策略王COM元件使用說明

-國內報價-

# 函式範例

## SKQuoteLib (國內報價)

*函式*

1. 連線

|  |  |  |
| --- | --- | --- |
| **功能** | **函式名稱** | **備註** |
| 1. **報價連線** | [SKQuoteLib\_EnterMonitorLONG](#_SKQuoteLib_EnterMonitorLONG) | LONG index 對應 |
| 1. **報價斷線** | [SKQuoteLib\_LeaveMonitor](#_SKQuoteLib_LeaveMonitor) |  |
| 1. **取得報價連線狀態** | [SKQuoteLib\_IsConnected](#_SKQuoteLib_IsConnected) |  |
| 1. **取得連線狀態** | [SKQuoteLib\_GetQuoteStatus](#_SKQuoteLib_GetQuoteStatus) | (僅限連線成功後使用) |
| 1. **主機時間查詢** | [SKQuoteLib\_RequestServerTime](#_SKQuoteLib_RequestServerTime) |  |

1. 大盤資訊

|  |  |  |
| --- | --- | --- |
| **功能** | **函式名稱** | **備註** |
| **取得大盤買賣** | [SKQuoteLib\_GetMarketBuySellUpDown](#_SKQuoteLib_GetMarketBuySellUpDown) | 不支援盤中零股。 |

1. 商品清單

|  |  |  |
| --- | --- | --- |
| **功能** | **函式名稱** | **備註** |
| **國內商品清單查詢** | [SKQuoteLib\_RequestStockList](#_SKQuoteLib_RequestStockList) | 盤中零股上市5、盤中零股上櫃 6  客製化期貨-9、客製化選擇權-10 |

1. 個股資訊

|  |  |  |
| --- | --- | --- |
| **功能** | **函式名稱** | **備註** |
| **取得商品物件** | [SKQuoteLib\_GetStockByNoLONG](#_SKQuoteLib_GetStockByNoLONG) | LONG index 對應  不適用盤中零股 |
| **取得商品物件** | [SKQuoteLib\_GetStockByMarketAndNo](#_SKQuoteLib_GetStockByMarketAndNo) | 適用盤中零股 |

1. 即時報價

|  |  |  |
| --- | --- | --- |
| **功能** | **函式名稱** | **備註** |
| 1. **訂閱報價** | [SKQuoteLib\_RequestStocks](#_SKQuoteLib_RequestStocks) | 不支援盤中零股。 |
| 1. **取得商品物件** | [SKQuoteLib\_GetStockByIndexLONG](#_SKQuoteLib_GetStockByIndexLONG) | LONG index 對應 |
| 1. **訂閱指定市場別及指定商品即時報價** | [SKQuoteLib\_RequestStocksWithMarketNo](#_SKQuoteLib_RequestStocksWithMarketN) | 適用盤中零股 |
| 1. **取消訂閱報價** | [SKQuoteLib\_CancelRequestStocks](#_SKQuoteLib_CancelRequestStocks) | 不支援盤中零股。 |
| 1. **取得證券市價特殊值** | [SKQuoteLib\_GetMarketPriceTS](#_SKQuoteLib_GetMarketPriceTS) |  |

1. 五檔&成交明細

|  |  |  |
| --- | --- | --- |
| **功能** | **函式名稱** | **備註** |
| 1. **訂閱Ticks及五檔** | [SKQuoteLib\_RequestTicks](#_SKQuoteLib_RequestTicks) | 不支援盤中零股。  [OnNotifyTicksLONG](#_OnNotifyTicksLONG_1)、[OnNotifyHistoryTicksLONG](#_OnNotifyHistoryTicksLONG)、[OnNotifyBest5LONG](#_OnNotifyBest5LONG) |
| 1. **取得Tick** | [SKQuoteLib\_GetTickLONG](#_SKQuoteLib_GetTickLONG) | LONG index 對應 |
| 1. **取得五檔** | [SKQuoteLib\_GetBest5LONG](#_SKQuoteLib_GetBest5LONG) | LONG index 對應 |
| 1. **取得即時成交明細Ticks&Best5** | [SKQuoteLib\_RequestTicksWithMarketNo](#_SKQuoteLib_RequestTicksWithMarketNo) | 適用盤中零股 |
| 1. **訂閱即時成交明細** | [SKQuoteLib\_RequestLiveTick](#_SKQuoteLib_RequestLiveTick) | 包含訂閱，僅即時成交明細 |
| 1. **取消訂閱Tick及五檔** | [SKQuoteLib\_CancelRequestTicks](#_SKQuoteLib_CancelRequestTicks) | 不支援盤中零股。 |

1. 技術分析

|  |  |  |
| --- | --- | --- |
| **功能** | **函式名稱** | **備註** |
| **MACD** | |  | | --- | | 訂閱[SKQuoteLib\_RequestMACD](#_SKQuoteLib_RequestMACD) | | 取得[SKQuoteLib\_GetMACDLONG](#_SKQuoteLib_GetMACDLONG) | | 目前只提供證券市場，不含盤中零股  LONG index 對應  不支援盤中零股。 |
| **布林通道** | |  | | --- | | 訂閱[SKQuoteLib\_RequestBoolTunel](#_SKQuoteLib_RequestBoolTunel) | | 取得[SKQuoteLib\_GetBoolTunelLONG](#_SKQuoteLib_GetBoolTunelLONG) | | 目前只提供證券市場，不含盤中零股  LONG index 對應  不支援盤中零股。 |
| **K線查詢** | |  | | --- | | **指定區間**[SKQuoteLib\_RequestKLineAMByDate](#_SKQuoteLib_RequestKLineAMByDate) | | (新) [SKQuoteLib\_RequestKLineAM](#_SKQuoteLib_RequestKLineAM) | | [SKQuoteLib\_RequestKLine](#_SKQuoteLib_RequestKLine) | | 不支援盤中零股。  不支援盤中零股。  國內期權可選盤別 |

1. 風險計算(選擇權)

|  |  |  |
| --- | --- | --- |
| **功能** | **函式名稱** | **備註** |
| **風險參數Delta** | [SKQuoteLib\_Delta](#_SKQuoteLib_Delta) |  |
| **風險參數Gamma** | SKQuoteLib\_[Gamma](#_SKQuoteLib_Gamma) |  |
| **風險參數Vega** | [SKQuoteLib\_Vega](#_SKQuoteLib_Vega) |  |
| **風險參數Theta** | [SKQuoteLib\_Theta](#_SKQuoteLib_Theta) |  |
| **風險參數Rho** | [SKQuoteLib\_Rho](#_SKQuoteLib_Rho) |  |

1. 交易資訊(期選)

|  |  |  |
| --- | --- | --- |
| **功能** | **函式名稱** | **備註** |
| **訂閱期貨商品資訊** | [SKQuoteLib\_RequestFutureTradeInfo](#_SKQuoteLib_RequestFutureTradeInfo) |  |
| **選擇權資訊** | [SKQuoteLib\_GetStrikePrices](#_SKQuoteLib_GetStrikePrices) |  |

*事件*

1. 連線

|  |  |  |
| --- | --- | --- |
| **功能** | **事件名稱** | **備註** |
| **連線狀態** | [OnConnection](#_OnConnection) | 避免在此處直接進行RequestStocks and RequestTicks 等報價訂閱 |
| **主機時間** | [OnNotifyServerTime](#_OnNotifyServerTime) |  |

1. 大盤資訊

|  |  |  |
| --- | --- | --- |
| **功能** | **事件名稱** | **備註** |
| **大盤張筆** | [OnNotifyMarketTot](#_OnNotifyMarketTot) |  |
| **大盤買賣** | [OnNotifyMarketBuySell](#_OnNotifyMarketBuySell) |  |
| **大盤漲跌家數** | |  | | --- | | [[OnNotifyMarketHighLow](#_4-4-j_OnNotifyMarketHighLow)NoWarrant](#_OnNotifyMarketHighLowNoWarrant) | | [OnNotifyMarketHighLow](#_OnNotifyMarketHighLow) | | (同時回傳含權證及不含權證)  (包含權證) |

1. 商品清單

|  |  |  |
| --- | --- | --- |
| **功能** | **事件名稱** | **備註** |
| **國內市場商品清單(含類別代碼)** | |  | | --- | | [OnNotifyCommodityListWithTypeNo](#_OnNotifyCommodityListWithTypeNo) | | [OnNotifyStockList](#_OnNotifyStockList) | | (上市、上櫃、期貨、選擇權、  興櫃) |

1. 即時報價

|  |  |  |
| --- | --- | --- |
| **功能** | **事件名稱** | **備註** |
| **報價更新通知** | [OnNotifyQuoteLONG](#_OnNotifyQuoteLONG) | LONG index 對應 |
| **整零價差即時行情** | [OnNotifyOddLotSpreadDeal](#_OnNotifyOddLotSpreadDeal) |  |

1. 五檔&成交明細

|  |  |  |
| --- | --- | --- |
| **功能** | **事件名稱** | **備註** |
| **即時五檔** | [OnNotifyBest5LONG](#_OnNotifyBest5LONG) | LONG index 對應 |
| **即時Tick** | [OnNotifyTicksLONG](#_OnNotifyTicksLONG) | LONG index 對應 |
| **回補Tick** | [OnNotifyHistoryTicksLONG](#_OnNotifyHistoryTicksLONG) | LONG index 對應 |

1. 技術分析

|  |  |  |
| --- | --- | --- |
| **功能** | **事件名稱** | **備註** |
| **MACD** | [OnNotifyMACDLONG](#_OnNotifyMACDLONG) | LONG index 對應 |
| **布林通道** | [OnNotifyBoolTunelLONG](#_OnNotifyBoolTunelLONG) | LONG index 對應 |
| **K線資料** | [OnNotifyKLineData](#_OnNotifyKLineData) |  |

1. 交易資訊(期選)

|  |  |  |
| --- | --- | --- |
| **功能** | **事件名稱** | **備註** |
| **期貨商品交易資訊** | [OnNotifyFutureTradeInfoLONG](#_OnNotifyFutureTradeInfoLONG) | LONG index 對應 |
| **選擇權資訊** | [OnNotifyStrikePrices](#_OnNotifyStrikePrices) | 選擇權 |

### 連線

#### SKQuoteLib\_EnterMonitorLONG

|  |  |  |
| --- | --- | --- |
| (LONG index)與報價伺服器建立連線。（含盤中零股市場商品） | | |
| **宣告** | Long SKQuoteLib\_EnterMonitorLONG(); | |
| **參數** | 無 |  |
| **回傳值** | 0表示成功，其餘非0數值都表示失敗。錯誤代碼可參考對照表。 | |
| **備註** | 連線狀態由 [OnConnection](#_OnConnection) 事件回傳。  需先簽署證券或期貨API下單聲明書，方可使用 | |

|  |
| --- |
| C#範例程式碼\_QuoteForm.cs |
| UI示意圖 |
| |  |  | | --- | --- | |  | 拉出 1個Button，命名(Name)為「buttonSKQuoteLib\_EnterMonitorLONG」。 | |
| 全域變數 |
| // 宣告物件  SKCenterLib m\_pSKCenter = new SKCenterLib(); // 登入&環境設定物件  SKQuoteLib m\_pSKQuote = new SKQuoteLib(); // 國內報價物件 |
| Current Block (可放在buttonSKQuoteLib\_EnterMonitorLONG\_Click的Click事件裡) |
| // 與報價伺服器連線  int nCode = m\_pSKQuote.SKQuoteLib\_EnterMonitorLONG();  // 取得回傳訊息  string msg = "【SKQuoteLib\_EnterMonitorLONG】" + m\_pSKCenter.SKCenterLib\_GetReturnCodeMessage(nCode);  richTextBoxMethodMessage.AppendText(msg + "\n"); |
| 執行成功畫面 |
|  |

#### SKQuoteLib\_LeaveMonitor

|  |  |  |
| --- | --- | --- |
| 中斷所有Solace伺服器連線。 | | |
| **宣告** | Long SKQuoteLib\_LeaveMonitor(); | |
| **參數** | 無 |  |
| **回傳值** | 0表示成功，其餘非0數值都表示失敗。錯誤代碼可參考對照表。 | |
| **備註** | 會中斷所有Solace連線，包含報價與回報，但不會中斷模擬平台回報連線與公告  ＊斷線後，重新連線即還原設定：  限制:『單一SKQuoteLib物件，僅一個訂閱Tick報價功能(10檔商品)及一個RequestStocks訂閱即時報價功能(100檔)』。 | |

|  |
| --- |
| C#範例程式碼\_QuoteForm.cs |
| UI示意圖 |
| |  |  | | --- | --- | |  | 拉出 1個Button，命名(Name)為「buttonSKQuoteLib\_LeaveMonitor」。 | |
| 全域變數 |
| // 宣告物件  SKCenterLib m\_pSKCenter = new SKCenterLib(); // 登入&環境設定物件  SKQuoteLib m\_pSKQuote = new SKQuoteLib(); // 國內報價物件 |
| Current Block (可放在buttonSKQuoteLib\_LeaveMonitor\_Click的Click事件裡) |
| // 中斷所有Solace伺服器連線  int nCode = m\_pSKQuote.SKQuoteLib\_LeaveMonitor();  // 取得回傳訊息  string msg = "【SKQuoteLib\_LeaveMonitor】" + m\_pSKCenter.SKCenterLib\_GetReturnCodeMessage(nCode);  richTextBoxMethodMessage.AppendText(msg + "\n"); |
| 執行成功畫面 |
|  |

#### SKQuoteLib\_IsConnected

|  |  |
| --- | --- |
| 檢查目前報價的連線狀態。 | |
| **宣告** | Long SKQuoteLib\_IsConnected(); |
| **參數** | 無 |
| **回傳值** | 0表示斷線。1表示連線中。2表示下載中。  錯誤代碼可參考對照表。 |
| **備註** | 請同時接收通知事件[OnConnection](#_OnConnection) |

|  |
| --- |
| C#範例程式碼\_QuoteForm.cs |
| UI示意圖 |
| |  |  | | --- | --- | |  | 拉出 1個Button，命名(Name)為「buttonSKQuoteLib\_IsConnected」。 | |
| 全域變數 |
| // 宣告物件  SKCenterLib m\_pSKCenter = new SKCenterLib(); // 登入&環境設定物件  SKQuoteLib m\_pSKQuote = new SKQuoteLib(); // 國內報價物件 |

|  |
| --- |
| Current Block (可放在buttonSKQuoteLib\_LeaveMonitor\_Click的Click事件裡) |
| string msg;  // 檢查目前報價的連線狀態  int nCode = m\_pSKQuote.SKQuoteLib\_IsConnected();  // 取得回傳訊息  switch (nCode)  {  case 0:  msg = "斷線";  break;  case 1:  msg = "連線中";  break;  case 2:  msg = "下載中";  break;  default:  msg = "出錯啦";  break;  }  msg = "【SKQuoteLib\_IsConnected】" + msg;  richTextBoxMessage.AppendText(msg + "\n"); |
| 執行成功畫面 |
|  |

#### SKQuoteLib\_GetQuoteStatus

|  |  |  |
| --- | --- | --- |
| 查詢報價連線狀態(是否超過報價連線限制,連線數資訊)。 | | |
| **宣告** | Long SKQuoteLib\_GetQuoteStatus([in,out] LONG\* pnConnectionCount,  [in,out] VARIANT\_BOOL\* pbIsOutLimit); | |
| **參數** | pnConnectionCount | 連線數:   * 當pbIsOutLimit**連線數超過限制**為true , 連線數為目前限制最大可使用連線數 * 當pbIsOutLimit**連線數超過限制**為false, 連線數為先前已使用連線數(不含當次新連線)   當pnConnectionCount 帶入任一數值，函式庫會回傳連線數，並回傳給呼叫端。 |
| pbIsOutLimit | 報價連線數是超過限制  當pbIsOutLimit帶入false，函式庫會回傳是否超過連線數布林值，並回傳給呼叫端。 |
| **回傳值** | 0表示成功，其餘非0數值都表示失敗。錯誤代碼可參考對照表。 | |
| **備註** | 限當次報價連線成功後，查詢用  一張含有 文字, 螢幕擷取畫面, 數字, 字型 的圖片  自動產生的描述  ＊EX：若最大連線數為2, 且目前連線超過限制,則回傳:2,True. | |

|  |
| --- |
| C#範例程式碼\_QuoteForm.cs |
| UI示意圖 |
| |  |  | | --- | --- | |  | 拉出 1個Button，命名(Name)為「buttonSKQuoteLib\_GetQuoteStatus」。 | |
| 全域變數 |
| // 宣告物件  SKCenterLib m\_pSKCenter = new SKCenterLib(); // 登入&環境設定物件  SKQuoteLib m\_pSKQuote = new SKQuoteLib(); // 國內報價物件 |
| Current Block (可放在buttonSKQuoteLib\_GetQuoteStatus\_Click的Click事件裡) |
| int pnConnectionCount = 0; //當pnConnectionCount 帶入任一數值，函式庫會回傳連線數，並回傳給呼叫端  bool pbIsOutLimit = false; //當pbIsOutLimit 帶入false，函式庫會回傳是否超過連線數布林值，並回傳給呼叫端  // 查詢報價連線狀態(是否超過報價連線限制,連線數資訊)  int nCode = m\_pSKQuote.SKQuoteLib\_GetQuoteStatus(ref pnConnectionCount, ref pbIsOutLimit);  // 取得回傳訊息  // 當pbIsOutLimit連線數超過限制為false, 連線數為先前已使用連線數(不含當次新連線)  string msg = "【SKQuoteLib\_GetQuoteStatus】" + "連線數:" + pnConnectionCount + "超過限制:" + pbIsOutLimit;  richTextBoxMessage.AppendText(msg + "\n"); |
| 執行成功畫面 |
|  |

