |
|            | 1 卖出                                 |
|            | 2 融资买入                               |
|            | 3 融券卖出                               |
|            | 4 买券还券                               |
|            | 5 卖券还款                               |
|            | 6 现券还券                               |
| nPriceType | - 报价方式的数组, 第 i 个元素表示第 i 个委托的报价方式     |
|            | 0 限价委托: 上海限价委托 / 深圳限价委托              |
|            | 1 市价委托(深圳对方最优价格)                     |
|            | 2 市价委托(深圳本方最优价格)                     |
|            | 3 市价委托(深圳即时成交剩余撤销)                   |
|            | 4 市价委托(上海五档即成剩撤 / 深圳五档即成剩撤)          |
|            | 5 市价委托(深圳全额成交或撤销)                    |
|            | 6 市价委托(上海五档即成转限价)                    |
| pszGddm    | - 股票代码数组, 第 i 个元素表示第 i 个委托的股票代码;     |
|            | 交易上海股票填上海的股票代码, 交易深圳的股票填入深圳的股票代码     |
| pszZqdm    | - 证券代码数组, 第 i 个元素表示第 i 个委托的证券代码      |
| fPrice     | - 委托价格数组, 第 i 个元素表示第 i 个委托的委托价格      |
| nQuantity  | - 委托数量数组, 第 i 个元素表示第 i 个委托的委托数量      |
| nCount     | - 委托的个数, 即数组的长度                      |

pszResult - 返回数据的数组，第 i 个元素表示第 i 个委托的返回信息。此 API 执行返回后，Result[i] 含义同上。

pszErrInfo - 错误信息的数组，第 i 个元素表示第 i 个委托的错误信息。此 API 执行返回后，ErrInfo[i] 含义同上。

返回值：

xxx，失败时返回负数

---

### 3.1.19. CancelMultiAccountsOrders

```
int WINAPI CancelMultiAccountsOrders(  
    int nClientID[],  
    char nMarket[],  
    const char *pszOrderID[],  
    int nCount,  
    char *pszResult[],  
    char *pszErrInfo[]);
```

功能：

批量向不同账户撤单；用数组传入每个委托的参数，数组第 i 个元素表示第 i 个撤委托的相应参数

参数：

nClientID - 客户端 ID  
nMarket - 交易所 ID  
0 深圳（招商证券，普通账户深圳是 2）  
1 上海

pszOrderID - 要撤的目标委托的编号

nCount - 要撤委托的个数，即数组的长度

pszResult - 返回数据的数组，第 i 个元素表示第 i 个委托的返回信息。此 API 执行返回后，Result[i] 含义同上。

pszErrInfo - 错误信息的数组，第 i 个元素表示第 i 个委托的错误信息。此 API 执行返回后，ErrInfo[i] 含义同上。

返回值：

xxx，失败时返回负数

---

### 3.1.20. GetMultiAccountsQuotes

```
int WINAPI GetMultiAccountsQuotes(  
    int nClientID[],
```

```
const char *pszZqdm[],  
int nCount,  
char *pszResult[],  
char *pszErrInfo[]);
```

功能:

批量向不同账户获取实时五档报价

参数:

nClientID	- 客户端 ID 数组, 第 i 个元素表示第 i 个查询的客户端 ID
pszZqdm	- 证券代码
nCount	- 待查询证券的数量, 即证券代码数组的长度
pszResult	- 返回数据的数组, 第 i 个元素表示第 i 个委托的返回信息. 此 API 执行返回后, Result[i]

含义同上。

pszErrInfo	- 错误信息的数组, 第 i 个元素表示第 i 个委托的错误信息. 此 API 执行返回后, ErrInfo[i]
------------	---

含义同上。

返回值:

xxx, 失败时返回负数

---

## 4. 普通行情 API

- 所有行情函数均为客户端主动请求查询, 而非服务器推送。
- 五档实时行情服务器侧允许单次请求股票的最大数量为 80; 假如需要获取更多股票, 需要分批循环查询。例如查询 3000 支股票, 则需要 40\*75。
- 实时行情数据每隔 3 秒刷新一次。

### 4.1. 使用流程

- 1) 应用程序先调用 TdxHq\_Connect 连接通达信行情服务器;
- 2) 然后才可以调用其他接口获取行情数据, 应用程序应自行处理网络断线问题, 接口是线程安全的。

## 4.2. 函数

### 4.2.1. TdxHq\_Connect

```
int WINAPI TdxHq_Connect(  
    const char *pszIP,  
    short nPort,  
    char *pszResult,  
    char *pszErrInfo);
```

功能:

连接通达信行情服务器

参数:

pszIP	- 服务器 IP, 可在券商软件登录界面中的通讯设置中查得。
nPort	- 服务器端口
pszResult	- 此 API 执行返回后, Result 内保存了返回的查询数据, 形式为表格数据, 行数据之间通过\n 字符分割, 列数据之间通过\t 分隔。一般要分配 1024*1024 字节的空间。出错时空字符串。
pszErrInfo	- 此 API 执行返回后, 如果出错, 保存了错误信息说明。一般要分配 256 字节的空间。没出错时空字符串。

返回值:

整型值。成功返回行情连接代码, 失败返回负值。

---

### 4.2.2. TdxHq\_Disconnect

```
void WINAPI TdxHq_Disconnect(int nConnID);
```

功能:

断开同服务器的连接

参数:

nConnID	- 行情连接代码
---------	----------

返回值: 无

---

### 4.2.3. TdxHq\_GetSecurityCount

```
bool WINAPI TdxHq_GetSecurityCount(  
    int nConnID,  
    char nMarket,  
    short *pnCount,  
    char *pszErrInfo);
```

功能:

获取指定市场内的证券数目

参数:

nConnID	- 行情连接代码
nMarket	- 市场代码 0 深圳 1 上海
pnCount	- 此 API 执行返回后, 保存了返回的证券数量
pszErrInfo	- 此 API 执行返回后, 如果出错, 保存了错误信息说明。一般要分配 256 字节的空间。没出错时为空字符串。

返回值:

成功返回 true, 失败返回 false

---

### 4.2.4. TdxHq\_GetSecurityList

```
bool WINAPI TdxHq_GetSecurityList(  
    int nConnID,  
    char nMarket,  
    short nStart,  
    short *pnCount,  
    char *pszResult,  
    char *pszErrInfo);
```

功能:

获取市场内某个范围内的 1000 支股票的股票代码

参数:

nConnID	- 行情连接代码
nMarket	- 市场代码 0 深圳 1 上海
nStart	- 范围开始位置, 第一个股票是 0, 第二个是 1, 依此类推,

位置信息依据 TdxHq\_GetSecurityCount 返回的证券总数确定

pnCount - 范围的大小, API 执行后, 保存了实际返回的股票数目

pszResult - 此 API 执行返回后, Result 内保存了返回的证券代码信息, 形式为表格数据, 行数据之间通过\n 字符分割, 列数据之间通过\t 分隔。  
一般要分配 1024\*1024 字节的空间。出错时空字符串。

pszErrInfo - 此 API 执行返回后, 如果出错, 保存了错误信息说明。一般要分配 256 字节的空间。没出错时空字符串。

返回值:

成功返回 true, 失败返回 false

## 4.2.5. TdxHq\_GetSecurityQuotes

```
bool WINAPI TdxHq_GetSecurityQuotes(
    int nConnID,
    char nMarket[],
    const char *pszZqdm[],
    short *pnCount,
    char *pszResult,
    char *pszErrInfo);
```

功能:

批量获取多个证券的五档报价数据

参数:

nConnID - 行情连接代码

nMarket - 市场代码; 第 i 个元素表示第 i 个证券的市场代码  
0 深圳  
1 上海

pszZqdm - 证券代码, pnCount 个证券代码组成的数组

pnCount - API 执行前, 表示用户要请求的证券数目, 最大 80;  
API 执行后, 保存了实际返回的数目

pszResult - 此 API 执行返回后, Result 内保存了返回的证券代码信息, 形式为表格数据, 行数据之间通过\n 字符分割, 列数据之间通过\t 分隔。  
一般要分配 1024\*1024 字节的空间。出错时空字符串。

pszErrInfo - 此 API 执行返回后, 如果出错, 保存了错误信息说明。一般要分配 256 字节的空间。没出错时空字符串。

返回值:

成功返回 true, 失败返回 false



## 4.2.6. TdxHq\_GetSecurityBars

```
bool WINAPI TdxHq_GetSecurityBars(
    int nConnID,
    char nCategory,
    char nMarket,
    const char *pszZqdm,
    short nStart,
    short *pnCount,
    char *pszResult,
    char *pszErrInfo);
```

功能:

获取某个范围内的证券 K 线数据

参数:

nConnID	- 行情连接代码
nCategory	- K 线种类
	0 5 分钟 K 线
	1 15 分钟 K 线
	2 30 分钟 K 线
	3 1 小时 K 线
	4 日 K 线
	5 周 K 线
	6 月 K 线
	7 1 分钟
	8 1 分钟 K 线
	9 日 K 线
	10 季 K 线
	11 年 K 线
nMarket	- 市场代码
	0 深圳
	1 上海
pszZqdm	- 证券代码
nStart	- 范围开始位置, 最后一条 K 线位置是 0, 前一条是 1, 依此类推
pnCount	- 范围大小, API 执行前, 表示用户要请求的 K 线数目, API 执行后, 保存了实际返回的 K 线数目, 最大值 800;
pszResult	- 此 API 执行返回后, Result 内保存了返回的查询数据, 形式为表格数据; 行数据之间通过\n 字符分割, 列数据之间通过\t 分隔; 一般要分配 1024*1024 字节的空间。出错时空字符串。
pszErrInfo	- 此 API 执行返回后, 如果出错, 保存了错误信息说明。 一般要分配 256 字节的空间。没出错时空字符串。

返回值:

成功返回 true，失败返回 false

---

## 4.2.7. TdxHq\_GetIndexBars

```
bool WINAPI TdxHq_GetIndexBars(  
    int nConnID,  
    char nCategory,  
    char nMarket,  
    const char *pszZqdm,  
    short nStart,  
    short *pnCount,  
    char *pszResult,  
    char *pszErrInfo);
```

功能:

获取某个范围内的指数 K 线数据

参数:

nConnID	- 行情连接代码
nCategory	- K 线种类
	0 5 分钟 K 线
	1 15 分钟 K 线
	2 30 分钟 K 线
	3 1 小时 K 线
	4 日 K 线
	5 周 K 线
	6 月 K 线
	7 1 分钟
	8 1 分钟 K 线
	9 日 K 线
	10 季 K 线
	11 年 K 线
nMarket	- 市场代码
	0 深圳
	1 上海
pszZqdm	- 证券代码
nStart	- 范围开始位置, 最后一条 K 线位置是 0, 前一条是 1, 依此类推
nCount	- 范围大小, API 执行前, 表示用户要请求的 K 线数目, API 执行后, 保存了实际返回的 K 线数目, 最大值 800;
pszResult	- 此 API 执行返回后, Result 内保存了返回的查询数据, 形式为表格数据; 行数据之间通过\n 字符分割, 列数据之间通过\t 分隔; 一般要分配 1024*1024 字节的空间。出错时为空字符串。
pszErrInfo	- 此 API 执行返回后, 如果出错, 保存了错误信息说明。

一般要分配 256 字节的空间。没出错时为空字符串。

返回值:

成功返回 true, 失败返回 false

---

#### 4.2.8. TdxHq\_GetMinuteTimeData

```
bool WINAPI TdxHq_GetMinuteTimeData(  
    int nConnID,  
    char nMarket,  
    const char *pszZqdm,  
    char *pszResult,  
    char *pszErrInfo);
```

功能:

获取分时行情数据

参数:

nConnID	- 行情连接代码
nMarket	- 市场代码 0 深圳 1 上海
pszZqdm	- 证券代码
pszResult	- 此 API 执行返回后, Result 内保存了返回的查询数据, 形式为表格数据; 行数据之间通过\n 字符分割, 列数据之间通过\t 分隔; 一般要分配 1024*1024 字节的空间。出错时为空字符串。
pszErrInfo	- 此 API 执行返回后, 如果出错, 保存了错误信息说明。 一般要分配 256 字节的空间。没出错时为空字符串。

返回值:

成功返回 true, 失败返回 false

---

#### 4.2.9. TdxHq\_GetHistoryMinuteTimeData

```
bool WINAPI TdxHq_GetHistoryMinuteTimeData(  
    int nConnID,  
    char nMarket,  
    const char *pszZqdm,  
    int nDate,  
    char *pszResult,
```

```
char *pszErrInfo);
```

功能:

获取历史分时行情数据

参数:

- |            |  |
|------------|--|
| nConnID    | - 行情连接代码   |
| nMarket    | - 市场代码<br>0 深圳<br>1 上海   |
| pszZqdm    | - 证券代码   |
| nDate      | - 日期, 比如 2017 年 2 月 1 日为整数 20170201  |
| pszResult  | - 此 API 执行返回后, Result 内保存了返回的查询数据, 形式为表格数据;<br>行数据之间通过\n 字符分割, 列数据之间通过\t 分隔;<br>一般要分配 1024*1024 字节的空间。出错时空字符串。 |
| pszErrInfo | - 此 API 执行返回后, 如果出错, 保存了错误信息说明。<br>一般要分配 256 字节的空间。没出错时空字符串。   |