#### SKQuoteLib\_RequestServerTime

|  |  |  |
| --- | --- | --- |
| 要求報價主機傳送目前時間。  目前solace固定每十秒會自動更新時間 | | |
| **宣告** | Long SKQuoteLib\_RequestServerTime(); | |
| **參數** | 無 |  |
| **回傳值** | 0表示成功，其餘非0數值都表示失敗。錯誤代碼可參考對照表。 | |
| **相關通知事件** | [OnNotifyServerTime](#_OnNotifyServerTime) | |
| **備註** |  | |

|  |
| --- |
| C#範例程式碼\_QuoteForm.cs |
| UI示意圖 |
| |  |  | | --- | --- | |  | 拉出 1個Button，命名(Name)為「buttonSKQuoteLib\_RequestServerTime」。 | |
| 全域變數 |
| // 宣告物件  SKCenterLib m\_pSKCenter = new SKCenterLib(); // 登入&環境設定物件  SKQuoteLib m\_pSKQuote = new SKQuoteLib(); // 國內報價物件 |
| Current Block (可放在buttonSKQuoteLib\_RequestServerTime\_Click的Click事件裡) |
| // 要求報價主機傳送目前時間。  int nCode = m\_pSKQuote.SKQuoteLib\_RequestServerTime();  // 取得回傳訊息  string msg = "【SKQuoteLib\_RequestServerTime】" + m\_pSKCenter.SKCenterLib\_GetReturnCodeMessage(nCode);  richTextBoxMethodMessage.AppendText(msg + "\n"); |
| 執行成功畫面 |
|  |

### 大盤資訊

#### SKQuoteLib\_GetMarketBuySellUpDown

|  |  |  |
| --- | --- | --- |
| 要求傳送上市與上櫃大盤資訊(成交數,買賣數,漲跌家數)。 | | |
| **宣告** | Long SKQuoteLib\_GetMarketBuySellUpDown(); | |
| **參數** | 無 |  |
| **回傳值** | 0表示成功，其餘非0數值都表示失敗。錯誤代碼可參考對照表。 | |
| **相關通知事件** | 1. **大盤張筆**[OnNotifyMarketTot](#_OnNotifyMarketTot) 2. **大盤買賣**[OnNotifyMarketBuySell](#_OnNotifyMarketBuySell) 3. **大盤漲跌家數**[OnNotifyMarketHighLow](#_OnNotifyMarketHighLow) 4. **大盤漲跌家數(『含權證家數』、 『不含權證家數』)**[OnNotifyMarketHighLow](#_OnNotifyMarketHighLowNoWarrant)NoWarrant | |
| **備註** | 不支援盤中零股。 | |

|  |
| --- |
| C#範例程式碼\_QuoteForm.cs |
| UI示意圖 |
| |  |  | | --- | --- | |  | 拉出 1個Button，命名(Name)為「buttonSKQuoteLib\_GetMarketBuySellUpDown」。 | |
| 全域變數 |
| // 宣告物件  SKCenterLib m\_pSKCenter = new SKCenterLib(); // 登入&環境設定物件  SKQuoteLib m\_pSKQuote = new SKQuoteLib(); // 國內報價物件 |
| Current Block (可放在buttonSKQuoteLib\_GetMarketBuySellUpDown\_Click的Click事件裡) |
| // 要求傳送上市與上櫃大盤資訊(成交數,買賣數,漲跌家數)  int nCode = m\_pSKQuote.SKQuoteLib\_GetMarketBuySellUpDown();  // 取得回傳訊息  string msg = "【SKQuoteLib\_GetMarketBuySellUpDown】" + m\_pSKCenter.SKCenterLib\_GetReturnCodeMessage(nCode);  richTextBoxMethodMessage.AppendText(msg + "\n"); |
| 執行成功畫面 |
|  |

### 商品清單

#### SKQuoteLib\_RequestStockList

|  |  |  |
| --- | --- | --- |
| 根據市場別編號，取得國內各市場代碼所包含的商品基本資料相關資訊。 | | |
| **宣告** | Long SKQuoteLib\_RequestStockList([in] SHORT sMarketNo); | |
| **參數** | sMarketNo | 市場別代碼。 |
| **回傳值** | 0表示成功，其餘非0數值都表示失敗。錯誤代碼可參考對照表。 | |
| **相關通知事件** | [OnNotifyCommodityListWithTypeNo](#_OnNotifyCommodityListWithTypeNo)  [OnNotifyStockList](#_OnNotifyStockList) | |
| **備註** | ＊請先[SKQuoteLib\_EnterMonitorLONG](#_SKQuoteLib_EnterMonitorLONG)，須等[OnConnection](#_OnConnection)收到SK\_SUBJECT\_CONNECTION\_STOCKS\_READY後，再進行商品檔查詢。  市場編號分別為：  上市 0、上櫃 1、期貨 2、選擇權 3、興櫃 4、  盤中零股-上市5、盤中零股-上櫃6  客製化期貨-9、客製化選擇權-10  若未開立證券帳戶或未簽署證券API下單同意書，將無法查詢  ”上市 0、上櫃 1、興櫃 4”、盤中零股-上市5、盤中零股-上櫃6。（錯誤代碼3031）   * 若未開立期貨帳戶或未簽署期貨API下單同意書，將無法查詢”期貨 2、選擇權 3”。（錯誤代碼3031） | |

|  |
| --- |
| C#範例程式碼\_QuoteForm.cs |
| UI示意圖 |
| |  |  | | --- | --- | |  | 拉出 1個ComboBox，命名(Name)為「comboBoxSKQuoteLib\_RequestStockList」。  拉出 1個Button，命名(Name)為「buttonSKQuoteLib\_RequestStockList」。 | |
| 全域變數 |
| // 宣告物件  SKCenterLib m\_pSKCenter = new SKCenterLib(); // 登入&環境設定物件  SKQuoteLib m\_pSKQuote = new SKQuoteLib(); // 國內報價物件 |
| Constructor |
| //comboBoxSKQuoteLib\_RequestStockList  {  comboBoxSKQuoteLib\_RequestStockList.Items.Add("上市");  comboBoxSKQuoteLib\_RequestStockList.Items.Add("上櫃");  comboBoxSKQuoteLib\_RequestStockList.Items.Add("期貨");  comboBoxSKQuoteLib\_RequestStockList.Items.Add("選擇權");  comboBoxSKQuoteLib\_RequestStockList.Items.Add("興櫃");  comboBoxSKQuoteLib\_RequestStockList.Items.Add("盤中零股-上市");  comboBoxSKQuoteLib\_RequestStockList.Items.Add("盤中零股-上櫃");  comboBoxSKQuoteLib\_RequestStockList.Items.Add("客製化期貨");  comboBoxSKQuoteLib\_RequestStockList.Items.Add("客製化選擇權");  } |

|  |
| --- |
| Current Block (可放在buttonSKQuoteLib\_RequestStockList\_Click的Click事件裡) |
| dataGridViewOnNotifyStockList.Rows.Clear();  short sMarketNo = 0;  string selectValue = comboBoxSKQuoteLib\_RequestStockList.Text;  if (selectValue == "上市") sMarketNo = 0;  else if (selectValue == "上櫃") sMarketNo = 1;  else if (selectValue == "期貨") sMarketNo = 2;  else if (selectValue == "選擇權") sMarketNo = 3;  else if (selectValue == "興櫃") sMarketNo = 4;  else if (selectValue == "盤中零股-上市") sMarketNo = 5;  else if (selectValue == "盤中零股-上櫃") sMarketNo = 6;  else if (selectValue == "客製化期貨") sMarketNo = 9;  else if (selectValue == "客製化選擇權") sMarketNo = 10;  // 根據市場別編號，取得國內各市場代碼所包含的商品基本資料相關資訊  int nCode = m\_pSKQuote.SKQuoteLib\_RequestStockList(sMarketNo);  // 取得回傳訊息  string msg = "【SKQuoteLib\_RequestStockList】" + m\_pSKCenter.SKCenterLib\_GetReturnCodeMessage(nCode);  richTextBoxMethodMessage.AppendText(msg + "\n"); |
| 執行成功畫面 |
|  |

### 個股資訊

#### SKQuoteLib\_GetStockByNoLONG

|  |  |  |
| --- | --- | --- |
| (LONG index)根據商品代號，取回商品報價的相關資訊。 | | |
| **宣告** | Long SKQuoteLib\_GetStockByNoLONG ([in] BSTR bstrStockNo, [in,out] struct **SKSTOCKLONG\*** pSKStock); | |
| **參數** | bstrStockNo | 商品代號，例如 6005。 |
| pSKStock | SKCOM元件中的**[SKSTOCKLONG](#_SKSTOCKLONG_(國內報價商品物件_LONG)**物件，將物件帶入此欄位中。 |
| **回傳值** | 0表示成功，其餘非0數值都表示失敗。錯誤代碼可參考對照表。 | |
| **備註** | 僅適用現股，不適用盤中零股-上市及盤中零股-上櫃  ＊須等收到SK\_SUBJECT\_CONNECTION\_STOCKS\_READY後，方可進行。  ＊須使用[SKQuoteLib\_EnterMonitorLONG](#_SKQuoteLib_EnterMonitorLONG)登入，方可執行此函式。  未訂閱即時報價，EX:只先訂成交明細SKQuoteLib\_ReqeustTicks時，執行此功能，僅可取得商品基本資料(商品名稱、昨收價，非即時性質欄位) | |

|  |
| --- |
| C#範例程式碼\_QuoteForm.cs |
| UI示意圖 |
| |  |  | | --- | --- | |  | 拉出 1個TextBox，命名(Name)為「textBoxSKQuoteLib\_GetStockByNoLONG」。  拉出 1個DataGridView，命名(Name)為「dataGridViewSKQuoteLib\_GetStockByNoLONG」。  拉出 1個Button，命名(Name)為「buttonSKQuoteLib\_GetStockByNoLONG」。 | |
| 全域變數 |
| // 宣告物件  SKCenterLib m\_pSKCenter = new SKCenterLib(); // 登入&環境設定物件  SKQuoteLib m\_pSKQuote = new SKQuoteLib(); // 國內報價物件 |

|  |
| --- |
| Constructor |
| // dataGridViewSKQuoteLib\_GetStockByNoLONG  {  dataGridViewSKQuoteLib\_GetStockByNoLONG.Columns.Add("Column1", "類股別");  dataGridViewSKQuoteLib\_GetStockByNoLONG.Columns.Add("Column2", "市埸代碼");  dataGridViewSKQuoteLib\_GetStockByNoLONG.Columns.Add("Column3", "商品代碼");  dataGridViewSKQuoteLib\_GetStockByNoLONG.Columns.Add("Column4", "商品名稱");  dataGridViewSKQuoteLib\_GetStockByNoLONG.Columns.Add("Column5", "最高價");  dataGridViewSKQuoteLib\_GetStockByNoLONG.Columns.Add("Column6", "開盤價");  dataGridViewSKQuoteLib\_GetStockByNoLONG.Columns.Add("Column7", "最低價");  dataGridViewSKQuoteLib\_GetStockByNoLONG.Columns.Add("Column8", "成交價");  dataGridViewSKQuoteLib\_GetStockByNoLONG.Columns.Add("Column9", "單量");  dataGridViewSKQuoteLib\_GetStockByNoLONG.Columns.Add("Column10", "昨收、參考價");  dataGridViewSKQuoteLib\_GetStockByNoLONG.Columns.Add("Column11", "買價");  dataGridViewSKQuoteLib\_GetStockByNoLONG.Columns.Add("Column12", "買量");  dataGridViewSKQuoteLib\_GetStockByNoLONG.Columns.Add("Column13", "賣價");  dataGridViewSKQuoteLib\_GetStockByNoLONG.Columns.Add("Column14", "賣量");  dataGridViewSKQuoteLib\_GetStockByNoLONG.Columns.Add("Column15", "買盤量(即外盤量)");  dataGridViewSKQuoteLib\_GetStockByNoLONG.Columns.Add("Column16", "賣盤量(即內盤量)");  dataGridViewSKQuoteLib\_GetStockByNoLONG.Columns.Add("Column17", "總量");  dataGridViewSKQuoteLib\_GetStockByNoLONG.Columns.Add("Column18", "昨量");  dataGridViewSKQuoteLib\_GetStockByNoLONG.Columns.Add("Column19", "漲停價");  dataGridViewSKQuoteLib\_GetStockByNoLONG.Columns.Add("Column20", "跌停價");  dataGridViewSKQuoteLib\_GetStockByNoLONG.Columns.Add("Column21", "揭示 0:一般 1:試算(試撮)");  dataGridViewSKQuoteLib\_GetStockByNoLONG.Columns.Add("Column22", "[限證券整股商品]可否當沖 0:一般 1:可先買後賣現股當沖2:可先買後賣和先賣後買現股當沖");  dataGridViewSKQuoteLib\_GetStockByNoLONG.Columns.Add("Column23", "交易日(YYYYMMDD)");  dataGridViewSKQuoteLib\_GetStockByNoLONG.Columns.Add("Column24", "[證券] 整股、盤中零股揭示註記 0:現股 ; 1:盤中零股");  } |

|  |
| --- |
| Current Block (可放在buttonSKQuoteLib\_GetStockByNoLONG\_Click的Click事件裡) |
| string bstrStockNo = textBoxSKQuoteLib\_GetStockByNoLONG.Text;  SKSTOCKLONG pSKStock = new SKSTOCKLONG();  // (LONG index)根據商品代號，取回商品報價的相關資訊  int nCode = m\_pSKQuote.SKQuoteLib\_GetStockByNoLONG(bstrStockNo, ref pSKStock);  // 取得回傳訊息  string msg = "【SKQuoteLib\_GetStockByNoLONG】" + m\_pSKCenter.SKCenterLib\_GetReturnCodeMessage(nCode);  richTextBoxMethodMessage.AppendText(msg + "\n");  // SKSTOCKLONG  bool found = false; // 找到就不新增資料  {  foreach (DataGridViewRow row in dataGridViewSKQuoteLib\_GetStockByNoLONG.Rows)  {  if (row.IsNewRow)  continue;  if (row.Cells[2].Value != null)  {  if (bstrStockNo == row.Cells[2].Value.ToString())// 搜尋過的商品代號  {  found = true;  break;  }  }    }  if (found == false)  dataGridViewSKQuoteLib\_GetStockByNoLONG.Rows.Add(pSKStock.sTypeNo, pSKStock.bstrMarketNo, pSKStock.bstrStockNo, pSKStock.bstrStockName, pSKStock.nHigh / 100.0, pSKStock.nOpen / 100.0, pSKStock.nLow / 100.0, pSKStock.nClose / 100.0, pSKStock.nTickQty, pSKStock.nRef / 100.0, pSKStock.nBid / 100.0, pSKStock.nBc, pSKStock.nAsk / 100.0, pSKStock.nAc, pSKStock.nTBc, pSKStock.nTAc, pSKStock.nTQty, pSKStock.nYQty, pSKStock.nUp / 100.0, pSKStock.nDown / 100.0, pSKStock.nSimulate, pSKStock.nDayTrade, pSKStock.nTradingDay, pSKStock.nTradingLotFlag);  } |
| 執行成功畫面 |
|  |