返回值:

成功返回 true, 失败返回 false

---

## 4.2.10. TdxHq\_GetTransactionData

```
bool WINAPI TdxHq_GetTransactionData(  
    int nConnID,  
    char nMarket,  
    const char *pszZqdm,  
    short nStart,  
    short *pnCount,  
    char *pszResult,  
    char *pszErrInfo);
```

功能:

获取某个范围的分笔成交数据

参数:

- |         |  |
|---------|--|
| nConnID | - 行情连接代码   |
| nMarket | - 市场代码<br>0 深圳<br>1 上海                                 |
| pszZqdm | - 证券代码   |
| nStart  | - 范围开始位置, 最后一条 K 线位置是 0, 前一条是 1, 依此类推                  |
| pnCount | - 范围大小, API 执行前, 表示用户要请求的 K 线数目, API 执行后, 保存了实际返回的 K 线 |

数目;

- pszResult - 此 API 执行返回后, Result 内保存了返回的查询数据, 形式为表格数据;  
行数据之间通过\n 字符分割, 列数据之间通过\t 分隔;  
一般要分配 1024\*1024 字节的空间。出错时为空字符串。
- pszErrInfo - 此 API 执行返回后, 如果出错, 保存了错误信息说明。  
一般要分配 256 字节的空间。没出错时为空字符串。

返回值:

成功返回 true, 失败返回 false

## 4.2.11. TdxHq\_GetHistoryTransactionData

```
bool WINAPI TdxHq_GetHistoryTransactionData(
    int nConnID,
    char nMarket,
    const char *pszZqdm,
    short nStart,
    short *pnCount,
    int date,
    char *pszResult,
    char *pszErrInfo);
```

功能:

获取某个范围的历史分笔成交数据

券商行情服务器侧限制最大允许单次返回 2000 条记录, 如果需要更多数据则需要多次调用循环读取。

参数:

- nConnID - 行情连接代码
- nMarket - 市场代码  
0 深圳  
1 上海
- pszZqdm - 证券代码
- nStart - 范围开始位置, 最后一条 K 线位置是 0, 前一条是 1, 依此类推
- pnCount - 范围大小, API 执行前, 表示用户要请求的 K 线数目, API 执行后, 保存了实际返回的 K 线

数目;

- date - 日期, 比如 2017 年 2 月 1 日为整数 20170201
- pszResult - 此 API 执行返回后, Result 内保存了返回的查询数据, 形式为表格数据;  
行数据之间通过\n 字符分割, 列数据之间通过\t 分隔;  
一般要分配 1024\*1024 字节的空间。出错时为空字符串。
- pszErrInfo - 此 API 执行返回后, 如果出错, 保存了错误信息说明。  
一般要分配 256 字节的空间。没出错时为空字符串。

返回值:

成功返回 true，失败返回 false

---

## 4.2.12. TdxHq\_GetCompanyInfoCategory

```
bool WINAPI TdxHq_GetCompanyInfoCategory(  
    int nConnID,  
    char nMarket,  
    const char *pszZqdm,  
    char *pszResult,  
    char *pszErrInfo);
```

功能:

获取 F10 资料的类别

参数:

nConnID	- 行情连接代码
nMarket	- 市场代码 0 深圳 1 上海
pszZqdm	- 证券代码
pszResult	- 此 API 执行返回后，Result 内保存了返回的查询数据，形式为表格数据； 行数据之间通过\n 字符分割，列数据之间通过\t 分隔； 一般要分配 1024*1024 字节的空间。出错时为空字符串。
pszErrInfo	- 此 API 执行返回后，如果出错，保存了错误信息说明。 一般要分配 256 字节的空间。没出错时为空字符串。

返回值:

成功返回 true，失败返回 false

---

## 4.2.13. TdxHq\_GetCompanyInfoContent

```
bool WINAPI TdxHq_GetCompanyInfoContent(  
    int nConnID,  
    char nMarket,  
    const char *pszZqdm,  
    const char *pszFileName,  
    int nStart,  
    int nLength,  
    char *pszResult,  
    char *pszErrInfo);
```

**功能:**

获取某一类 F10 资料的具体内容

**参数:**

nConnID	- 行情连接代码
nMarket	- 市场代码 0 深圳 1 上海
pszZqdm	- 证券代码
pszFileName	- 类目的文件名, 由 TdxHq_GetCompanyInfoCategory 返回信息中获取
nStart	- 类目的开始位置, 由 TdxHq_GetCompanyInfoCategory 返回信息中获取
nLength	- 类目的长度, 由 TdxHq_GetCompanyInfoCategory 返回信息中获取
pszResult	- 此 API 执行返回后, Result 内保存了返回的查询数据。出错时为空字符串。
pszErrInfo	- 此 API 执行返回后, 如果出错, 保存了错误信息说明。 一般要分配 256 字节的空间。没出错时为空字符串。

**返回值:**

成功返回 true, 失败返回 false

---

## 4.2.14. TdxHq\_GetXDXRInfo

```
bool WINAPI TdxHq_GetXDXRInfo(  
    int nConnID,  
    char nMarket,  
    const char *pszZqdm,  
    char *pszResult,  
    char *pszErrInfo);
```

**功能:**

获取除权除息数据

**参数:**

nConnID	- 行情连接代码
nMarket	- 市场代码 0 深圳 1 上海
pszZqdm	- 证券代码
pszResult	- 此 API 执行返回后, Result 内保存了返回的查询数据, 出错时为空字符串。
pszErrInfo	- 此 API 执行返回后, 如果出错, 保存了错误信息说明。 一般要分配 256 字节的空间。没出错时为空字符串。

**返回值:**

成功返回 true，失败返回 false

返回信息的保留字段含义：

- 1 除权除息
  - 2 送配股上市
  - 3 非流通股上市
  - 4 未知股本变动
  - 5 股本变化
  - 6 增发新股
  - 7 股份回购
  - 8 增发新股上市
  - 9 转配股上市
  - 10 可转债上市
  - 11 扩缩股
  - 12 非流通股缩股
  - 13 送认购权证
  - 14 送认沽权证
- 

## 4.2.15. TdxHq\_GetFinanceInfo

```
bool WINAPI TdxHq_GetFinanceInfo(  
    int nConnID,  
    char nMarket,  
    const char *pszZqdm,  
    char *pszResult,  
    char *pszErrInfo);
```