##### SKQuoteLib\_GetStockByMarketAndNo

|  |  |  |
| --- | --- | --- |
| (LONG index)根據市場別編號與商品代號，取回商品報價的相關資訊。 | | |
| **宣告** | Long SKQuoteLib\_GetStockByMarketAndNo ([in]SHORT sMarketNo,[in] BSTR bstrStockNo, [in,out] struct **SKSTOCKLONG\*** pSKStock); | |
| **參數** | sMarketNo | 市場別編號 |
| bstrStockNo | 商品代號，例如 6005。 |
| pSKStock | SKCOM元件中的**[SKSTOCKLONG](#_SKSTOCKLONG_(國內報價商品物件_LONG)**物件，將物件帶入此欄位中。 |
| **回傳值** | 0表示成功，其餘非0數值都表示失敗。錯誤代碼可參考對照表。 | |
| **備註** | 適用盤中零股  ＊須等收到SK\_SUBJECT\_CONNECTION\_STOCKS\_READY後，方可進行。  未訂閱即時報價[SKQuoteLib\_RequestStocksWithMarketNo](#_SKQuoteLib_RequestStocksWithMarketN)  ，執行此功能，僅可取得商品基本資料(商品名稱、昨收價，非即時性質欄位) | |

### 即時報價

#### SKQuoteLib\_RequestStocks

|  |  |  |
| --- | --- | --- |
| 此功能不支援盤中零股。  訂閱指定商品即時報價  要求伺服器針對 bstrStockNos 內的商品代號訂閱商品報價通知動作。 | | |
| **宣告** | Long SKQuoteLib\_RequestStocks([**in,out**] SHORT\* psPageNo, [in] BSTR bstrStockNos); | |
| **參數** | psPageNo | psPageNo 每一次送出要求報價的動作都需要向報價伺服器指定一個特定的 Page 編號，伺服器會以編號當作Key值維護每一次送出不同的商品代號。  當要變更某一 Page 索取報價的內容，即重複使用相同的 Page 編號，即可取消之前索取的內容，以新的內容取代。當psPageNo=-1時帶入，函式庫會指定一個新的編號，並回傳給呼叫端。 |
| bstrStockNos | 欲訂閱的商品代號，一筆以上的資料時，每檔商品代號以”,”做區隔。 |
| **回傳值** | 0表示成功，其餘非0數值都表示失敗。錯誤代碼可參考對照表。 | |
| **相關通知事件** | [OnNotifyQuoteLONG](#_OnNotifyQuoteLONG) | |
| **備註** | ＊請先[SKQuoteLib\_EnterMonitorLONG](#_SKQuoteLib_EnterMonitorLONG)，須等[OnConnection](#_OnConnection)收到SK\_SUBJECT\_CONNECTION\_STOCKS\_READY後，方可進行訂閱商品報價。  \*因應檔數限制，一個SKQuoteLib物件，僅可擇一使用一個即時報價訂閱  ([SKQuoteLib\_RequestStocks](#_SKQuoteLib_RequestTicks)功能或 [SKQuoteLib\_RequestStocksWithMarketNo](#_SKQuoteLib_RequestTicksWithMarketNo)功  能)。  可重新連線即還原限制與設定。  舉例說明：  psPageNo = -1 時，bstrStockNos = “1101,1102,1103,1104,1108,1109,1110”，表示向伺服器索取這七檔股票的報價。如果是第一次執行，則 psPageNo 將會是回傳1，表示這次的索取是用Page 編號為 1 的 Page。  當想要取消這七檔的股票，改換其他股票時，則：  psPageNo = 1， bstrStockNos = “ bla bla…What you want” 帶入即可。    如果目前用的 Page 僅有 3 頁，但是帶入的 Page>3，則會新增一頁並取代，即回傳 Page=4。  ~~如果帶入的股票數超過100檔，則僅以100檔處理，並不會回傳錯誤。~~  如果帶入的股票代號不存在，則會直接略過不處理，也不會回傳錯誤。  特殊行情：  部分期選商品有T+1盤別交易，若不需要T+1盤的資訊，可在商品代號最後加上AM以取得純AM盤行情，需注意委託商品時加上AM是無法下單的，委託請使用原始商品代號  例：TX00的純AM盤行情為TX00AM | |

修改比較表:

|  |  |  |
| --- | --- | --- |
| 修改前 |  | 修改後V2.13.45 |
| 當psPageNo=-1時帶入，函式庫會指定一個新的編號，並回傳給呼叫端。 | 參數psPageNo | 第一次訂閱，當psPageNo=-1時帶入，函式庫會指定一個新的編號，並回傳給呼叫端。 |
|  | 備註欄位  新增盤後交易介紹及參考資料 | 註:AM代表國內期選早盤行情，可參考臺灣期貨交易所股份有限公司 各商品規格。  查看交易時間  <https://www.taifex.com.tw/cht/2/tX>  T+1盤：  盤後交易介紹  <https://www.taifex.com.tw/cht/4/aHIntroduction> |

|  |
| --- |
| C#範例程式碼\_QuoteForm.cs |
| UI示意圖 |
| |  | | --- | |  | | 拉出 2個TextBox，命名(Name)為「textBoxpsPageNo2」、「textBoxStockNos」。  拉出 1個Button，命名(Name)為「buttonSKQuoteLib\_RequestStocks」。 | |
| 全域變數 |
| // 宣告物件  SKCenterLib m\_pSKCenter = new SKCenterLib(); // 登入&環境設定物件  SKQuoteLib m\_pSKQuote = new SKQuoteLib(); // 國內報價物件 |
| Current Block (可放在buttonSKQuoteLib\_RequestStocks\_Click的Click事件裡) |
| short psPageNo = short.Parse(textBoxpsPageNo2.Text);  // 訂閱指定商品即時報價(註冊)  int nCode = m\_pSKQuote.SKQuoteLib\_RequestStocks(psPageNo, textBoxStockNos.Text);  // 取得回傳訊息  string msg = "【SKQuoteLib\_RequestStocks】" + m\_pSKCenter.SKCenterLib\_GetReturnCodeMessage(nCode);  richTextBoxMethodMessage.AppendText(msg + "\n"); |
| 執行成功畫面 |
|  |

##### SKQuoteLib\_GetStockByIndexLONG

|  |  |  |
| --- | --- | --- |
| 請先訂閱即時報價([SKQuoteLib\_ReqeustStocks](#_SKQuoteLib_RequestTicks)),方可取得商品報價  未訂閱即時報價,僅可取得商品基本資料  (LONG index)根據市場別編號與系統所編的索引代碼，取回商品報價的及商品相關資訊。 | | |
| **宣告** | Long SKQuoteLib\_GetStockByIndexLONG([in] SHORT sMarketNo, [in] LONG nIndex, [in,out] struct **SKSTOCKLONG\*** pSKStock); | |
| **參數** | sMarketNo | 市場別代碼。 |
| nIndex | 系統所編的索引代碼。 |
| pSKStock | SKCOM元件中的 [**SKSTOCKLONG**](#_SKSTOCKLONG_(國內報價商品物件_LONG)物件，將物件帶入此欄位中。 |
| **回傳值** | 0表示成功，其餘非0數值都表示失敗。錯誤代碼可參考對照表。 | |
| **備註** | 市場編號分別為：**與SKSTOCKLONG物件內之BSTR bstrMarketNo 型態不同**。   |  |  | | --- | --- | | 上市 0x00  上櫃 0x01  期貨 0x02  選擇權 0x03  興櫃 0x04 | 盤中零股-上市 0x05  盤中零股-上櫃 0x06  客製化期貨 0x09  客製化選擇權 0x0A |   ＊須等收到SK\_SUBJECT\_CONNECTION\_STOCKS\_READY後，方可進行。  ＊須使用[SKQuoteLib\_EnterMonitorLONG](#_SKQuoteLib_EnterMonitorLONG)登入，方可執行此函式。  ＊依國內或海外市場，可能需搭配MarketNo  ＊未訂閱即時報價，EX:只先訂成交明細[SKQuoteLib\_ReqeustTicks](#_SKQuoteLib_RequestTicks)時，執行此功能，僅可取得商品基本資料(商品名稱、昨收價，非即時性質欄位)  一張含有 文字, 螢幕擷取畫面, 字型, 電子藍 的圖片  自動產生的描述 | |

#### SKQuoteLib\_RequestStocksWithMarketNo

|  |  |  |
| --- | --- | --- |
| 訂閱指定市場別及指定商品即時報價  要求伺服器針對sMarketNo市場別、 bstrStockNos 內的商品代號訂閱商品報價通知動作。 | | |
| **宣告** | Long SKQuoteLib\_RequestStocksWithMarketNo([**in,out**] SHORT\* psPageNo, [**in**] SHORT sMarketNo, [in] BSTR bstrStockNos); | |
| **參數** | psPageNo | psPageNo 每一次送出要求報價的動作都需要向報價伺服器指定一個特定的 Page 編號，伺服器會以編號當作Key值維護每一次送出不同的商品代號。  當要變更某一 Page 索取報價的內容，即重複使用相同的 Page 編號，即可取消之前索取的內容，以新的內容取代。當psPageNo=-1時帶入，函式庫會指定一個新的編號，並回傳給呼叫端。 |
| sMarketNo | (目前提供)  盤中零股-上市(5)、盤中零股-上櫃(6)市場代碼  客製化期貨-9、客製化選擇權-10 |
| bstrStockNos | 欲訂閱的商品代號，一筆以上的資料時，每檔商品代號以”,”做區隔。 |
| **回傳值** | 0表示成功，其餘非0數值都表示失敗。錯誤代碼可參考對照表。 | |
| **相關通知事件** | [OnNotifyQuoteLONG](#_OnNotifyQuoteLONG)  [OnNotifyOddLotSpreadDeal](#_OnNotifyOddLotSpreadDeal) | |
| **備註** | 適用盤中零股  ＊請先[SKQuoteLib\_EnterMonitorLONG](#_SKQuoteLib_EnterMonitorLONG)，須等[OnConnection](#_OnConnection)收到SK\_SUBJECT\_CONNECTION\_STOCKS\_READY後，方可進行訂閱商品報價。  \*因應檔數限制，一個SKQuoteLib物件，僅可擇一使用一個即時報價訂閱([SKQuoteLib\_RequestStocks](#_SKQuoteLib_RequestStocks)功能或 [SKQuoteLib\_RequestStocksWithMarketNo](#_SKQuoteLib_RequestStocksWithMarketN)功能)。  可重新連線即還原限制與設定。 | |

|  |
| --- |
| C#範例程式碼\_QuoteForm.cs |
| UI示意圖 |
| |  | | --- | | 一張含有 螢幕擷取畫面, 文字, 軟體, 網頁 的圖片  自動產生的描述 | | 拉出 1個ComboBox，命名(Name)為「comboBoxSKQuoteLib\_RequestStocksWithMarketNo」。  拉出 1個Button，命名(Name)為「buttonSKQuoteLib\_RequestStocksWithMarketNo」。 | |
| 全域變數 |
| // 宣告物件  SKCenterLib m\_pSKCenter = new SKCenterLib(); // 登入&環境設定物件  SKQuoteLib m\_pSKQuote = new SKQuoteLib(); // 國內報價物件 |
| Constructor |
| //comboBoxSKQuoteLib\_RequestStocksWithMarketNo  {  comboBoxSKQuoteLib\_RequestStocksWithMarketNo.Items.Add("盤中零股-上市(5)");  comboBoxSKQuoteLib\_RequestStocksWithMarketNo.Items.Add("盤中零股-上櫃(6)");  comboBoxSKQuoteLib\_RequestStocksWithMarketNo.Items.Add("客製化期貨-9");  comboBoxSKQuoteLib\_RequestStocksWithMarketNo.Items.Add("客製化選擇權-10");  } |

|  |
| --- |
| Current Block (可放在buttonSKQuoteLib\_RequestStocksWithMarketNo\_Click的Click事件裡) |
| // 訂閱指定市場別及指定商品即時報價  // 要求伺服器針對sMarketNo市場別、 bstrStockNos 內的商品代號訂閱商品報價通知動作  short psPageNo = short.Parse(textBoxpsPageNo2.Text);  short sMarketNo = 5; //盤中零股-上市(5)、盤中零股- 上櫃(6)市場代碼  string selectedValue = comboBoxSKQuoteLib\_RequestStocksWithMarketNo.Text;  if (selectedValue == "盤中零股-上市(5)") sMarketNo = 5;  else if (selectedValue == "盤中零股-上櫃(6)") sMarketNo = 6;  else if (selectedValue == "客製化期貨-9") sMarketNo = 9;  else if (selectedValue == "客製化選擇權-10") sMarketNo = 10;  int nCode = m\_pSKQuote.SKQuoteLib\_RequestStocksWithMarketNo(psPageNo, sMarketNo, textBoxStockNos.Text);  // 取得回傳訊息  string msg = "【SKQuoteLib\_RequestStocksWithMarketNo】" + m\_pSKCenter.SKCenterLib\_GetReturnCodeMessage(nCode);  richTextBoxMethodMessage.AppendText(msg + "\n"); |
| 執行成功畫面 |
|  |

#### SKQuoteLib\_CancelRequestStocks

|  |  |  |
| --- | --- | --- |
| 取消訂閱SKQuoteLib\_RequestStocks的報價通知，並停止更新商品報價。 | | |
| **宣告** | Long SKQuoteLib\_CancelRequestStocks([in] BSTR bstrStockNos); | |
| **參數** | bstrStockNos | 欲解除訂閱的商品代號，一筆以上的資料時，每檔股票代號以”,”做區隔。 |
| **回傳值** | 0表示成功，其餘非0數值都表示失敗。錯誤代碼可參考對照表。 | |
| **備註** | 不支援盤中零股。 | |

|  |
| --- |
| C#範例程式碼\_QuoteForm.cs |
| UI示意圖 |
| |  | | --- | |  | | 拉出 1個Button，命名(Name)為「buttonSKQuoteLib\_CancelRequestStocks」。 | |
| 全域變數 |
| // 宣告物件  SKCenterLib m\_pSKCenter = new SKCenterLib(); // 登入&環境設定物件  SKQuoteLib m\_pSKQuote = new SKQuoteLib(); // 國內報價物件 |

|  |
| --- |
| Current Block (可放在buttonSKQuoteLib\_CancelRequestStocks的Click事件裡) |
| // 取消訂閱SKQuoteLib\_RequestStocks的報價通知，並停止更新商品報價  int nCode = m\_pSKQuote.SKQuoteLib\_CancelRequestStocks(textBoxStockNos.Text);  string[] values = (textBoxStockNos.Text).Split(',');  List<string> listValues = values.ToList();  List<DataGridViewRow> rowsToRemove = new List<DataGridViewRow>();  foreach (DataGridViewRow row in dataGridViewStocks.Rows)  {  foreach(string value in listValues)  {  if (row.Cells[0].Value.ToString() == value)  {  rowsToRemove.Add(row); // 找到要刪除的Rows  }  }  }  // 刪除標記的Rows  foreach (DataGridViewRow rowToRemove in rowsToRemove)  {  dataGridViewStocks.Rows.Remove(rowToRemove);  }  // 取得回傳訊息  string msg = "【SKQuoteLib\_CancelRequestStocks】" + m\_pSKCenter.SKCenterLib\_GetReturnCodeMessage(nCode);  richTextBoxMethodMessage.AppendText(msg + "\n"); |
| 執行成功畫面 |
|  |

#### SKQuoteLib\_GetMarketPriceTS

|  |  |
| --- | --- |
| 取得證券市場逐筆交易價格欄位為市價時之特殊值。 | |
| **宣告** | Long SKQuoteLib\_GetMarketPriceTS(); |
| **參數** | 無 |
| **回傳值** | 市價:-2147483647 |
| **備註** | 證券即時報價，凡買價、賣價為「市價」時判斷依據  如下圖範例所示：  D:\document\策略王_API_包版文件\109\0322-+\市價_即時報價+2207.jpg |