功能：

获取财务数据

参数：

- |            |   |
|------------|---|
| nConnID    | - 行情连接代码                                  |
| nMarket    | - 市场代码                                    |
|            | 0 深圳                                      |
|            | 1 上海                                      |
| pszZqdm    | - 证券代码                                    |
| pszResult  | - 此 API 执行返回后，Result 内保存了返回的查询数据，出错时空字符串。 |
| pszErrInfo | - 此 API 执行返回后，如果出错，保存了错误信息说明。             |
|            | 一般要分配 256 字节的空间。没出错时空字符串。                 |

返回值：

成功返回 true，失败返回 false



## 5. Level 2 行情 API

所有行情函数均为客户端主动请求查询，不是服务器推送；Level 2 行情 API 包括以下 19 个函数：

Level 2 行情服务器侧允许单次请求股票的最大数量为 50；假如需要获取更多股票，需要分批循环查询。例如查询 3000 支股票，则需要  $50 \times 60 = 3000$ 。

### 5.1. 使用流程

应用程序先调用 TdxL2Hq\_Connect 连接通达信行情服务器, 然后才可以调用其他接口获取行情数据, 应用程序应自行处理网络断线问题，接口是线程安全的

### 5.2. 函数

#### 5.2.1. TdxL2Hq\_Connect

```
int WINAPI TdxL2Hq_Connect(  
    int nConnID,  
    const char *pszIP,  
    short nPort,  
    const char *pszL2User,  
    const char *pszL2Password,  
    char *pszResult,  
    char *pszErrInfo);
```

功能：

连接券商行情服务器

参数：

nConnID	- 行情连接代码
pszIP	- 服务器 IP, 可在演示版内查得
nPort	- 服务器端口

- |               |   |
|---------------|---|
| pszL2User     | - 通达信 Level 2 行情账户名，用户需向通达信购买   |
| pszL2Password | - 通达信 Level 2 行情账户密码  |
| pszResult     | - 此 API 执行返回后，Result 内保存了返回的查询数据，形式为表格数据，行数据之间通过\n 字符分割，列数据之间通过\t 分隔。<br>一般要分配 1024*1024 字节的空间。出错时空字符串。 |
| pszErrInfo    | - 此 API 执行返回后，如果出错，保存了错误信息说明。<br>一般要分配 256 字节的空间。没出错时空字符串。  |

返回值：

整型值。成功返回行情连接代码，失败返回负值。

适用版本： 标准版，批量版，多账户版

---

## 5.2.2. TdxL2Hq\_Disconnect

```
void WINAPI TdxL2Hq_Disconnect(int nConnID);
```

功能：

断开同服务器的连接

参数：

nConnID	- 行情连接代码
---------	----------

适用版本： 标准版，批量版，多账户版

---

## 5.2.3. TdxL2Hq\_GetSecurityCount

```
bool WINAPI TdxL2Hq_GetSecurityCount(  
    int nConnID,  
    char nMarket,  
    short *nCount,  
    char *pszErrInfo);
```

功能：

获取指定市场内的所有证券数量

参数：

nConnID	- 行情连接代码
nMarket	- 市场代码 0 深圳 1 上海

- nCount - 此 API 执行返回后，保存了返回的证券数量
- pszErrInfo - 此 API 执行返回后，如果出错，保存了错误信息说明。  
一般要分配 256 字节的空间。没出错时为空字符串。

返回值:

成功返回 true，失败返回 false

---

## 5.2.4. TdxL2Hq\_GetSecurityList

```
bool WINAPI TdxL2Hq_GetSecurityList(  
    int nConnID,  
    char nMarket,  
    short nStart,  
    short *nCount,  
    char *pszResult,  
    char *pszErrInfo);
```

功能:

获取市场内某个范围内的 1000 支股票的股票代码

参数:

- nConnID - 行情连接代码
- nMarket - 市场代码  
0 深圳  
1 上海
- nStart - 范围开始位置, 第一个股票是 0, 第二个是 1, 依此类推;  
位置信息依据 TdxL2Hq\_GetSecurityCount 返回的证券总数量确定
- nCount - 范围大小, API 执行后, 保存了实际返回的股票数目
- pszResult - 此 API 执行返回后, Result 内保存了返回的证券代码信息, 形式为表格数据;  
行数据之间通过\n 字符分割, 列数据之间通过\t 分隔。  
一般要分配 1024\*1024 字节的空间。出错时为空字符串。
- pszErrInfo - 此 API 执行返回后, 如果出错, 保存了错误信息说明。  
一般要分配 256 字节的空间。没出错时为空字符串。

返回值:

成功返回 true，失败返回 false

---

## 5.2.5. TdxL2Hq\_GetSecurityQuotes

```
bool WINAPI TdxL2Hq_GetSecurityQuotes(  
    int nConnID,
```

```
char nMarket[],  
const char *pszZqdm[],  
short *nCount,  
char *pszResult,  
char *pszErrInfo);
```

功能:

批量获取多个证券的五档报价数据

参数:

nConnID	- 行情连接代码
nMarket	- 市场代码; 第 i 个元素表示第 i 个证券的市场代码 0 深圳 1 上海
pszZqdm	- 证券代码, nCount 个证券代码组成的数组;
nCount	- API 执行前, 表示用户要请求的证券数目, 最大 80 (不同券商可能不同, 具体请咨询券商

或测试);

	API 执行后, 保存了实际返回的股票数目。
pszResult	- 此 API 执行返回后, Result 内保存了返回的证券代码信息, 形式为表格数据; 行数据之间通过\n 字符分割, 列数据之间通过\t 分隔。 一般要分配 1024*1024 字节的空间。出错时为空字符串。
pszErrInfo	- 此 API 执行返回后, 如果出错, 保存了错误信息说明。 一般要分配 256 字节的空间。没出错时为空字符串。

返回值:

成功返回 true, 失败返回 false

---

## 5.2.6. TdxL2Hq\_GetSecurityBars

```
bool WINAPI TdxL2Hq_GetSecurityBars(  
    int nConnID,  
    char nCategory,  
    char nMarket,  
    const char *pszZqdm,  
    short nStart,  
    short *nCount,  
    char *pszResult,  
    char *pszErrInfo);
```

功能:

获取证券在某个范围内的 K 线数据

参数:

nConnID	- 行情连接代码
nCategory	- K 线种类
	0 5 分钟 K 线
	1 15 分钟 K 线
	2 30 分钟 K 线
	3 1 小时 K 线
	4 日 K 线
	5 周 K 线
	6 月 K 线
	7 1 分钟
	8 1 分钟 K 线
	9 日 K 线
	10 季 K 线
	11 年 K 线
nMarket	- 市场代码
	0 深圳
	1 上海
pszZqdm	- 证券代码
nStart	- 范围开始位置, 最后一条 K 线位置是 0, 前一条是 1, 依此类推
nCount	- 范围大小, API 执行前, 表示用户要请求的 K 线数目, API 执行后, 保存了实际返回的 K 线数目, 最大值 800;
pszResult	- 此 API 执行返回后, Result 内保存了返回的查询数据, 形式为表格数据: 行数据之间通过\n 字符分割, 列数据之间通过\t 分隔; 一般要分配 1024*1024 字节的空间。出错时空字符串。
pszErrInfo	- 此 API 执行返回后, 如果出错, 保存了错误信息说明。 一般要分配 256 字节的空间。没出错时空字符串。

返回值:

成功返回 true, 失败返回 false

## 5.2.7. TdxL2Hq\_GetIndexBars

```
bool WINAPI TdxL2Hq_GetIndexBars(
    int nConnID,
    char nCategory,
    char nMarket,
    const char *pszZqdm,
    short nStart,
    short *nCount,
    char *pszResult,
    char *pszErrInfo);
```

功能:

获取指数在某个范围内的 K 线数据

参数:

nConnID	- 行情连接代码
nCategory	- K 线种类
	0 5 分钟 K 线
	1 15 分钟 K 线
	2 30 分钟 K 线
	3 1 小时 K 线
	4 日 K 线
	5 周 K 线
	6 月 K 线
	7 1 分钟
	8 1 分钟 K 线
	9 日 K 线
	10 季 K 线
	11 年 K 线
nMarket	- 市场代码
	0 深圳
	1 上海
pszZqdm	- 证券代码
nStart	- 范围开始位置, 最后一条 K 线位置是 0, 前一条是 1, 依此类推
nCount	- 范围大小, API 执行前, 表示用户要请求的 K 线数目, API 执行后, 保存了实际返回的 K 线数目, 最大值 800;
pszResult	- 此 API 执行返回后, Result 内保存了返回的查询数据, 形式为表格数据; 行数据之间通过\n 字符分割, 列数据之间通过\t 分隔; 一般要分配 1024*1024 字节的空间。出错时空字符串。
pszErrInfo	- 此 API 执行返回后, 如果出错, 保存了错误信息说明。 一般要分配 256 字节的空间。没出错时空字符串。

返回值:

成功返回 true, 失败返回 false

## 5.2.8. TdxL2Hq\_GetMinuteTimeData

```
bool WINAPI TdxL2Hq_GetMinuteTimeData(
    int nConnID,
    char nMarket,
    const char *pszZqdm,
    char *pszResult,
    char *pszErrInfo);
```

功能:

## 获取分时行情数据

### 参数:

nConnID	- 行情连接代码
nMarket	- 市场代码 0 深圳 1 上海
pszZqdm	- 证券代码
pszResult	- 此 API 执行返回后, Result 内保存了返回的查询数据, 形式为表格数据; 行数据之间通过\n 字符分割, 列数据之间通过\t 分隔; 一般要分配 1024*1024 字节的空间。出错时为空字符串。
pszErrInfo	- 此 API 执行返回后, 如果出错, 保存了错误信息说明。 一般要分配 256 字节的空间。没出错时为空字符串。

### 返回值:

成功返回 true, 失败返回 false

---

## 5.2.9. TdxL2Hq\_GetHistoryMinuteTimeData

```
bool WINAPI TdxL2Hq_GetHistoryMinuteTimeData(  
    int nConnID,  
    char nMarket,  
    const char *pszZqdm,  
    int nDate,  
    char *pszResult,  
    char *pszErrInfo);
```

### 功能:

获取历史分时行情数据

### 参数:

nConnID	- 行情连接代码
nMarket	- 市场代码 0 深圳 1 上海
pszZqdm	- 证券代码
nDate	- 日期, 比如 2017 年 2 月 1 日为整数 20170201
pszResult	- 此 API 执行返回后, Result 内保存了返回的查询数据, 形式为表格数据; 行数据之间通过\n 字符分割, 列数据之间通过\t 分隔; 一般要分配 1024*1024 字节的空间。出错时为空字符串。
pszErrInfo	- 此 API 执行返回后, 如果出错, 保存了错误信息说明。 一般要分配 256 字节的空间。没出错时为空字符串。

返回值:

成功返回 true, 失败返回 false

---

## 5.2.10. TdxL2Hq\_GetTransactionData

```
bool WINAPI TdxL2Hq_GetTransactionData(  
    int nConnID,  
    char nMarket,  
    const char *pszZqdm,  
    short nStart,  
    short *nCount,  
    char *pszResult,  
    char *pszErrInfo);
```

功能:

获取某个范围内的分笔成交数据

参数:

nConnID	- 行情连接代码
nMarket	- 市场代码 0 深圳 1 上海
pszZqdm	- 证券代码
nStart	- 范围开始位置, 最后一条 K 线位置是 0, 前一条是 1, 依此类推
nCount	- 范围大小, API 执行前, 表示用户要请求的 K 线数目, API 执行后, 保存了实际返回的 K 线

数目:

pszResult	- 此 API 执行返回后, Result 内保存了返回的查询数据, 形式为表格数据; 行数据之间通过\n 字符分割, 列数据之间通过\t 分隔; 一般要分配 1024*1024 字节的空间。出错时空字符串。
pszErrInfo	- 此 API 执行返回后, 如果出错, 保存了错误信息说明。 一般要分配 256 字节的空间。没出错时空字符串。

返回值:

成功返回 true, 失败返回 false

---

## 5.2.11. TdxL2Hq\_GetHistoryTransactionData

```
bool WINAPI TdxL2Hq_GetHistoryTransactionData(  
    int nConnID,  
    char nMarket,
```



```
const char *pszZqdm,  
short nStart,  
short *nCount,  
int date,  
char *pszResult,  
char *pszErrInfo);
```

**功能:**

获取某个范围内的历史分笔成交数据

券商行情服务器侧限制最大允许单次返回 2000 条记录，如果需要更多数据则需要多次调用循环读取。

**参数:**

nConnID	- 行情连接代码
nMarket	- 市场代码 0 深圳 1 上海
pszZqdm	- 证券代码
nStart	- 范围开始位置, 最后一条 K 线位置是 0, 前一条是 1, 依此类推
nCount	- 范围大小, API 执行前, 表示用户要请求的 K 线数目, API 执行后, 保存了实际返回的 K 线

**数目:**

date	- 日期, 比如 2017 年 2 月 1 日为整数 20170201
pszResult	- 此 API 执行返回后, Result 内保存了返回的查询数据, 形式为表格数据: 行数据之间通过\n 字符分割, 列数据之间通过\t 分隔; 一般要分配 1024*1024 字节的空间。出错时为空字符串。
pszErrInfo	- 此 API 执行返回后, 如果出错, 保存了错误信息说明。 一般要分配 256 字节的空间。没出错时为空字符串。

**返回值:**

成功返回 true, 失败返回 false

---

## 5.2.12. TdxL2Hq\_GetCompanyInfoCategory

```
bool WINAPI TdxL2Hq_GetCompanyInfoCategory(  
    int nConnID,  
    char nMarket,  
    const char *pszZqdm,  
    char *pszResult,  
    char *pszErrInfo);
```

**功能:**

获取 F10 资料的类别

**参数:**

nConnID	- 行情连接代码
nMarket	- 市场代码 0 深圳 1 上海
pszZqdm	- 证券代码
pszResult	- 此 API 执行返回后，Result 内保存了返回的查询数据，形式为表格数据； 行数据之间通过\n 字符分割，列数据之间通过\t 分隔； 一般要分配 1024*1024 字节的空间。出错时为空字符串。
pszErrInfo	- 此 API 执行返回后，如果出错，保存了错误信息说明。 一般要分配 256 字节的空间。没出错时为空字符串。

返回值：

成功返回 true，失败返回 false

### 5.2.13. TdxL2Hq\_GetCompanyInfoContent

```
bool WINAPI TdxL2Hq_GetCompanyInfoContent(
    int nConnID,
    char nMarket,
    const char *pszZqdm,
    const char *pszFileName,
    int nStart,
    int nLength,
    char *pszResult,
    char *pszErrInfo);
```

功能：

获取某一类 F10 资料的具体内容

参数：

nConnID	- 行情连接代码
nMarket	- 市场代码 0 深圳 1 上海
pszZqdm	- 证券代码
pszFileName	- 类目的文件名，由 TdxL2Hq_GetCompanyInfoCategory 返回信息中获取
nStart	- 类目的开始位置，由 TdxL2Hq_GetCompanyInfoCategory 返回信息中获取
nLength	- 类目的长度，由 TdxL2Hq_GetCompanyInfoCategory 返回信息中获取
pszResult	- 此 API 执行返回后，Result 内保存了返回的查询数据。出错时为空字符串。
pszErrInfo	- 此 API 执行返回后，如果出错，保存了错误信息说明。 一般要分配 256 字节的空间。没出错时为空字符串。

返回值：

成功返回 true，失败返回 false

---

## 5.2.14. TdxL2Hq\_GetXDXRInfo

```
bool WINAPI TdxL2Hq_GetXDXRInfo(  
    int nConnID,  
    char nMarket,  
    const char *pszZqdm,  
    char *pszResult,  
    char *pszErrInfo);
```

功能:

获取除权除息数据

参数:

nConnID	- 行情连接代码
nMarket	- 市场代码 0 深圳 1 上海
pszZqdm	- 证券代码
pszResult	- 此 API 执行返回后，Result 内保存了返回的查询数据，出错时空字符串。
pszErrInfo	- 此 API 执行返回后，如果出错，保存了错误信息说明。 一般要分配 256 字节的空间。没出错时空字符串。

返回值:

成功返回 true，失败返回 false

---

## 5.2.15. TdxL2Hq\_GetFinanceInfo

```
bool WINAPI TdxL2Hq_GetFinanceInfo(  
    int nConnID,  
    char nMarket,  
    const char *pszZqdm,  
    char *pszResult,  
    char *pszErrInfo);
```

功能:

获取财务数据

参数:

- nConnID - 行情连接代码
- nMarket - 市场代码
  - 0 深圳
  - 1 上海
- pszZqdm - 证券代码
- pszResult - 此 API 执行返回后, Result 内保存了返回的查询数据, 出错时空字符串。
- pszErrInfo - 此 API 执行返回后, 如果出错, 保存了错误信息说明。  
一般要分配 256 字节的空间。没出错时空字符串。

返回值:

成功返回 true, 失败返回 false

## 5.2.16. TdxL2Hq\_GetSecurityQuotes10

```
bool WINAPI TdxL2Hq_GetSecurityQuotes10(
    int nConnID,
    char nMarket[],
    const char *pszZqdm[],
    short *nCount,
    char* pszResult,
    char* pszErrInfo);
```

功能:

批量获取多个证券的十档报价

参数:

- nConnID - 行情连接代码
- nMarket - 市场代码; 第 i 个元素表示第 i 个证券的市场代码
  - 0 深圳
  - 1 上海
- pszZqdm - 证券代码; Count 个证券代码组成的数组
- nCount - API 执行前, 表示用户要请求的证券数目, 最大 100 (不同券商可能不同, 具体请咨询券商或测试);

API 执行后, 保存了实际返回的股票数目。

- pszResult - 此 API 执行返回后, Result 内保存了返回的查询数据, 形式为表格数据;  
行数据之间通过\n 字符分割, 列数据之间通过\t 分隔。  
一般要分配 1024\*1024 字节的空间。出错时空字符串。
- pszErrInfo - 此 API 执行返回后, 如果出错, 保存了错误信息说明。  
一般要分配 256 字节的空间。没出错时空字符串。

返回值:

成功返回 true, 失败返回 false

---

## 5.2.17. TdxL2Hq\_GetDetailTransactionData

```
bool WINAPI TdxL2Hq_GetDetailTransactionData(  
    int nConnID,  
    char nMarket,  
    const char *pszZqdm,  
    short nStart,  
    short *nCount,  
    char* pszResult,  
    char* pszErrInfo);
```

功能:

获取某个范围内的逐笔成交数据

参数:

nConnID	- 行情连接代码
nMarket	- 市场代码 0 深圳 1 上海
pszZqdm	- 证券代码
nStart	- 范围开始位置, 最后一条 K 线位置是 0, 前一条是 1, 依此类推
nCount	- 范围大小, API 执行前, 表示用户要请求的 K 线数目; API 执行后, 保存了实际返回的 K 线数目; 最大 2000。
pszResult	- 此 API 执行返回后, Result 内保存了返回的查询数据, 形式为表格数据; 行数据之间通过\n 字符分割, 列数据之间通过\t 分隔; 一般要分配 1024*1024 字节的空间。出错时为空字符串。
pszErrInfo	- 此 API 执行返回后, 如果出错, 保存了错误信息说明。 一般要分配 256 字节的空间。没出错时为空字符串。