在事件「[OnNotifyQuoteLONG](#_OnNotifyQuoteLONG)」有範例程式碼。

### 五檔&成交明細

#### SKQuoteLib\_RequestTicks

|  |  |  |
| --- | --- | --- |
| 此功能不支援盤中零股。  訂閱要求傳送成交明細以及五檔。 | | |
| **宣告** | Long SKQuoteLib\_RequestTicks([in,out] SHORT\* psPageNo, [in] BSTR bstrStockNo); | |
| **參數** | psPageNo | 每一次送出要求成交明細的動作都需要向報價伺服器指定一個特定的 Page 編號 (請參考RequestStocks觀念說明)。 |
| bstrStockNo | 索取的商品代號，**一個Page僅能索取一檔**。 |
| **回傳值** | 0表示成功，其餘非0數值都表示失敗。錯誤代碼可參考對照表。 | |
| **相關通知事件** | 1. 即時Tick由[OnNotifyTicksLONG](#_OnNotifyTicksLONG)事件通知。 2. Tick回補由 [OnNotifyHistoryTicksLONG](#_OnNotifyHistoryTicksLONG)事件通知。 3. 最佳五檔由[OnNotifyBest5LONG](#_OnNotifyBest5LONG)事件通知。 | |
| **備註** | ＊請先[SKQuoteLib\_EnterMonitorLONG](#_SKQuoteLib_EnterMonitorLONG)，須等[OnConnection](#_OnConnection)收到SK\_SUBJECT\_CONNECTION\_STOCKS\_READY後，方可進行訂閱商品成交明細及最佳五檔。  特殊行情：  部分期選商品有T+1盤別交易，若不需要T+1盤的資訊，可在商品代號最後加上AM以取得純AM盤行情，需注意委託商品時加上AM是無法下單的，委託請使用原始商品代號  例：TX00的純AM盤行情為TX00AM  **\*盤中零股：請使用[SKQuoteLib\_RequestTicksWithMarketNo](#_SKQuoteLib_RequestTicksWithMarketNo)** | |

|  |
| --- |
| C#範例程式碼\_QuoteForm.cs |
| UI示意圖 |
| |  | | --- | |  | | 拉出 2個TextBox，命名(Name)為「textBoxpsPageNo」、「textBoxTicks」。  拉出 1個Button，命名(Name)為「buttonSKQuoteLib\_RequestTicks」。 | |
| 全域變數 |
| // 宣告物件  SKCenterLib m\_pSKCenter = new SKCenterLib(); // 登入&環境設定物件  SKQuoteLib m\_pSKQuote = new SKQuoteLib(); // 國內報價物件 |
| Current Block (可放在buttonSKQuoteLib\_RequestTicks\_Click的Click事件裡) |
| // 此功能不支援盤中零股。訂閱要求傳送成交明細以及五檔  short psPageNo = short.Parse(textBoxpsPageNo.Text);  int nCode = m\_pSKQuote.SKQuoteLib\_RequestTicks(psPageNo, textBoxTicks.Text);  // 取得回傳訊息  string msg = "【SKQuoteLib\_RequestTicks】" + m\_pSKCenter.SKCenterLib\_GetReturnCodeMessage(nCode);  richTextBoxMethodMessage.AppendText(msg + "\n"); |
| 執行成功畫面 |
|  |

##### SKQuoteLib\_GetTickLONG

|  |  |  |
| --- | --- | --- |
| (LONG index)(需**先訂閱即時成交明細RequestTicks**)取得成交明細資訊。 | | |
| **宣告** | Long SKQuoteLib\_GetTickLONG([in] SHORT sMarketNo, [in] LONG nIndex, [in] LONG nPtr, [in,out] struct **SKTICK\*** pSKTick); | |
| **參數** | sMarketNo | 市場別代碼。 |
| nIndex | 系統所編的索引代碼。 |
| nPtr | 第幾筆成交明細，又稱成交明細順序。可以根據此 Ptr 取得**指定第幾筆**成交明細資訊。 |
| pSKTick | SKCOM元件中的 [**SKTICK**](#_SKTICK(國內報價TICK物件))物件，將物件帶入此欄位中。 |
| **回傳值** | 0表示成功，其餘非0數值都表示失敗。錯誤代碼可參考對照表。 | |
| **備註** | 避免在[SKQuoteLib\_OnNotifyHistoryTicksLONG](#_OnNotifyHistoryTicksLONG)、[SKQuoteLib\_OnNotifyTicksLONG](#_OnNotifyTicksLONG)通知事件裡進行[SKQuoteLib\_GetTickLONG](#_SKQuoteLib_GetTickLONG)  ＊須使用[SKQuoteLib\_EnterMonitorLONG](#_SKQuoteLib_EnterMonitorLONG)登入，方可執行此函式。  \*Ptr: 第幾筆成交明細，由0開始  一張含有 文字, 螢幕擷取畫面, 字型, Rectangle 的圖片  自動產生的描述 | |

##### SKQuoteLib\_GetBest5LONG

|  |  |  |
| --- | --- | --- |
| (LONG index)(需先訂閱最佳五檔RequestTicks)取得最佳五檔價格資訊。 | | |
| **宣告** | Long SKQuoteLib\_GetBest5LONG([in] SHORT sMarketNo, [in] LONG nIndex, [in,out] struct **SKBEST5\*** pSKBest5); | |
| **參數** | sMarketNo | 市場別代碼。 |
| nIndex | 系統所編的索引代碼。 |
| pSKStock | SKCOM元件中的 [**SKBEST5**](#_SKBEST5(五檔價格物件))物件，將物件帶入此欄位中。 |
| **回傳值** | 0表示成功，其餘非0數值都表示失敗。錯誤代碼可參考對照表。 | |
| **備註** | 避免在[SKQuoteLib\_OnNotifyBest5LONG](#_OnNotifyBest5LONG)通知事件裡進行[SKQuoteLib\_GetBest5LONG](#_SKQuoteLib_GetBest5LONG)  ＊須使用[SKQuoteLib\_EnterMonitorLONG](#_SKQuoteLib_EnterMonitorLONG)登入，方可執行此函式。 | |

#### SKQuoteLib\_RequestTicksWithMarketNo

|  |  |  |
| --- | --- | --- |
| 訂閱要求傳送成交明細以及五檔。 | | |
| **宣告** | Long SKQuoteLib\_RequestTicksWithMarketNo([in,out] SHORT\* psPageNo, [in] SHORT sMarketNo ,[in] BSTR bstrStockNo); | |
| **參數** | psPageNo | 每一次送出要求成交明細的動作都需要向報價伺服器指定一個特定的 Page 編號  (請參考[SKQuoteLib\_RequestStocks](#_SKQuoteLib_RequestStocks)觀念說明)。 |
| sMarketNo | 盤中零股-上市(5)、盤中零股- 上櫃(6)市場代碼  客製化期貨-9、客製化選擇權-10 |
| bstrStockNo | 索取的商品代號，**一個Page僅能索取一檔**。 |
| **回傳值** | 0表示成功，其餘非0數值都表示失敗。錯誤代碼可參考對照表。 | |
| **相關通知事件** | 1. 即時Tick由[OnNotifyTicksLONG](#_OnNotifyTicksLONG)事件通知。 2. Tick回補由 [OnNotifyHistoryTicksLONG](#_OnNotifyHistoryTicksLONG)事件通知。 3. 最佳五檔由[OnNotifyBest5LONG](#_OnNotifyBest5LONG)事件通知。 | |
| **備註** | 適用盤中零股  ＊請先[SKQuoteLib\_EnterMonitorLONG](#_SKQuoteLib_EnterMonitorLONG)，須等[OnConnection](#_OnConnection)收到SK\_SUBJECT\_CONNECTION\_STOCKS\_READY後，方可進行訂閱商品成交明細及最佳五檔。 | |

|  |
| --- |
| C#範例程式碼\_QuoteForm.cs |
| UI示意圖 |
| |  | | --- | |  | | 拉出1個ComboBox，命名(Name)為「comboBoxSKQuoteLib\_RequestTicksWithMarketNosMarketNo」。  拉出1個Button，命名(Name)為「buttonSKQuoteLib\_RequestTicksWithMarketNo」。 | |
| 全域變數 |
| // 宣告物件  SKCenterLib m\_pSKCenter = new SKCenterLib(); // 登入&環境設定物件  SKQuoteLib m\_pSKQuote = new SKQuoteLib(); // 國內報價物件 |
| Constructor |
| //comboBoxSKQuoteLib\_RequestTicksWithMarketNosMarketNo  {  comboBoxSKQuoteLib\_RequestTicksWithMarketNosMarketNo.Items.Add("盤中零股-上市(5)");  comboBoxSKQuoteLib\_RequestTicksWithMarketNosMarketNo.Items.Add("盤中零股-上櫃(6)");  comboBoxSKQuoteLib\_RequestTicksWithMarketNosMarketNo.Items.Add("客製化期貨-9");  comboBoxSKQuoteLib\_RequestTicksWithMarketNosMarketNo.Items.Add("客製化選擇權-10");  } |

|  |
| --- |
| Current Block (可放在buttonSKQuoteLib\_RequestTicksWithMarketNo\_Click的Click事件裡) |
| // 適用盤中零股，訂閱要求傳送成交明細以及五檔  short psPageNo = short.Parse(textBoxpsPageNo.Text);  short sMarketNo = 5; //盤中零股-上市(5)、盤中零股- 上櫃(6)市場代碼  string selectedValue = comboBoxSKQuoteLib\_RequestTicksWithMarketNosMarketNo.Text;  if (selectedValue == "盤中零股-上市(5)") sMarketNo = 5;  else if (selectedValue == "盤中零股-上櫃(6)") sMarketNo = 6;  else if (selectedValue == "客製化期貨-9") sMarketNo = 9;  else if (selectedValue == "客製化選擇權-10") sMarketNo = 10;  int nCode = m\_pSKQuote.SKQuoteLib\_RequestTicksWithMarketNo(psPageNo, sMarketNo, textBoxTicks.Text);  // 取得回傳訊息  string msg = "【SKQuoteLib2\_RequestTicksWithMarketNo】" + m\_pSKCenter.SKCenterLib\_GetReturnCodeMessage(nCode);  richTextBoxMethodMessage.AppendText(msg + "\n"); |
| 執行成功畫面 |
|  |

#### SKQuoteLib\_RequestLiveTick

|  |  |  |
| --- | --- | --- |
| 訂閱與要求傳送**即時**成交明細。  (本功能不會訂閱最佳五檔，亦不包含歷史Ticks) | | |
| **宣告** | Long SKQuoteLib\_RequestLiveTick([in,out] SHORT\* psPageNo, [in] BSTR bstrStockNo); | |
| **參數** | psPageNo | 每一次送出要求即時成交明細的動作都需要向報價伺服器指定一個特定的 Page 編號，**當代入 Page 為 50 會取消該檔的Tic**k。  (請參考[SKQuoteLib\_RequestStocks](#_SKQuoteLib_RequestStocks)觀念說明) |
| bstrStockNo | 索取的商品代號，**一個Page僅能索取一檔**。 |
| **回傳值** | 0表示成功，其餘非0數值都表示失敗。錯誤代碼可參考對照表。 | |
| **相關通知事件** | 即時Tick由[OnNotifyTicksLONG](#_OnNotifyTicksLONG)事件通知。 | |
| **備註** | ＊請先[SKQuoteLib\_EnterMonitorLONG](#_SKQuoteLib_EnterMonitorLONG)，須等[OnConnection](#_OnConnection)收到SK\_SUBJECT\_CONNECTION\_STOCKS\_READY後，再進行商品檔查詢。  與[SKQuoteLib\_RequestTicks](#_SKQuoteLib_RequestTicks)不同，**不會**回補歷史成交明細資料，請擇一使用;因無回補歷史成交明細資料，[SKQuoteLib\_GetTickLONG](#_SKQuoteLib_GetTickLONG)僅查閱，**訂閱後**之即時成交明細資料。 | |

作法類同「[SKQuoteLib\_RequestTicks](#_SKQuoteLib_RequestTicks)」，請擇一使用。

#### SKQuoteLib\_CancelRequestTicks

|  |  |  |
| --- | --- | --- |
| 取消訂閱RequestTicks的成交明細及五檔。 | | |
| **宣告** | Long SKQuoteLib\_CancelRequestTicks([in] BSTR bstrStockNo); | |
| **參數** | bstrStockNo | 欲解除訂閱的商品代號，**一次僅能解訂閱一檔**。 |
| **回傳值** | 0表示成功，其餘非0數值都表示失敗。錯誤代碼可參考對照表。 | |
| **備註** | 不支援盤中零股。 | |

|  |
| --- |
| C#範例程式碼\_QuoteForm.cs |
| UI示意圖 |
| |  | | --- | |  | | 拉出 1個Button，命名(Name)為「buttonSKQuoteLib\_CancelRequestTicks」。 | |
| 全域變數 |
| // 宣告物件  SKCenterLib m\_pSKCenter = new SKCenterLib(); // 登入&環境設定物件  SKQuoteLib m\_pSKQuote = new SKQuoteLib(); // 國內報價物件 |
| Current Block (可放在buttonSKQuoteLib\_CancelRequestTicks的Click事件裡) |
| // 取消訂閱RequestTicks的成交明細及五檔  int nCode = m\_pSKQuote.SKQuoteLib\_CancelRequestTicks(textBoxTicks.Text);  // 取得回傳訊息  string msg = "【SKQuoteLib\_CancelRequestTicks】" + m\_pSKCenter.SKCenterLib\_GetReturnCodeMessage(nCode);  richTextBoxMethodMessage.AppendText(msg + "\n"); |
| 執行成功畫面 |
|  |

### 技術分析

#### MACD

##### SKQuoteLib\_RequestMACD

|  |  |  |
| --- | --- | --- |
| 要求傳送商品技術指標MACD。(平滑異同平均線) | | |
| **宣告** | Long SKQuoteLib\_RequestMACD([in,out] SHORT\* psPageNo, [in] BSTR bstrStockNo); | |
| **參數** | psPageNo | 每一次送出要求的動作都需要向報價伺服器指定一個  特定的 Page 編號，**當代入 Page 為 50 會取消該檔的MACD**。  (請參考[SKQuoteLib\_RequestStocks](#_SKQuoteLib_RequestStocks)觀念說明)。 |
| bstrStockNo | 索取的商品代號，**一個Page僅能索取一檔**。 |
| **回傳值** | 0表示成功，其餘非0數值都表示失敗。錯誤代碼可參考對照表。 | |
| **相關通知事件** | [OnNotifyMACDLONG](#_OnNotifyMACDLONG) | |
| **備註** | 目前只提供證券市場，不含盤中零股。 | |

|  |
| --- |
| C#範例程式碼\_QuoteForm.cs |
| UI示意圖 |
| |  | | --- | |  | | 拉出 2個TextBox，命名(Name)為「textBoxSKQuoteLib\_RequestMACDpsPageNo」、「textBoxbstrStockNo」。  拉出 1個Button，命名(Name)為「buttonSKQuoteLib\_RequestMACD」。 | |
| 全域變數 |
| // 宣告物件  SKCenterLib m\_pSKCenter = new SKCenterLib(); // 登入&環境設定物件  SKQuoteLib m\_pSKQuote = new SKQuoteLib(); // 國內報價物件 |
| Current Block (可放在buttonSKQuoteLib\_RequestMACD\_Click的Click事件裡) |
| short psPageNo = short.Parse(textBoxSKQuoteLib\_RequestMACDpsPageNo.Text);  string bstrStockNo = textBoxbstrStockNo.Text;  // 要求傳送商品技術指標MACD。(平滑異同平均線)  int nCode = m\_pSKQuote.SKQuoteLib\_RequestMACD(psPageNo, bstrStockNo);  // 取得回傳訊息  string msg = "【SKQuoteLib\_RequestMACD】" + m\_pSKCenter.SKCenterLib\_GetReturnCodeMessage(nCode);  richTextBoxMethodMessage.AppendText(msg + "\n"); |
| 執行成功畫面 |
|  |

##### SKQuoteLib\_GetMACDLONG

|  |  |  |
| --- | --- | --- |
| (LONG index)取得商品技術指標MACD資訊。(平滑異同平均線) | | |
| **宣告** | Long SKQuoteLib\_GetMACDLONG([in] SHORT sMarketNo, [in] LONG nIndex, [in,out] struct **SKMACD\*** pSKMACD); | |
| **參數** | sMarketNo | 市場別代碼。 |
| nIndex | 系統所編的索引代碼。 |
| pSKStock | SKCOM元件中的**[SKMACD](#_SKMACD(國內報價SKMACD物件))**物件，將物件帶入此欄位中。 |
| **回傳值** | 0表示成功，其餘非0數值都表示失敗。錯誤代碼可參考對照表。 | |
| **備註** | 目前只提供證券市場  ＊須使用[SKQuoteLib\_EnterMonitorLONG](#_SKQuoteLib_EnterMonitorLONG)登入，方可執行此函式。 | |