返回值:

成功返回 true, 失败返回 false

---

## 5.2.18. TdxL2Hq\_GetDetailOrderData

```
bool WINAPI TdxL2Hq_GetDetailOrderData(  
    int nConnID,  
    char nMarket,  
    const char *pszZqdm,  
    int nStart,  
    short *nCount,
```

```
char* pszResult,  
char* pszErrInfo);
```

功能:

获取某个范围内的逐笔委托数据

参数:

nConnID	- 行情连接代码
nMarket	- 市场代码 0 深圳 1 上海
pszZqdm	- 证券代码
nStart	- 范围开始位置, 最后一条 K 线位置是 0, 前一条是 1, 依此类推
nCount	- 范围大小, API 执行前, 表示用户要请求的 K 线数目; API 执行后, 保存了实际返回的 K 线数目; 最大 2000。
pszResult	- 此 API 执行返回后, Result 内保存了返回的查询数据, 形式为表格数据; 行数据之间通过\n 字符分割, 列数据之间通过\t 分隔; 一般要分配 1024*1024 字节的空间。出错时空字符串。
pszErrInfo	- 此 API 执行返回后, 如果出错, 保存了错误信息说明。 一般要分配 256 字节的空间。没出错时空字符串。

返回值:

成功返回 true, 失败返回 false

---

## 5.2.19. TdxL2Hq\_GetBuySellQueue

```
bool WINAPI TdxL2Hq_GetBuySellQueue(  
    int nConnID,  
    char nMarket,  
    const char* pszZqdm,  
    char* pszResult,  
    char* pszErrInfo);
```

功能:

获取买卖队列数据

参数:

nConnID	- 行情连接代码
nMarket	- 市场代码 0 深圳 1 上海
pszZqdm	- 证券代码
pszResult	- 此 API 执行返回后, Result 内保存了返回的查询数据, 形式为表格数据;

行数据之间通过\n 字符分割，列数据之间通过\t 分隔。  
一般要分配 1024\*1024 字节的空间。出错时空字符串。

pszErrInfo - 此 API 执行返回后，如果出错，保存了错误信息说明。  
一般要分配 256 字节的空间。没出错时空字符串。

返回值：

成功返回 true，失败返回 false

---

## 6. 扩展行情 API

所有行情函数均为客户端主动请求查询，不是服务器推送；扩展行情 API 包括以下 11 个函数：

### 6.1. 函数

#### 6.1.1. TdxExHq\_Connect

```
int WINAPI TdxExHq_Connect(  
    const char *pszIP,  
    short nPort,  
    char *pszResult,  
    char *pszErrInfo);
```

功能：

连接通达信扩展行情服务器

参数：

pszIP	- 服务器 IP，可在券商软件登录界面中的通讯设置中查得
nPort	- 服务器端口
pszResult	- 此 API 执行返回后，Result 内保存了返回的查询数据，形式为表格数据， 行数据之间通过\n 字符分割，列数据之间通过\t 分隔。 一般要分配 1024*1024 字节的空间。出错时空字符串。
pszErrInfo	- 此 API 执行返回后，如果出错，保存了错误信息说明。一般要分配 256 字节的空间。没 出错时空字符串。

返回值：

成功返回行情连接代码，失败返回 0。

---

### 6.1.2. TdxExHq\_Disconnect

```
void WINAPI TdxExHq_Disconnect(int nConnID);
```

功能:

断开同服务器的连接

参数:

nConnID            - 行情连接代码

返回值:     无

---

### 6.1.3. TdxExHq\_GetMarkets

```
bool WINAPI TdxExHq_GetMarkets(  
    int nConnID,  
    char *pszResult,  
    char *pszErrInfo);
```

功能:

获取扩展行情中支持的各个市场的市场代码

参数:

nConnID            - 行情连接代码

pszResult          - 此 API 执行返回后, Result 内保存了返回的查询数据, 形式为表格数据,  
                    行数据之间通过\n 字符分割, 列数据之间通过\t 分隔。  
                    一般要分配 1024\*1024 字节的空间。出错时为空字符串。

pszErrInfo         - 此 API 执行返回后, 如果出错, 保存了错误信息说明。一般要分配 256 字节的空间。没  
出错时为空字符串。

返回值:

成功返回行情连接代码, 失败返回 0。

---

### 6.1.4. TdxExHq\_GetInstrumentCount

```
bool WINAPI TdxExHq_GetInstrumentCount(  
    int nConnID,
```



```
int *nCount,  
char *pszErrInfo);
```

功能:

获取所有商品的总数量

参数:

nConnID	- 行情连接代码
nCount	- 此 API 执行返回后, 保存了返回的商品总数。
pszErrInfo	- 此 API 执行返回后, 如果出错, 保存了错误信息说明。一般要分配 256 字节的空间。没出错时为空字符串。

返回值:

成功返回行情连接代码, 失败返回 0。

---

### 6.1.5. TdxExHq\_GetInstrumentInfo

```
bool WINAPI TdxExHq_GetInstrumentInfo(  
    int nConnID,  
    int nStart,  
    short* pnCount,  
    char* pszResult,  
    char* pszErrInfo);
```

功能:

获取指定范围内商品的代码

参数:

nConnID	- 行情连接代码
nStart	- 范围的开始位置, 由 TdxExHq_GetInstrumentCount 返回信息中确定
pnCount	- 商品数量, 由 TdxExHq_GetInstrumentCount 返回信息中获取
pszResult	- 此 API 执行返回后, Result 内保存了返回的证券代码信息, 形式为表格数据, 行数据之间通过\n 字符分割, 列数据之间通过\t 分隔。一般要分配 1024*1024 字节的空间。出错时为空字符串。
pszErrInfo	- 此 API 执行返回后, 如果出错, 保存了错误信息说明。一般要分配 256 字节的空间。没出错时为空字符串。

返回值:

成功返回 true, 失败返回 false

## 6.1.6. TdxExHq\_GetInstrumentQuote

```
bool WINAPI TdxExHq_GetInstrumentQuote(  
    int nConnID,  
    char nMarket,  
    const char* pszZqdm,  
    char* pszResult,  
    char* pszErrInfo);
```

### 功能:

获取指定商品的盘口五档报价

### 参数:

- |            |   |
|------------|---|
| nConnID    | - 行情连接代码  |
| nMarket    | - 市场代码  |
| pszZqdm    | - 商品代码, pnCount 个证券代码组成的数组  |
| pszResult  | - 此 API 执行返回后, Result 内保存了返回的证券代码信息, 形式为表格数据, 行数据之间通过\n 字符分割, 列数据之间通过\t 分隔。<br>一般要分配 1024*1024 字节的空间。出错时空字符串。 |
| pszErrInfo | - 此 API 执行返回后, 如果出错, 保存了错误信息说明。一般要分配 256 字节的空间。没  |

出错时空字符串。

### 返回值:

成功返回 true, 失败返回 false

---

## 6.1.7. TdxExHq\_GetInstrumentBars

```
bool WINAPI TdxExHq_GetInstrumentBars(  
    int nConnID,  
    char nCategory,  
    char nMarket,  
    const char* pszZqdm,  
    int nStart,  
    short* pnCount,  
    char* pszResult,  
    char* pszErrInfo);
```

### 功能:

获取指定商品的 K 线数据

### 参数:

- |         |          |
|---------|----------|
| nConnID | - 行情连接代码 |
|---------|----------|

nCategory	- K 线种类
	0 5 分钟 K 线
	1 15 分钟 K 线
	2 30 分钟 K 线
	3 1 小时 K 线
	4 日 K 线
	5 周 K 线
	6 月 K 线
	7 1 分钟
	8 1 分钟 K 线
	9 日 K 线
	10 季 K 线
	11 年 K 线
nMarket	- 市场代码
pszZqdm	- 商品代码
nStart	- 范围开始位置, 最后一条 K 线位置是 0, 前一条是 1, 依此类推
pnCount	- 范围大小, API 执行前, 表示用户要请求的 K 线数目, API 执行后, 保存了实际返回的 K 线数目, 最大值 800;
pszResult	- 此 API 执行返回后, Result 内保存了返回的查询数据, 形式为表格数据; 行数据之间通过\n 字符分割, 列数据之间通过\t 分隔; 一般要分配 1024*1024 字节的空间。出错时空字符串。
pszErrInfo	- 此 API 执行返回后, 如果出错, 保存了错误信息说明。一般要分配 256 字节的空间。没出错时空字符串。

返回值:

成功返回 true, 失败返回 false

### 6.1.8. TdxExHq\_GetMinuteTimeData

```
bool WINAPI TdxExHq_GetMinuteTimeData(
    int nConnID,
    char nMarket,
    const char* pszZqdm,
    char* pszResult,
    char* pszErrInfo);
```

功能:

获取指定商品的分时行情数据

参数:

nConnID	- 行情连接代码
nMarket	- 市场代码
pszZqdm	- 证券代码

- pszResult        - 此 API 执行返回后, Result 内保存了返回的查询数据, 形式为表格数据;  
                  行数据之间通过\n 字符分割, 列数据之间通过\t 分隔;  
                  一般要分配 1024\*1024 字节的空间。出错时空字符串。
- pszErrInfo       - 此 API 执行返回后, 如果出错, 保存了错误信息说明。  
                  一般要分配 256 字节的空间。没出错时空字符串。

返回值:

成功返回 true, 失败返回 false

---

### 6.1.9. TdxExHq\_GetHistoryMinuteTimeData

```
bool WINAPI TdxExHq_GetHistoryMinuteTimeData(  
    int nConnID,  
    char nMarket,  
    const char* pszZqdm,  
    int nDate,  
    char* pszResult,  
    char* pszErrInfo);
```

功能:

获取指定商品的历史分时行情数据

参数:

- nConnID        - 行情连接代码
- nMarket        - 市场代码
- pszZqdm        - 证券代码
- nDate          - 日期, 比如 2017 年 2 月 1 日为整数 20170201
- pszResult      - 此 API 执行返回后, Result 内保存了返回的查询数据, 形式为表格数据;  
                  行数据之间通过\n 字符分割, 列数据之间通过\t 分隔;  
                  一般要分配 1024\*1024 字节的空间。出错时空字符串。
- pszErrInfo     - 此 API 执行返回后, 如果出错, 保存了错误信息说明。  
                  一般要分配 256 字节的空间。没出错时空字符串。

返回值:

成功返回 true, 失败返回 false

---

### 6.1.10. TdxExHq\_GetTransactionData

```
bool WINAPI TdxExHq_GetTransactionData(  
    int nConnID,
```

```
char nMarket,  
char* pszZqdm,  
int nStart,  
short* pnCount,  
char* pszResult,  
char* pszErrInfo);
```

**功能:**

获取指定商品在某个范围的分笔成交数据

**参数:**

nConnID	- 行情连接代码
nMarket	- 市场代码
pszZqdm	- 证券代码
nStart	- 范围开始位置, 最后一条 K 线位置是 0, 前一条是 1, 依此类推
pnCount	- 范围大小, API 执行前, 表示用户要请求的 K 线数目, API 执行后, 保存了实际返回的 K 线

**数目:**

pszResult	- 此 API 执行返回后, Result 内保存了返回的查询数据, 形式为表格数据; 行数据之间通过\n 字符分割, 列数据之间通过\t 分隔; 一般要分配 1024*1024 字节的空间。出错时空字符串。
pszErrInfo	- 此 API 执行返回后, 如果出错, 保存了错误信息说明。 一般要分配 256 字节的空间。没出错时空字符串。

**返回值:**

成功返回 true, 失败返回 false

---

## 6.1.11. TdxExHq\_GetHistoryTransactionData

```
bool WINAPI TdxExHq_GetHistoryTransactionData(  
    int nConnID,  
    char nMarket,  
    const char* pszZqdm,  
    int nDate,  
    int nStart,  
    short* pnCount,  
    char* pszResult,  
    char* pszErrInfo);
```

**功能:**

获取指定商品在某个范围的历史分笔成交数据

券商行情服务器侧限制最大允许单次返回 2000 条记录, 如果需要更多数据则需要多次调用循环读取。

**参数:**

nConnID	- 行情连接代码
nMarket	- 市场代码
pszZqdm	- 证券代码
nStart	- 范围开始位置, 最后一条 K 线位置是 0, 前一条是 1, 依此类推
pnCount	- 范围大小, API 执行前, 表示用户要请求的 K 线数目, API 执行后, 保存了实际返回的 K 线数目;
date	- 日期, 比如 2017 年 2 月 1 日为整数 20170201
pszResult	- 此 API 执行返回后, Result 内保存了返回的查询数据, 形式为表格数据: 行数据之间通过\n 字符分割, 列数据之间通过\t 分隔; 一般要分配 1024*1024 字节的空间。出错时空字符串。
pszErrInfo	- 此 API 执行返回后, 如果出错, 保存了错误信息说明。 一般要分配 256 字节的空间。没出错时空字符串。

返回值:

成功返回 true, 失败返回 false

## 7. 常见问题

### 7.1.1. 常见问题

#### 1. C#返回值