在事件「[OnNotifyMACDLONG](#_OnNotifyMACDLONG)」有範例程式碼。

#### BoolTunel

##### SKQuoteLib\_RequestBoolTunel

|  |  |  |
| --- | --- | --- |
| 要求傳送商品布林通道BoolTunel。 | | |
| **宣告** | Long SKQuoteLib\_RequestBoolTunel([in,out] SHORT\* psPageNo, [in] BSTR bstrStockNo); | |
| **參數** | psPageNo | 每一次送出要求的動作都需要向報價伺服器指定一個  特定的 Page 編號，**當代入 Page 為 50 會取消該檔的BoolTunel**。  (請參考[SKQuoteLib\_RequestStocks](#_SKQuoteLib_RequestStocks)觀念說明)。 |
| bstrStockNo | 索取的商品代號，**一個Page僅能索取一檔**。 |
| **回傳值** | 0表示成功，其餘非0數值都表示失敗。錯誤代碼可參考對照表。 | |
| **相關通知事件** | [OnNotifyBoolTunelLONG](#_OnNotifyBoolTunelLONG) | |
| **備註** | 目前只提供證券市場，不含盤中零股。 | |

|  |
| --- |
| C#範例程式碼\_QuoteForm.cs |
| UI示意圖 |
| |  | | --- | |  | | 拉出 1個Button，命名(Name)為「buttonSKQuoteLib\_RequestBoolTunel」。 | |
| 全域變數 |
| // 宣告物件  SKCenterLib m\_pSKCenter = new SKCenterLib(); // 登入&環境設定物件  SKQuoteLib m\_pSKQuote = new SKQuoteLib(); // 國內報價物件 |
| Current Block (可放在buttonSKQuoteLib\_RequestBoolTunel\_Click的Click事件裡) |
| short psPageNo = short.Parse(textBoxSKQuoteLib\_RequestMACDpsPageNo.Text);  string bstrStockNo = textBoxbstrStockNo.Text;  // 要求傳送商品布林通道BoolTunel  int nCode = m\_pSKQuote.SKQuoteLib\_RequestBoolTunel(psPageNo, bstrStockNo);  // 取得回傳訊息  string msg = "【SKQuoteLib\_RequestBoolTunel】" + m\_pSKCenter.SKCenterLib\_GetReturnCodeMessage(nCode);  richTextBoxMethodMessage.AppendText(msg + "\n"); |
| 執行成功畫面 |
|  |

##### SKQuoteLib\_GetBoolTunelLONG

|  |  |  |
| --- | --- | --- |
| (LONG index)取得商品BoolTunel資訊。 | | |
| **宣告** | Long SKQuoteLib\_GetBoolTunelLONG([in] SHORT sMarketNo, [in] LONG nIndex, [in,out] struct **SKBoolTunel\*** pSKBoolTunel); | |
| **參數** | sMarketNo | 市場別代碼。 |
| nIndex | 系統所編的索引代碼。 |
| pSKBoolTunel | SKCOM元件中的**[SKBoolTunel](#_SKBoolTunel_(國內報價SKBoolTunel物件))**物件，將物件帶入此欄位中。 |
| **回傳值** | 0表示成功，其餘非0數值都表示失敗。錯誤代碼可參考對照表。 | |
| **備註** | 目前只提供證券市場  ＊須使用[SKQuoteLib\_EnterMonitorLONG](#_SKQuoteLib_EnterMonitorLONG)登入，方可執行此函式。 | |

在事件「[OnNotifyBoolTunelLONG](#_OnNotifyBoolTunelLONG)」有範例程式碼。

#### Kline

##### SKQuoteLib\_RequestKLineAMByDate

|  |  |  |
| --- | --- | --- |
| （僅提供歷史資料）向報價伺服器提出，取得單一商品技術分析資訊需求，可選AM盤或全盤，可指定日期區間，分K時可指定幾分K。 | | |
| **宣告** | Long SKQuoteLib\_RequestKLineAMByDate([in] BSTR bstrStockNo, [in] SHORT sKLineType, [in] SHORT sOutType, [in] SHORT sTradeSession, [in] BSTR bstrStartDate, [in] BSTR bstrEndDate, [in] SHORT sMinuteNumber); | |
| **參數** | bstrStockNo | 商品代號，例如 6005。 |
| sKLineType | 0 = 分線。  4 = 日線。  5 = 週線。  6 = 月線。 |
| sOutType | 0 = 舊版輸出格式。  1 = 新版輸出格式。 |
| sTradeSession | 僅國內期權技術分析有效。  0 = 全盤技術分析  1 = AM盤技術分析 |
| bstrStartDate | 起始日期，格式為：YYYYMMDD (ex:20201001) |
| bstrEndDate | 結束日期，格式為：YYYYMMDD (ex:20201010) |
| sMinuteNumber | 指定幾分K (ex : 1=1分K, 3=3分K)  **sKLineType = 0時, 此參數才有意義** |
| **回傳值** | 0表示成功，其餘非0數值都表示失敗。錯誤代碼可參考對照表。 | |
| **相關通知事件** | [OnNotifyKLineData](#_OnNotifyKLineData) | |
| **備註** | 不支援盤中零股。   * 證券:該查詢以整股為主計算，不包含「鉅額交易、盤中零股、盤後零股」 * 國內期選—全盤:   EX: 指定日期區間為2023/9/5全盤K LINE，將包含2023/9/4 17:25開盤後~2023/9/5 05:00，及2023/9/5 8:45~16:15。   * 國內期選—AM盤:當日T盤     \*若您未開立證券帳戶，無法訂閱或取得證券上市櫃、興櫃市場商品及上市櫃指數資料(含歷史K線)。  \*若您未開立期貨帳戶，無法訂閱或取得期貨選擇權相關市場商品資料(含歷史K線)。 | |

比較表:

|  |  |  |
| --- | --- | --- |
| 修改前 | 說明 | V2.13.45 |
| N/A | 備註  新增說明—證券市場 | 該查詢以整股為主計算，不包含「鉅額交易、盤中零股、盤後零股」 |
| N/A | 新增說明—國內期選盤別 | 全盤及AM盤 |

|  |
| --- |
| C#範例程式碼\_QuoteForm.cs |
| UI示意圖 |
| |  | | --- | |  | | 拉出 3個ComboBox，命名(Name)為「comboBoxsKLineType」、「comboBoxsOutType」、「comboBoxsTradeSession」。  拉出 3個TextBox，命名(Name)為「textBoxbstrStartDate」、「textBoxbstrEndDate」、「textBoxsMinuteNumber」。  拉出 1個Button，命名(Name)為「buttonSKQuoteLib\_RequestKLineAMByDate」。 | |
| 全域變數 |
| // 宣告物件  SKCenterLib m\_pSKCenter = new SKCenterLib(); // 登入&環境設定物件  SKQuoteLib m\_pSKQuote = new SKQuoteLib(); // 國內報價物件 |

|  |
| --- |
| Constructor |
| // KLine  {  //comboBoxsKLineType  {  comboBoxsKLineType.Items.Add("分線");  comboBoxsKLineType.Items.Add("日線");  comboBoxsKLineType.Items.Add("週線");  comboBoxsKLineType.Items.Add("月線");  }  //comboBoxsOutType  {  comboBoxsOutType.Items.Add("舊版");  comboBoxsOutType.Items.Add("新版");  }  //sTradeSession  {  comboBoxsTradeSession.Items.Add("全盤");  comboBoxsTradeSession.Items.Add("AM盤");  }  } |

|  |
| --- |
| Current Block (可放在buttonSKQuoteLib\_RequestKLineAMByDate\_Click的Click事件裡) |
| dataGridViewKLine.Rows.Clear();  short sKLineType = 0;  short sOutType = 0;  short sTradeSession = 0;  string bstrStartDate = textBoxbstrStartDate.Text;  string bstrEndDate = textBoxbstrEndDate.Text;  short sMinuteNumber = short.Parse(textBoxsMinuteNumber.Text);  string selectValue = comboBoxsKLineType.SelectedItem.ToString();  if (selectValue == "分線") sKLineType = 0;  else if (selectValue == "日線") sKLineType = 4;  else if (selectValue == "週線") sKLineType = 5;  else if (selectValue == "月線") sKLineType = 6;  selectValue = comboBoxsOutType.SelectedItem.ToString();  if (selectValue == "舊版") sOutType = 0;  else if (selectValue == "新版") sOutType = 1;  selectValue = comboBoxsTradeSession.SelectedItem.ToString();  if (selectValue == "全盤") sTradeSession = 0;  else if (selectValue == "AM盤") sTradeSession = 1;  // （僅提供歷史資料）向報價伺服器提出，取得單一商品技術分析資訊需求，可選AM盤或全盤，可指定日期區間，分K時可指定幾分K  int nCode = m\_pSKQuote.SKQuoteLib\_RequestKLineAMByDate(textBoxbstrStockNo.Text, sKLineType, sOutType, sTradeSession, bstrStartDate, bstrEndDate, sMinuteNumber);  // 取得回傳訊息  string msg = "【SKQuoteLib\_RequestKLineAMByDate】" + m\_pSKCenter.SKCenterLib\_GetReturnCodeMessage(nCode);  richTextBoxMethodMessage.AppendText(msg + "\n"); |
| 執行成功畫面 |
|  |

###### SKQuoteLib\_RequestKLineAM

|  |  |  |
| --- | --- | --- |
| （僅提供歷史資料）向報價伺服器提出，取得單一商品技術分析資訊需求，可選AM盤或全盤。 | | |
| **宣告** | Long SKQuoteLib\_RequestKLineAM([in] BSTR bstrStockNo, [in] SHORT sKLineType, [in] BSTR sOutType, [in] SHORT sTradeSession); | |
| **參數** | bstrStockNo | 商品代號，例如 6005。 |
| sKLineType | 0 = 1分鐘線。  4 =完整日線。  5 =週線。  6 =月線。 |
| sOutType | 0=舊版輸出格式。  1=新版輸出格式。 |
| sTradeSession | 僅國內期權技術分析有效。  0=全盤技術分析  1=AM盤技術分析 |
| **回傳值** | 0表示成功，其餘非0數值都表示失敗。錯誤代碼可參考對照表。 | |
| **相關通知事件** | [OnNotifyKLineData](#_OnNotifyKLineData) | |
| **備註** | * 國內期選—全盤:   EX: 2023/9/5全盤K LINE，將包含2023/9/4 17:25開盤後~2023/9/5 05:00，及2023/9/5 8:45~16:15。   * 國內期選—AM盤:當日T盤   \*若您未開立期貨帳戶，無法訂閱或取得期貨選擇權相關市場商品資料(含歷史K線)。  目前僅提供1分鐘K，其他5分鐘、30分鐘K Line可自行以1分鐘K Line為基礎組成，目前未提供288日K Line。 | |

比較表:

|  |  |  |
| --- | --- | --- |
| 修改前 | 說明 | V2.13.45 |
| N/A | 新增說明—國內期選盤別 | 全盤及AM盤 |

###### SKQuoteLib\_RequestKLine

|  |  |  |
| --- | --- | --- |
| （僅提供歷史資料）向報價伺服器提出，取得單一商品技術分析資訊需求。 | | |
| **宣告** | Long SKQuoteLib\_RequestKLine([in] BSTR bstrStockNo, [in] SHORT sKLineType, [in] BSTR sOutType); | |
| **參數** | bstrStockNo | 商品代號，例如 6005。 |
| sKLineType | 0 = 1分鐘線。  4 =完整日線。  5 =週線。  6 =月線。 |
| sOutType | 0=舊版輸出格式。  1=新版輸出格式。 |
| **回傳值** | 0表示成功，其餘非0數值都表示失敗。錯誤代碼可參考對照表。 | |
| **相關通知事件** | [OnNotifyKLineData](#_OnNotifyKLineData) | |
| **備註** | * 不支援盤中零股。 * 證券:該查詢以整股為主計算，不包含「鉅額交易、盤中零股、盤後零股」 * 國內期選—全盤:   EX: 指定日期區間為2023/9/5全盤K LINE，將包含2023/9/4 17:25開盤後~2023/9/5 05:00，及2023/9/5 8:45~16:15。   * 國內期選—AM盤:當日T盤   \*若您未開立證券帳戶，無法訂閱或取得證券上市櫃、興櫃市場商品及上市櫃指數資料(含歷史K線)。  \*若您未開立期貨帳戶，無法訂閱或取得期貨選擇權相關市場商品資料(含歷史K線)。  solace目前僅提供1分鐘K，其他5分鐘、30分鐘K Line可自行以1分鐘K Line為基礎組成，目前未提供288日K Line。  僅輸出AM盤，若需全盤請使用[SKQuoteLib\_RequestKLineAM](#_4-4-21_SKQuoteLib_RequestKLineAM) | |

比較表:

|  |  |  |
| --- | --- | --- |
| 修改前 | 說明 | V2.13.45 |
| N/A | 備註  新增說明—證券市場 | 該查詢以整股為主計算，不包含「鉅額交易、盤中零股、盤後零股」 |
| N/A | 新增說明—國內期選盤別 | 全盤及AM盤 |

### 風險計算(選擇權)

#### SKQuoteLib\_Delta

|  |  |  |
| --- | --- | --- |
| 輸入  1. nCallPut 商品買賣權別(Call: 0, Put: 1)  2. S 指數  3. K StrikePrice(履約價)  4. R RisklessRate(無風險利率)  5. T RemainMaturity(剩餘天數)  6. sigma  得到Delta值 | | |
| **宣告** | Long SKQuoteLib\_Delta([in] SHORT nCallPut, [in] DOUBLE S, [in] DOUBLE K, [in] DOUBLE R, [in] DOUBLE T, [in] DOUBLE sigma, [out] DOUBLE\* dDelta); | |
| **參數** | nCallPut | 商品買賣權別( Call: 0 , Put: 1) |
| S | 指數 |
| K | StrikePrice(履約價) |
| R | RisklessRate(無風險利率) |
| T | RemainMaturity(剩餘天數) |
| sigma | sigma |
| **回傳值** | 0表示成功，其餘非0數值都表示失敗。錯誤代碼可參考對照表。 | |

|  |
| --- |
| C#範例程式碼\_QuoteForm.cs |
| UI示意圖 |
| |  | | --- | |  | | 拉出 1個ComboBox，命名(Name)為「comboBoxnCallPut」。  拉出 5個TextBox，命名(Name)為「textBoxS」、「textBoxK」、「textBoxR」、「textBoxT」、「textBoxsigma」。  拉出 5個Label，命名(Name)為「labelDelta」、「labelGamma」、「labelVega」、「labelTheta」、「labelRho」。  拉出 1個Button，命名(Name)為「buttonSKQuoteLib\_Gamma」。 | |
| 全域變數 |
| // 宣告物件  SKCenterLib m\_pSKCenter = new SKCenterLib(); // 登入&環境設定物件  SKQuoteLib m\_pSKQuote = new SKQuoteLib(); // 國內報價物件 |

|  |
| --- |
| Current Block (可放在buttonSKQuoteLib\_Gamma\_Click的Click事件裡) |
| double Gamma;  // 輸入1~5得到 DGamma值  int nCode = m\_pSKQuote.SKQuoteLib\_Gamma(double.Parse(textBoxS.Text), double.Parse(textBoxK.Text), double.Parse(textBoxR.Text), double.Parse(textBoxT.Text), double.Parse(textBoxsigma.Text), out Gamma);  labelGamma.Text = "Gamma:" + Gamma;  // 取得回傳訊息  string msg = "【SKQuoteLib\_Gamma】" + m\_pSKCenter.SKCenterLib\_GetReturnCodeMessage(nCode);  richTextBoxMethodMessage.AppendText(msg + "\n");  double Vega;  // 輸入1~5得到 Vega值  nCode = m\_pSKQuote.SKQuoteLib\_Vega(double.Parse(textBoxS.Text), double.Parse(textBoxK.Text), double.Parse(textBoxR.Text), double.Parse(textBoxT.Text), double.Parse(textBoxsigma.Text), out Vega);  labelVega.Text = "Vega:" + Vega;  // 取得回傳訊息  msg = "【SKQuoteLib\_Vega】" + m\_pSKCenter.SKCenterLib\_GetReturnCodeMessage(nCode);  richTextBoxMethodMessage.AppendText(msg + "\n");  short nCallPut = 0; // 商品買賣權別( Call: 0 , Put: 1)  if (comboBoxnCallPut.Text == "Call") nCallPut = 0;  else if (comboBoxnCallPut.Text == "Put") nCallPut = 1;  if (comboBoxnCallPut.SelectedItem != null)  {  double Delta;  // 輸入1~6得到 Delta值  nCode = m\_pSKQuote.SKQuoteLib\_Delta(nCallPut, double.Parse(textBoxS.Text), double.Parse(textBoxK.Text), double.Parse(textBoxR.Text), double.Parse(textBoxT.Text), double.Parse(textBoxsigma.Text), out Delta);  labelDelta.Text = "Delta:" + Delta;  // 取得回傳訊息  msg = "【SKQuoteLib\_Delta】" + m\_pSKCenter.SKCenterLib\_GetReturnCodeMessage(nCode);  richTextBoxMethodMessage.AppendText(msg + "\n");  double Theta;  // 輸入1~6得到 Theta值  nCode = m\_pSKQuote.SKQuoteLib\_Theta(nCallPut, double.Parse(textBoxS.Text), double.Parse(textBoxK.Text), double.Parse(textBoxR.Text), double.Parse(textBoxT.Text), double.Parse(textBoxsigma.Text), out Theta);  labelTheta.Text = "Theta:" + Theta;  // 取得回傳訊息  msg = "【SKQuoteLib\_Theta】" + m\_pSKCenter.SKCenterLib\_GetReturnCodeMessage(nCode);  richTextBoxMethodMessage.AppendText(msg + "\n");  double Rho;  // 輸入1~6得到 Rho值  nCode = m\_pSKQuote.SKQuoteLib\_Rho(nCallPut, double.Parse(textBoxS.Text), double.Parse(textBoxK.Text), double.Parse(textBoxR.Text), double.Parse(textBoxT.Text), double.Parse(textBoxsigma.Text), out Rho);  labelRho.Text = "Rho:" + Rho;  // 取得回傳訊息  msg = "【SKQuoteLib\_Rho】" + m\_pSKCenter.SKCenterLib\_GetReturnCodeMessage(nCode);  richTextBoxMethodMessage.AppendText(msg + "\n");  } |
| 執行成功畫面 |
| |  | | --- | |  | |  | |  | |  | |  | |

#### SKQuoteLib\_Gamma

|  |  |  |
| --- | --- | --- |
| 輸入  1. S 指數  2. K StrikePrice(履約價)  3. R RisklessRate(無風險利率)  4. T RemainMaturity(剩餘天數)  5. sigma  得到Gamma值 | | |
| **宣告** | Long SKQuoteLib\_Gamma([[in] DOUBLE S, [in] DOUBLE K, [in] DOUBLE R, [in] DOUBLE T, [in] DOUBLE sigma, [out] DOUBLE\* dGamma); | |
| **參數** | S | 指數 |
| K | StrikePrice(履約價) |
| R | RisklessRate(無風險利率) |
| T | RemainMaturity(剩餘天數) |
| sigma | sigma |
| **回傳值** | 0表示成功，其餘非0數值都表示失敗。錯誤代碼可參考對照表。 | |

在「[SKQuoteLib\_Delta](#_SKQuoteLib_Delta)」範例程式碼。

#### SKQuoteLib\_Vega

|  |  |  |
| --- | --- | --- |
| 輸入  1. S 台指期成交價  2. K StrikePrice(履約價)  3. R RisklessRate(無風險利率)  4. T RemainMaturity(剩餘天數)  5. v 波動率=sigma  得到Vega值 | | |
| **宣告** | Long SKQuoteLib\_Vega([[in] DOUBLE S, [in] DOUBLE K, [in] DOUBLE R, [in] DOUBLE T, [in] DOUBLE sigma, [out] DOUBLE\* dVega); | |
| **參數** | S | 台指期成交價 |
| K | StrikePrice(履約價) |
| R | RisklessRate(無風險利率) |
| T | RemainMaturity(剩餘天數) |
| sigma | V波動率=sigma |
| **回傳值** | 0表示成功，其餘非0數值都表示失敗。錯誤代碼可參考對照表。 | |

在「[SKQuoteLib\_Delta](#_SKQuoteLib_Delta)」範例程式碼。

#### SKQuoteLib\_Theta

|  |  |  |
| --- | --- | --- |
| 輸入  1. nCallPut 商品買賣權別(Call: 0,Put: 1)  2. S 台指期成交價  3. K StrikePrice(履約價)  4. R RisklessRate(無風險利率)  5. T RemainMaturity(剩餘天數)  6. v 波動率=sigma  得到Theta值 | | |
| **宣告** | Long SKQuoteLib\_Theta(([[in] SHORT nCallPut, [in] DOUBLE S, [in] DOUBLE K, [in] DOUBLE R, [in] DOUBLE T, [in] DOUBLE v, [out] DOUBLE\* dTheta); | |
| **參數** | nCallPut | 商品買賣權別( Call: 0 , Put: 1) |
| S | 台指期成交價 |
| K | StrikePrice(履約價) |
| R | RisklessRate(無風險利率) |
| T | RemainMaturity(剩餘天數) |
| sigma | V波動率=sigma |
| **回傳值** | 0表示成功，其餘非0數值都表示失敗。錯誤代碼可參考對照表。 | |

在「[SKQuoteLib\_Delta](#_SKQuoteLib_Delta)」範例程式碼。

#### SKQuoteLib\_Rho

|  |  |  |
| --- | --- | --- |
| 輸入  1. nCallPut 商品買賣權別(Call: 0, Put: 1)  2. S 台指期成交價  3. K StrikePrice(履約價)  4. R RisklessRate(無風險利率)  5. T RemainMaturity(剩餘天數)  6. v 波動率=sigma  得到Rho值 | | |
| **宣告** | Long SKQuoteLib\_Rho(([[in] SHORT nCallPut, [in] DOUBLE S, [in] DOUBLE K, [in] DOUBLE R, [in] DOUBLE T, [in] DOUBLE v, [out] DOUBLE\* dRho); | |
| **參數** | nCallPut | 商品買賣權別( Call: 0 , Put: 1) |
| S | 指數 |
| K | StrikePrice(履約價) |
| R | RisklessRate(無風險利率) |
| T | RemainMaturity(剩餘天數) |
| sigma | V波動率=sigma |
| **回傳值** | 0表示成功，其餘非0數值都表示失敗。錯誤代碼可參考對照表。 | |

在「[SKQuoteLib\_Delta](#_SKQuoteLib_Delta)」範例程式碼。

### 交易資訊(期選)

#### SKQuoteLib\_RequestFutureTradeInfo

|  |  |  |
| --- | --- | --- |
| 取得報價函式庫註冊接收**期貨**商品的交易資訊。 | | |
| **宣告** | Long SKQuoteLib\_RequestFutureTradeInfo([in] SHORT\* psPageNo, [in] BSTR bstrStockNo); | |
| **參數** | psPageNo | 每一次送出要求的動作都需要向報價伺服器指定一個  特定的 Page 編號。  (請參考[SKQuoteLib\_RequestStocks](#_SKQuoteLib_RequestStocks)觀念說明)。 |
| bstrStockNo | 索取的商品代號，**一個Page僅能索取一檔**。 |
|  |  |
| **相關通知事件** | [OnNotifyFutureTradeInfoLONG](#_OnNotifyFutureTradeInfoLONG) | |
| **回傳值** | 0表示成功，其餘非0數值都表示失敗。錯誤代碼可參考對照表。  若未簽署期貨API下單同意書，將無法查詢。（錯誤代碼3031） | |

|  |
| --- |
| C#範例程式碼\_QuoteForm.cs |
| UI示意圖 |
| |  | | --- | |  | | 拉出 2個TextBox，命名(Name)為「textBoxSKQuoteLib\_RequestFutureTradeInfopsPageNo」、「textBoxSKQuoteLib\_RequestFutureTradeInfobstrStockNo」。  拉出 1個Button，命名(Name)為「buttonSKQuoteLib\_RequestFutureTradeInfo」。 | |
| 全域變數 |
| // 宣告物件  SKCenterLib m\_pSKCenter = new SKCenterLib(); // 登入&環境設定物件  SKQuoteLib m\_pSKQuote = new SKQuoteLib(); // 國內報價物件 |

|  |
| --- |
| Current Block (可放在buttonSKQuoteLib\_RequestFutureTradeInfo\_Click的Click事件裡) |
| short psPageNo = short.Parse(textBoxSKQuoteLib\_RequestFutureTradeInfopsPageNo.Text);  string bstrStockNo = textBoxSKQuoteLib\_RequestFutureTradeInfobstrStockNo.Text;  // 取得報價函式庫註冊接收期貨商品的交易資訊  int nCode = m\_pSKQuote.SKQuoteLib\_RequestFutureTradeInfo(psPageNo, bstrStockNo);  // 取得回傳訊息  string msg = "【SKQuoteLib\_RequestFutureTradeInfo】" + m\_pSKCenter.SKCenterLib\_GetReturnCodeMessage(nCode);  richTextBoxMethodMessage.AppendText(msg + "\n"); |
| 執行成功畫面 |
|  |

#### SKQuoteLib\_GetStrikePrices

|  |  |  |
| --- | --- | --- |
| 取得報價函式庫選擇權交易商品資訊。 | | |
| **宣告** | Long SKQuoteLib\_GetStrikePrices() | |
| **參數** | 無 |  |
| **回傳值** | 0表示成功，其餘非0數值都表示失敗。錯誤代碼可參考對照表。  若未簽署期貨API下單同意書，將無法查詢。（錯誤代碼3031） | |
| **相關通知事件** | [OnNotifyStrikePrices](#_OnNotifyStrikePrices) | |

|  |
| --- |
| C#範例程式碼\_QuoteForm.cs |
| UI示意圖 |
| |  | | --- | |  | | 拉出 1個Button，命名(Name)為「buttonSKQuoteLib\_GetStrikePrices」。 | |
| 全域變數 |
| // 宣告物件  SKCenterLib m\_pSKCenter = new SKCenterLib(); // 登入&環境設定物件  SKQuoteLib m\_pSKQuote = new SKQuoteLib(); // 國內報價物件 |
| Current Block (可放在buttonSKQuoteLib\_GetStrikePrices\_Click的Click事件裡) |
| // 取得報價函式庫選擇權交易商品資訊  int nCode = m\_pSKQuote.SKQuoteLib\_GetStrikePrices();  // 取得回傳訊息  string msg = "【SKQuoteLib\_GetStrikePrices】" + m\_pSKCenter.SKCenterLib\_GetReturnCodeMessage(nCode);  richTextBoxMethodMessage.AppendText(msg + "\n"); |
| 執行成功畫面 |
|  |

# 事件

## 連線

### OnConnection

|  |  |  |
| --- | --- | --- |
| 接收連線狀態。 | | |
| **宣告** | void OnConnection([in] LONG nKind, [in] LONG nCode); | |
| **參數** | nKind | SK\_SUBJECT\_CONNECTION\_CONNECTED 表示為連線事件。  SK\_SUBJECT\_CONNECTION\_DISCONNECT 表示為斷線事件。  SK\_SUBJECT\_CONNECTION\_STOCKS\_READY 表示為商品基本資料已下載完成。  其他錯誤代碼，可參考代碼定義表 |
| nCode | 表示事件所獲得的代碼，代碼為0值即表示執行正確，非0表示例外事件。 |
| **備註** | 避免在此處直接進行[EnterMonitorLONG](#_SKQuoteLib_EnterMonitorLONG)、[LeaveMonitor](#_SKQuoteLib_LeaveMonitor)、[RequestStocks](#_SKQuoteLib_RequestStocks) and [RequestTicks](#_SKQuoteLib_RequestTicks) 等報價連線、斷線、訂閱，若全市場商品未下載完成，即無法進行商品訂閱。 | |

|  |
| --- |
| C#範例程式碼\_QuoteForm.cs |
| UI示意圖 |
| 拉出1個richTextBox，命名(Name)為「richTextBoxMessage」。 |
| 全域變數 |
| // 宣告物件  SKCenterLib m\_pSKCenter = new SKCenterLib(); // 登入&環境設定物件  SKQuoteLib m\_pSKQuote = new SKQuoteLib(); // 國內報價物件 |
| Current Block (可放在Form的Load事件裡) |
| //當連線成功或失敗，會透過此事件函式告知連線結果  {  m\_pSKQuote.OnConnection += new \_ISKQuoteLibEvents\_OnConnectionEventHandler(OnConnection);  void OnConnection(int nKind, int nCode)  {  // 取得回傳訊息  string msg = "【OnConnection】" + m\_pSKCenter.SKCenterLib\_GetReturnCodeMessage(nKind);  richTextBoxMessage.AppendText(msg + "\n");  }  } |

|  |
| --- |
| 成功示意畫面 |
|  |

### OnNotifyServerTime

|  |  |  |
| --- | --- | --- |
| 事件回傳查詢主機時間的結果。 | | |
| **宣告** | void OnNotifyServerTime([in] SHORT sHour, [in] SHORT sMinute, [in] SHORT sSecond, [in] LONG nTotal); | |
| **參數** | sHour | 時 |
| sMinute | 分 |
| sSecond | 秒 |
| nTotal | 總秒數 |
| **備註** |  | |

|  |
| --- |
| C#範例程式碼\_QuoteForm.cs |
| UI示意圖 |
| 拉出1個Label，命名(Name)為「labelOnNotifyServerTime2」。 |
| 全域變數 |
| // 宣告物件  SKCenterLib m\_pSKCenter = new SKCenterLib(); // 登入&環境設定物件  SKQuoteLib m\_pSKQuote = new SKQuoteLib(); // 國內報價物件 |
| Current Block (可放在Form的Load事件裡) |
| //事件回傳查詢主機時間的結果  {  m\_pSKQuote.OnNotifyServerTime += new \_ISKQuoteLibEvents\_OnNotifyServerTimeEventHandler(OnNotifyServerTime);  void OnNotifyServerTime(short sHour, short sMinute, short sSecond, int nTotal)  {  labelOnNotifyServerTime2.Text = sHour + ":" + sMinute + ":" + sSecond + " Total:" + nTotal;  }  } |

|  |
| --- |
| 成功示意畫面 |
|  |

## 大盤資訊

### OnNotifyMarketTot

|  |  |  |
| --- | --- | --- |
| 透過呼叫[SKQuoteLib\_GetMarketBuySellUpDown](#_SKQuoteLib_GetMarketBuySellUpDown)後，  事件回傳大盤成交張筆資料。 | | |
| **宣告** | void OnNotifyMarketTot([in] SHORT sMarketNo, [in] SHORT sPtr, [in] LONG nTime, [in] LONG nTotv, [in] LONG nTots, [in] LONG nTotc); | |
| **參數** | sMarketNo | 市場別代號 (0x00 上市, 0x01 上櫃) |
| sPtr | 目前第幾筆資料。  可以根據此 index 取得該筆資訊。 |
| nTime | 大盤成交時間。Ex. lTime = 92000 ，為早上九點20分(09:20)。 |
| nTotv | 大盤成交值(億)。Ex. lTotv = 88542，為885.42 (億)。 |
| nTots | 大盤成交張數。 |
| nTotc | 大盤成交筆數。 |
| **備註** |  | |

|  |
| --- |
| C#範例程式碼\_QuoteForm.cs |
| UI示意圖 |
| |  | | --- | |  | | 拉出19個Label (順序由上到下，左到右)，命名(Name)為「labelMarketTotPtr」、「labelnTotv」、「labelnTots」、「labelnTotc」、「labelnUp」、「labelnDown」、「labelnHigh」、「labelnLow」、「labelnNoChange」、「labelMarketTotTime」、「labelnBs」、「labelnSs」、「labelnBc」、「labelnSc」、「labelnUpNoW」、「labelnDownNoW」、「labelnHighNoW」、「labelnLowNoW」、「labelnNoChangeNoW」。 | |  | | 拉出19個Label (順序由上到下，左到右)，命名(Name)為「labelMarketTotPtr2」、「labelnTotv2」、「labelnTots2」、「labelnTotc2」、「labelnUp2」、「labelnDown2」、「labelnHigh2」、「labelnLow2」、「labelnNoChange2」、「labelMarketTotTime2」、「labelnBs2」、「labelnSs2」、「labelnBc2」、「labelnSc2」、「labelnUpNoW2」、「labelnDownNoW2」、「labelnHighNoW2」、「labelnLowNoW2」、「labelnNoChangeNoW2」。 | |
| 全域變數 |
| // 宣告物件  SKCenterLib m\_pSKCenter = new SKCenterLib(); // 登入&環境設定物件  SKQuoteLib m\_pSKQuote = new SKQuoteLib(); // 國內報價物件 |
| Current Block (可放在Form的Load事件裡) |
| //透過呼叫 SKQuoteLib\_GetMarketBuySellUpDown 後，事件回傳大盤成交張筆資料  {  m\_pSKQuote.OnNotifyMarketTot += new \_ISKQuoteLibEvents\_OnNotifyMarketTotEventHandler(OnNotifyMarketTot);  void OnNotifyMarketTot(short sMarketNo, short sPtr, int nTime, int nTotv, int nTots, int nTotc)  {  if (sMarketNo == 0) // 上市  {  labelMarketTotPtr.Text = "目前第" + sPtr + "筆資料";  labelMarketTotTime.Text = "大盤成交時間:" + nTime / 10000 + "點" + (nTime % 10000) / 100 + "分" + nTime % 100 + "秒";  labelnTotv.Text = "成交量(億):" + (nTotv / 100.00);  labelnTots.Text = "成交張數:" + nTots;  labelnTotc.Text = "成交筆數:" + nTotc;  }  else // 上櫃2  {  labelMarketTotPtr2.Text = "目前第" + sPtr + "筆資料";  labelMarketTotTime2.Text = "大盤成交時間:" + nTime / 10000 + "點" + (nTime % 10000) / 100 + "分" + nTime % 100 + "秒";  labelnTotv2.Text = "成交量(億):" + (nTotv / 100.00);  labelnTots2.Text = "成交張數:" + nTots;  labelnTotc2.Text = "成交筆數:" + nTotc;  }  }  }  //透過呼叫 SKQuoteLib\_GetMarketBuySellUpDown 後，事件回傳大盤成交買賣張筆數資料  {  m\_pSKQuote.OnNotifyMarketBuySell += new \_ISKQuoteLibEvents\_OnNotifyMarketBuySellEventHandler(OnNotifyMarketBuySell);  void OnNotifyMarketBuySell(short sMarketNo, short sPtr, int nTime, int nBc, int nSc, int nBs, int nSs)  {  //labelMarketBuySellPtr.Text = "目前第" + sPtr + "筆資料";  //labelMarketBuySellTime.Text = "大盤成交時間:" + nTime / 10000 + "點" + (nTime % 10000) / 100 + "分" + nTime % 100 + "秒";  if (sMarketNo == 0) // 上市  {  labelnBs.Text = "成交買進張數:" + nBs;  labelnSs.Text = "成交賣出張數:" + nSs;  labelnBc.Text = "成交買進筆數:" + nBc;  labelnSc.Text = "成交賣出筆數:" + nSc;  }  else // 上櫃2  {  labelnBs2.Text = "成交買進張數:" + nBs;  labelnSs2.Text = "成交賣出張數:" + nSs;  labelnBc2.Text = "成交買進筆數:" + nBc;  labelnSc2.Text = "成交賣出筆數:" + nSc;  }  }  }  //透過呼叫 SKQuoteLib\_GetMarketBuySellUpDown 後，事件回傳大盤成交上漲下跌家數資料(包含『含權證家數』、 『不含權證家數』)  {  m\_pSKQuote.OnNotifyMarketHighLowNoWarrant += new \_ISKQuoteLibEvents\_OnNotifyMarketHighLowNoWarrantEventHandler(OnNotifyMarketHighLowNoWarrant);  void OnNotifyMarketHighLowNoWarrant(short sMarketNo, int nPtr, int nTime, int nUp, int nDown, int nHigh, int nLow, int nNoChange, int nUpNoW, int nDownNoW, int nHighNoW, int nLowNoW, int nNoChangeNoW)  {  if (sMarketNo == 0) // 上市  {  labelnUp.Text = "成交上漲家數:" + nUp;  labelnDown.Text = "成交下跌家數:" + nDown;  labelnHigh.Text = "成交漲停家數:" + nHigh;  labelnLow.Text = "成交跌停家數:" + nLow;  labelnNoChange.Text = "平盤家數:" + nNoChange;  labelnUpNoW.Text = "(不含權證)成交上漲家數:" + nUpNoW;  labelnDownNoW.Text = "(不含權證)成交下跌家數:" + nDownNoW;  labelnHighNoW.Text = "(不含權證)成交漲停家數:" + nHighNoW;  labelnLowNoW.Text = "(不含權證)成交跌停家數:" + nLowNoW;  labelnNoChangeNoW.Text = "(不含權證)平盤家數:" + nNoChangeNoW;  }  else // 上櫃2  {  labelnUp2.Text = "成交上漲家數:" + nUp;  labelnDown2.Text = "成交下跌家數:" + nDown;  labelnHigh2.Text = "成交漲停家數:" + nHigh;  labelnLow2.Text = "成交跌停家數:" + nLow;  labelnNoChange2.Text = "平盤家數:" + nNoChange;  labelnUpNoW2.Text = "(不含權證)成交上漲家數:" + nUpNoW;  labelnDownNoW2.Text = "(不含權證)成交下跌家數:" + nDownNoW;  labelnHighNoW2.Text = "(不含權證)成交漲停家數:" + nHighNoW;  labelnLowNoW2.Text = "(不含權證)成交跌停家數:" + nLowNoW;  labelnNoChangeNoW2.Text = "(不含權證)平盤家數:" + nNoChangeNoW;  }  }  } |

|  |
| --- |
| 成功示意畫面 |
|  |

#### OnNotifyMarketBuySell

|  |  |  |
| --- | --- | --- |
| 透過呼叫[SKQuoteLib\_GetMarketBuySellUpDown](#_SKQuoteLib_GetMarketBuySellUpDown)後，  事件回傳大盤成交買賣張筆數資料。 | | |
| **宣告** | void OnNotifyMarketBuySell([in] SHORT sMarketNo, [in] SHORT sPtr, [in] LONG nTime, [in] LONG nBc, [in] LONG nSc, [in] LONG nBs, [in] LONG nSs); | |
| **參數** | sMarketNo | 市場別代號 (0x00 上市, 0x01 上櫃) |
| sPtr | 目前第幾筆資料。  可以根據此 index 取得該筆資訊。 |
| nTime | 大盤成交時間。Ex. lTime = 92000 ，為早上九點20分(09:20)。 |
| nBc | 大盤成交買進筆數 |
| nSc | 大盤成交賣出筆數 |
| nBs | 大盤成交買進張數 |
| nSs | 大盤成交賣出張數 |
| **備註** |  | |

#### OnNotifyMarketHighLowNoWarrant

|  |  |  |
| --- | --- | --- |
| 透過呼叫[SKQuoteLib\_GetMarketBuySellUpDown](#_SKQuoteLib_GetMarketBuySellUpDown)後，  事件回傳大盤成交上漲下跌家數資料(包含『含權證家數』、 『不含權證家數』)。 | | |
| **宣告** | void OnNotifyMarketHighLowNoWarrant([in] SHORT sMarketNo, [in] LONG nPtr, [in] LONG nTime, [in] LONG nUp, [in] LONG nDown, [in] LONG nHigh, [in] LONG nLow, [in] LONG nUnChange, [in] LONG nUpNoW, [in] LONG nDownNoW, [in] LONG nHighNoW, [in] LONG nLowNoW, [in] LONG nUnChangeNoW); | |
| **參數** | sMarketNo | 市場別代號 (0x00 上市, 0x01 上櫃) |
| sPtr | 目前第幾筆資料。  可以根據此 index 取得該筆資訊。 |
| nTime | 大盤成交時間。Ex. lTime = 92000 ，為早上九點20分(09:20)。 |
| nUp | 大盤成交上漲家數 |
| nDown | 大盤成交下跌家數 |
| nHigh | 大盤成交漲停家數 |
| nLow | 大盤成交跌停家數 |
| nNoChange | 大盤平盤家數 |
| nUpNoW | (不含權證)大盤成交上漲家數 |
| nDownNoW | (不含權證)大盤成交下跌家數 |
| nHighNoW | (不含權證)大盤成交漲停家數 |
| nLowNoW | (不含權證)大盤成交跌停家數 |
| nUnChangeNoW | (不含權證)大盤平盤家數 |
| **備註** |  | |

##### OnNotifyMarketHighLow

|  |  |  |
| --- | --- | --- |
| 透過呼叫[SKQuoteLib\_GetMarketBuySellUpDown](#_SKQuoteLib_GetMarketBuySellUpDown)後，  事件回傳大盤成交上漲下跌家數資料。 | | |
| **宣告** | void OnNotifyMarketHighLow([in] SHORT sMarketNo, [in] SHORT sPtr, [in] LONG nTime, [in] SHORT sUp, [in] SHORT sDown, [in] SHORT sHigh, [in] SHORT sLow, [in] SHORT sNoChange); | |
| **參數** | sMarketNo | 市場別代號 (0x00 上市, 0x01 上櫃) |
| sPtr | 目前第幾筆資料。  可以根據此 index 取得該筆資訊。 |
| nTime | 大盤成交時間。Ex. lTime = 92000 ，為早上九點20分(09:20)。 |
| sUp | 大盤成交上漲家數 |
| sDown | 大盤成交下跌家數 |
| sHigh | 大盤成交漲停家數 |
| sLow | 大盤成交跌停家數 |
| sNoChange | 大盤平盤家數 |
| **備註** |  | |

## 商品清單

### OnNotifyCommodityListWithTypeNo

|  |  |  |
| --- | --- | --- |
| 事件回傳指定國內市場－含類別代碼及類別中文名稱之商品清單。 | | |
| **宣告** | void OnNotifyCommodityListWithTypeNo([in] SHORT sMarketNo, [in] BSTR bstrCommodityData); | |
| **參數** | sMarketNo | 市場別代號 |
| bstrCommodityData | **開頭含類別代碼，**格式為「%」、類別代碼「%」類別中文名稱，**結尾為「%」。**  EX: %1%水泥%   |  |  | | --- | --- | | 0 | **類別代碼**。例如，1、2。 | | 1 | **類別中文名稱**。例如，水泥、食品等。 |   接著為商品資料(商品代碼及商品中文名稱)  以上市0 – 水泥類股為例，以「,」分隔每一欄，欄位依序為：  [商品代碼],[商名名稱],[最後交易日],[交易所商品代碼];   |  |  | | --- | --- | | 0 | Product商品代碼。  例如，1101。 | | 1 | Name中文名稱。例如，台泥。 | | 2 | LastTradeDay 最後交易日，目前僅提供期貨及選擇權。  例如，20180719  當為0時代表沒有最後交易日或為不可交易商品。 | | 3 | OrderID 交易所商品代碼，目前僅提供期貨及選擇權。  例如，TXFI9。 |   另以「；」做為下一筆資料分隔，  而各類別結尾（水泥、食品、塑膠等）則以「換行」做分隔。  以上市商品-代碼0  **%1%水泥%**1101,台泥,0,;1102,亞泥,0,;1103,嘉泥,0,;1104,環泥,0,;1108,幸福,0,;1109,信大,0,;1110,東泥,0,;  **%2%食品%**1201,味全,0,;1203,味王,0,; 1210,大成,0,; 1213,大飲,0,;1215,卜蜂,0,;….; |
| **備註** | 當全部資料已經全部回傳完畢，將回傳一筆以「##」開頭的內容，表示查詢結束。 | |

|  |
| --- |
| C#範例程式碼\_QuoteForm.cs |
| UI示意圖 |
| 拉出1個DataGridView，命名(Name)為「dataGridViewOnNotifyCommodityListWithTypeNo」。 |
| 全域變數 |
| // 宣告物件  SKCenterLib m\_pSKCenter = new SKCenterLib(); // 登入&環境設定物件  SKQuoteLib m\_pSKQuote = new SKQuoteLib(); // 國內報價物件 |
| Constructor |
| // dataGridViewOnNotifyCommodityListWithTypeNo  {  dataGridViewOnNotifyCommodityListWithTypeNo.Columns.Add("Column1", "商品代碼");  dataGridViewOnNotifyCommodityListWithTypeNo.Columns.Add("Column2", "商名名稱");  dataGridViewOnNotifyCommodityListWithTypeNo.Columns.Add("Column3", "最後交易日(僅期權)");  dataGridViewOnNotifyCommodityListWithTypeNo.Columns.Add("Column4", "交易所商品代碼(僅期權)");  } |

|  |
| --- |
| Current Block (可放在Form的Load事件裡) |
| //事件回傳指定國內市場－各類股商品清單  {  m\_pSKQuote.OnNotifyCommodityListWithTypeNo += new \_ISKQuoteLibEvents\_OnNotifyCommodityListWithTypeNoEventHandler(OnNotifyCommodityListWithTypeNo);  void OnNotifyCommodityListWithTypeNo(short sMarketNo, string bstrStockData)  {  // 先把類別代碼及類別中文名稱取出  string[] valueClass = bstrStockData.Split('%');  if (bstrStockData != "##,,;")  {  dataGridViewOnNotifyCommodityListWithTypeNo.Rows.Add("類別代碼:" + valueClass[1] + valueClass[2]);  bstrStockData = valueClass[3];  }  // 先分成各筆資料 ';'  string[] valuesSemicolon = bstrStockData.Split(';');  for (int i = 0; i < valuesSemicolon.Length - 1; i++) // 切割會把最後一個;也算進去，所以真實長度要-1  {  string[] valuesComma = valuesSemicolon[i].Split(',');  if (bstrStockData != "##,,;") dataGridViewOnNotifyCommodityListWithTypeNo.Rows.Add(valuesComma[0], valuesComma[1], valuesComma[2], valuesComma[3]);  }  }  } |

|  |
| --- |
| 成功示意畫面 |
|  |

#### OnNotifyStockList

|  |  |  |
| --- | --- | --- |
| 事件回傳指定國內市場－各類股商品清單。 | | |
| **宣告** | void OnNotifyStockList([in] SHORT sMarketNo, [in] BSTR bstrStockData); | |
| **參數** | sMarketNo | 市場別代號 |
| bstrStockData | 商品資料(商品代碼及商品中文名稱)  以上市0 – 水泥類股為例，以「,」分隔每一欄，欄位依序為：  [商品代碼],[商名名稱],[最後交易日],[交易所商品代碼];   |  |  | | --- | --- | | 0 | Product商品代碼。  例如，1101。 | | 1 | Name中文名稱。例如，台泥。 | | 2 | LastTradeDay 最後交易日，目前僅提供期貨及選擇權  例如，20180719  當為0時代表沒有最後交易日或為不可交易商品。 | | 3 | OrderID 交易所商品代碼，目前僅提供期貨及選擇權。  例如，TXFI9。 |   另以「；」做為下一筆資料分隔，  而各類股結尾（水泥、食品、塑膠等）則以「換行」做分隔。  1101,台泥,0,;1102,亞泥,0,;1103,嘉泥,0,;1104,環泥,0,;1108,幸福,0,;1109,信大,0,;1110,東泥,0,;  1201,味全,0,;1203,味王,0,; 1210,大成,0,; 1213,大飲,0,;1215,卜蜂,0,;….; |
| **備註** | 當全部資料已經全部回傳完畢，將回傳一筆以「##」開頭的內容，表示查詢結束。 | |

## 即時報價

### OnNotifyQuoteLONG

|  |  |  |
| --- | --- | --- |
| (LONG index)當有索取的個股報價異動時，將透過此事件通知應用程式處理。 | | |
| **宣告** | void OnNotifyQuoteLONG([in] SHORT sMarketNo, [in] LONG nIndex); | |
| **參數** | sMarketNo | 報價有異動的商品市場別。 |
| nIndex | 系統所編的索引代碼。 |
| **備註** | 透過以上兩個參數，可以再使用[SKQuoteLib\_GetStockByIndexLONG](#_SKQuoteLib_GetStockByIndexLONG) 取出報價的內容。  \*開發人員可於OnUpDateDataRow(SKSTOCK pStock) 取得Stock揭示值，另外自行接手處理  \*依國內或海外市場，可能需搭配MarketNo  ＊須使用[SKQuoteLib\_EnterMonitorLONG](#_SKQuoteLib_EnterMonitorLONG)登入，該事件才會被觸發。  一張含有 文字, 螢幕擷取畫面, 字型, 電子藍 的圖片  自動產生的描述 | |

|  |
| --- |
| C#範例程式碼\_QuoteForm.cs |
| UI示意圖 |
| 拉出1個DataGridView，命名(Name)為「dataGridViewOnNotifyOddLotSpreadDeal」。 |
| 全域變數 |
| // 宣告物件  SKCenterLib m\_pSKCenter = new SKCenterLib(); // 登入&環境設定物件  SKQuoteLib m\_pSKQuote = new SKQuoteLib(); // 國內報價物件 |
| Constructor |
| // dataGridViewStocks  {  dataGridViewStocks.Columns.Add("Column1", "代碼");  dataGridViewStocks.Columns.Add("Column2", "名稱");  dataGridViewStocks.Columns.Add("Column3", "開盤價");  dataGridViewStocks.Columns.Add("Column4", "成交價");  dataGridViewStocks.Columns.Add("Column5", "最高");  dataGridViewStocks.Columns.Add("Column6", "最低");  dataGridViewStocks.Columns.Add("Column7", "漲停價");  dataGridViewStocks.Columns.Add("Column8", "跌停價");  dataGridViewStocks.Columns.Add("Column9", "買盤量(外盤)");  dataGridViewStocks.Columns.Add("Column10", "賣盤量(內盤)");  dataGridViewStocks.Columns.Add("Column11", "總量");  dataGridViewStocks.Columns.Add("Column12", "昨收(參考價)");  dataGridViewStocks.Columns.Add("Column13", "昨量");  dataGridViewStocks.Columns.Add("Column14", "買價");  dataGridViewStocks.Columns.Add("Column15", "賣價");  } |

|  |
| --- |
| Current Block (可放在Form的Load事件裡) |
| //事件報價通知  {  m\_pSKQuote.OnNotifyQuoteLONG += new \_ISKQuoteLibEvents\_OnNotifyQuoteLONGEventHandler(OnNotifyQuoteLONG);  void OnNotifyQuoteLONG(short sMarketNo, int nIndex)  {  bool stockFound = false;  SKSTOCKLONG pSKStock = new SKSTOCKLONG();  int nCode = m\_pSKQuote.SKQuoteLib\_GetStockByIndexLONG(sMarketNo, nIndex, ref pSKStock);    foreach (DataGridViewRow row in dataGridViewStocks.Rows)  {  if (row.Cells[0].Value.ToString() == pSKStock.bstrStockNo)  {  row.Cells[2].Value = pSKStock.nOpen / 100.0;  row.Cells[3].Value = pSKStock.nClose / 100.0;  row.Cells[4].Value = pSKStock.nHigh / 100.0;  row.Cells[5].Value = pSKStock.nLow / 100.0;  row.Cells[6].Value = pSKStock.nUp / 100.0;  row.Cells[7].Value = pSKStock.nDown / 100.0;  row.Cells[8].Value = pSKStock.nTBc.ToString();  row.Cells[9].Value = pSKStock.nTAc;  row.Cells[10].Value = pSKStock.nTQty;  row.Cells[11].Value = pSKStock.nRef / 100.0;  row.Cells[12].Value = pSKStock.nYQty;  if (pSKStock.nBid == m\_pSKQuote.SKQuoteLib\_GetMarketPriceTS()) // (買價) 取得證券市場逐筆交易價格欄位為市價時之特殊值  {  row.Cells[13].Value = "市價";  }  else  {  row.Cells[13].Value = pSKStock.nBid / 100.0;  }  if (pSKStock.nBid == m\_pSKQuote.SKQuoteLib\_GetMarketPriceTS()) // (賣價) 取得證券市場逐筆交易價格欄位為市價時之特殊值  {  row.Cells[14].Value = "市價";  }  else  {  row.Cells[14].Value = pSKStock.nAsk / 100.0;  }  stockFound = true;  break;  }  }  if (stockFound == false)  {  if (pSKStock.nBid == m\_pSKQuote.SKQuoteLib\_GetMarketPriceTS() && pSKStock.nAsk == m\_pSKQuote.SKQuoteLib\_GetMarketPriceTS())  {  dataGridViewStocks.Rows.Add(pSKStock.bstrStockNo, pSKStock.bstrStockName, pSKStock.nOpen / 100.0, pSKStock.nClose / 100.0, pSKStock.nHigh / 100.0, pSKStock.nLow / 100.0, pSKStock.nUp / 100.0, pSKStock.nDown / 100.0, pSKStock.nTBc.ToString(), pSKStock.nTAc, pSKStock.nTQty, pSKStock.nRef / 100.0, pSKStock.nYQty, "市價", "市價");  }  else if (pSKStock.nBid == m\_pSKQuote.SKQuoteLib\_GetMarketPriceTS() && pSKStock.nAsk != m\_pSKQuote.SKQuoteLib\_GetMarketPriceTS())  {  dataGridViewStocks.Rows.Add(pSKStock.bstrStockNo, pSKStock.bstrStockName, pSKStock.nOpen / 100.0, pSKStock.nClose / 100.0, pSKStock.nHigh / 100.0, pSKStock.nLow / 100.0, pSKStock.nUp / 100.0, pSKStock.nDown / 100.0, pSKStock.nTBc.ToString(), pSKStock.nTAc, pSKStock.nTQty, pSKStock.nRef / 100.0, pSKStock.nYQty, "市價", pSKStock.nAsk / 100.0);  }  else if (pSKStock.nBid != m\_pSKQuote.SKQuoteLib\_GetMarketPriceTS() && pSKStock.nAsk == m\_pSKQuote.SKQuoteLib\_GetMarketPriceTS())  {  dataGridViewStocks.Rows.Add(pSKStock.bstrStockNo, pSKStock.bstrStockName, pSKStock.nOpen / 100.0, pSKStock.nClose / 100.0, pSKStock.nHigh / 100.0, pSKStock.nLow / 100.0, pSKStock.nUp / 100.0, pSKStock.nDown / 100.0, pSKStock.nTBc.ToString(), pSKStock.nTAc, pSKStock.nTQty, pSKStock.nRef / 100.0, pSKStock.nYQty, pSKStock.nBid / 100.0, "市價");  }  else  {  dataGridViewStocks.Rows.Add(pSKStock.bstrStockNo, pSKStock.bstrStockName, pSKStock.nOpen / 100.0, pSKStock.nClose / 100.0, pSKStock.nHigh / 100.0, pSKStock.nLow / 100.0, pSKStock.nUp / 100.0, pSKStock.nDown / 100.0, pSKStock.nTBc.ToString(), pSKStock.nTAc, pSKStock.nTQty, pSKStock.nRef / 100.0, pSKStock.nYQty, pSKStock.nBid / 100.0, pSKStock.nAsk / 100.0);  }  }    }  } |

|  |
| --- |
| 成功示意畫面 |
|  |

### OnNotifyOddLotSpreadDeal

|  |  |  |
| --- | --- | --- |
| 事件回傳證券市場－整零價差即時行情 | | |
| **宣告** | void OnNotifyOddLotSpreadDeal([in] SHORT sMarketNo, [in] BSTR bstrStockNo, [in] LONG nDealPrice, [in] LONG nDigit); | |
| **參數** | sMarketNo | 市場別代號 |
| bstrStockNo | 商品代碼。 |
| nDealPrice | 整零成交價差(負數則含-負號) |
| nDigit | 小數位數 |
| **備註** | 目前只提供證券市場；實際價差，需再透過小數位數自行計算。  \*公式:  整零價差=整股成交價-零股成交價。  若整股當日無成交價，以昨收價計算。  若零股當日無成交價，不計算整零價差。 | |

V2.13.46 修正文件宣告內容敘述 (移除[in] CHAR cSignDeal)

|  |
| --- |
| C#範例程式碼\_QuoteForm.cs |
| UI示意圖 |
| 拉出1個DataGridView，命名(Name)為「dataGridViewOnNotifyOddLotSpreadDeal」。 |
| 全域變數 |
| // 宣告物件  SKCenterLib m\_pSKCenter = new SKCenterLib(); // 登入&環境設定物件  SKQuoteLib m\_pSKQuote = new SKQuoteLib(); // 國內報價物件 |
| Constructor |
| //dataGridViewOnNotifyOddLotSpreadDeal  {  {  dataGridViewOnNotifyOddLotSpreadDeal.Columns.Add("Column1", "市場別代號");  dataGridViewOnNotifyOddLotSpreadDeal.Columns.Add("Column2", "商品代碼");  dataGridViewOnNotifyOddLotSpreadDeal.Columns.Add("Column3", "整零成交價差(負數則含-負號)");  }  } |
| Current Block (可放在Form的Load事件裡) |
| // 事件回傳證券市場－整零價差即時行情  {  m\_pSKQuote.OnNotifyOddLotSpreadDeal += new \_ISKQuoteLibEvents\_OnNotifyOddLotSpreadDealEventHandler(OnNotifyOddLotSpreadDeal);  void OnNotifyOddLotSpreadDeal(short sMarketNo, string bstrStockNo, int nDealPrice, short sDigit)  {  double dResult = (double)nDealPrice;  while (sDigit-- != 0)  {  dResult /= 10.0;  }  bool found = false; // 找不到就新增一筆資料  foreach (DataGridViewRow row in dataGridViewOnNotifyOddLotSpreadDeal.Rows)  {  if (bstrStockNo == row.Cells[1].Value.ToString()) // 找到就更新價差  {  row.Cells[2].Value = nDealPrice;  found = true;  break;  }  }  if (found == false)  dataGridViewOnNotifyOddLotSpreadDeal.Rows.Add(sMarketNo, bstrStockNo, dResult);  }  } |

|  |
| --- |
| 成功示意畫面 |
|  |

## 五檔&成交明細

### OnNotifyBest5LONG

|  |  |  |
| --- | --- | --- |
| (LONG index)當有索取的個股五檔價格有所異動，即透過向此註冊的事件進行處理。此函式會回傳所異動的個股五檔價格。 | | |
| **事件宣告** | void OnNotifyBest5LONG([in] SHORT sMarketNo, [in] LONG nStockidx, [in] LONG nBestBid1, [in] LONG nBestBidQty1, [in] LONG nBestBid2, [in] LONG nBestBidQty2, [in] LONG nBestBid3, [in] LONG nBestBidQty3, [in] LONG nBestBid4, [in] LONG nBestBidQty4, [in] LONG nBestBid5, [in] LONG nBestBidQty5, [in] LONG nExtendBid, [in] LONG nExtendBidQty, [in] LONG nBestAsk1, [in] LONG nBestAskQty1 , [in] LONG nBestAsk2, [in] LONG nBestAskQty2, [in] LONG nBestAsk3, [in] LONG nBestAskQty3, [in] LONG nBestAsk4, [in] LONG nBestAskQty4, [in] LONG nBestAsk5, [in] LONG nBestAskQty5, [in] LONG nExtendAsk, [in] LONG nExtendAskQty, [in] LONG nSimulate); | |
| **參數** | sMarketNo | 市埸代碼。 |
| nStockidx | 系統所編的索引代碼。 |
| nBestBid1 | 第一檔買價。 |
| nBestBidQty1 | 第一檔買量。 |
| nBestBid2 | 第二檔買價。 |
| nBestBidQty2 | 第二檔買量。 |
| nBestBid3 | 第三檔買價。 |
| nBestBidQty3 | 第三檔買量。 |
| nBestBid4 | 第四檔買價。 |
| nBestBidQty4 | 第四檔買量。 |
| nBestBid5 | 第五檔買價。 |
| nBestBidQty5 | 第五檔買量。 |
| nExtendBid | 衍生一檔買價。 |
| nExtendBidQty | 衍生一檔買量。 |
| nBestAsk1 | 第一檔賣價。 |
| nBestAskQty1 | 第一檔賣量。 |
| nBestAsk2 | 第二檔賣價。 |
| nBestAskQty2 | 第二檔賣量。 |
| nBestAsk3 | 第三檔賣價。 |
| nBestAskQty3 | 第三檔賣量。 |
| nBestAsk4 | 第四檔賣價。 |
| nBestAskQty4 | 第四檔賣量。 |
| nBestAsk5 | 第五檔賣價。 |
| nBestAskQty5 | 第五檔賣量。 |
| nExtendAsk | 衍生一檔賣價。 |
| nExtendAskQty | 衍生一檔賣量。 |
| nSimulate | 0:一般揭示 1:試算揭示 |
| **相關函式** | [SKQuote\_RequestTicks](#_SKQuoteLib_RequestStocks) 相關通知事件。 | |
| **備註** | 未進行揭示處理，開發人員需自行接手處理(判斷接收的五檔報價為一般或試算揭示)。  ＊盤中零股: 因盤中仍會進入試撮，請留意是否收取『試撮資料』，即試算揭示。  ＊須使用[SKQuoteLib\_EnterMonitorLONG](#_SKQuoteLib_EnterMonitorLONG)登入，該事件才會被觸發。 | |

|  |
| --- |
| C#範例程式碼\_QuoteForm.cs |
| UI示意圖 |
| 拉出1個DataGridView，命名(Name)為「dataGridViewTicks」。  拉出1個Label，命名(Name)為「labelnSimulate」。 |
| 全域變數 |
| // 宣告物件  SKCenterLib m\_pSKCenter = new SKCenterLib(); // 登入&環境設定物件  SKQuoteLib m\_pSKQuote = new SKQuoteLib(); // 國內報價物件 |
| Constructor |
| // Ticks五檔初始化  {  dataGridViewTicks.Columns.Add("Column1", "買量");  dataGridViewTicks.Columns.Add("Column2", "買價");  dataGridViewTicks.Columns.Add("Column3", "賣價");  dataGridViewTicks.Columns.Add("Column4", "賣量");  for (int i = 0; i < 5; i++)  dataGridViewTicks.Rows.Add();  } |
| Current Block (可放在Form的Load事件裡) |
| //事件最佳五檔(國內報價)  {  m\_pSKQuote.OnNotifyBest5LONG += new \_ISKQuoteLibEvents\_OnNotifyBest5LONGEventHandler(OnNotifyBest5LONG);  void OnNotifyBest5LONG(short sMarketNo, int nStockidx, int nBestBid1, int nBestBidQty1, int nBestBid2, int nBestBidQty2, int nBestBid3, int nBestBidQty3, int nBestBid4, int nBestBidQty4, int nBestBid5, int nBestBidQty5, int nExtendBid, int nExtendBidQty, int nBestAsk1, int nBestAskQty1, int nBestAsk2, int nBestAskQty2, int nBestAsk3, int nBestAskQty3, int nBestAsk4, int nBestAskQty4, int nBestAsk5, int nBestAskQty5, int nExtendAsk, int nExtendAskQty, int nSimulate)  {  if (nSimulate == 0) labelnSimulate.Text = "一般揭示";  else if (nSimulate == 1) labelnSimulate.Text = "試算揭示";  if (dataGridViewTicks.Rows.Count > 0)  {  DataGridViewRow Row1 = dataGridViewTicks.Rows[0];  Row1.Cells[0].Value = nBestBidQty1; //買量  Row1.Cells[1].Value = nBestBid1 / 100.0; //買價  Row1.Cells[2].Value = nBestAsk1 / 100.0; //賣價  Row1.Cells[3].Value = nBestAskQty1; //賣量  DataGridViewRow Row2 = dataGridViewTicks.Rows[1];  Row2.Cells[0].Value = nBestBidQty2; //買量  Row2.Cells[1].Value = nBestBid2 / 100.0; //買價  Row2.Cells[2].Value = nBestAsk2 / 100.0; //賣價  Row2.Cells[3].Value = nBestAskQty2; //賣量  DataGridViewRow Row3 = dataGridViewTicks.Rows[2];  Row3.Cells[0].Value = nBestBidQty3; //買量  Row3.Cells[1].Value = nBestBid3 / 100.0; //買價  Row3.Cells[2].Value = nBestAsk3 / 100.0; //賣價  Row3.Cells[3].Value = nBestAskQty3; //賣量  DataGridViewRow Row4 = dataGridViewTicks.Rows[3];  Row4.Cells[0].Value = nBestBidQty4; //買量  Row4.Cells[1].Value = nBestBid4 / 100.0; //買價  Row4.Cells[2].Value = nBestAsk4 / 100.0; //賣價  Row4.Cells[3].Value = nBestAskQty4; //賣量  DataGridViewRow Row5 = dataGridViewTicks.Rows[4];  Row5.Cells[0].Value = nBestBidQty5; //買量  Row5.Cells[1].Value = nBestBid5 / 100.0; //買價  Row5.Cells[2].Value = nBestAsk5 / 100.0; //賣價  Row5.Cells[3].Value = nBestAskQty5; //賣量  }  }  } |

|  |
| --- |
| 成功示意畫面 |
|  |

### OnNotifyTicksLONG

|  |  |  |
| --- | --- | --- |
| (LONG index)當有索取的個股成交明細有所異動，即透過向此註冊事件回傳所異動的個股成交明細。 | | |
| **宣告** | void OnNotifyTicksLONG([in] SHORT sMarketNo, [in] LONG nIndex, [in] LONG nPtr, [in] LONG nDate, [in] LONG nTimehms, [in] LONG nTimemillismicros, [in] LONG nBid, [in] LONG nAsk, [in] LONG nClose, [in] LONG nQty, [in] LONG nSimulate); | |
| **參數** | sMarketNo | 報價有異動的商品市場別。 |
| nIndex | 系統所編的索引代碼。 |
| nPtr | 表示資料的位址(Key) |
| nDate | 交易日期。(YYYYMMDD) |
| nTimehms | 時間1。(時：分：秒) |
| nTimemillismicros | 時間2。(‘毫秒"微秒) |
| nBid | 買價。 |
| nAsk | 賣價。 |
| nClose | 成交價。 |
| nQty | 成交量。 |
| nSimulate | 0:一般揭示 1:試算揭示 |
| **備註** | * + 1. nTimehms提供單位：時分秒hh:mm:ss。     2. nTimemillismicros提供單位：毫秒微秒’millissecond’’micorsecond，預設提供毫秒微秒，開發人員自行決定是否顯示。(目前solace只提供證券商品)     3. 未進行揭示處理，開發人員需自行接手處理(判斷收的tick 為一般或試算揭示)。     4. T盤切換T+1盤，不保留前一盤資料。開發者需自行清除前一盤資料。     5. 避免在[SKQuoteLib\_OnNotifyTicksLONG](#_OnNotifyTicksLONG_1)通知事件裡進行[SKQuoteLib\_GetTickLONG](#_SKQuoteLib_GetTickLONG)()、[SKQuoteLib\_GetStockByIndexLONG](#_SKQuoteLib_GetStockByIndexLONG)(只可取得基本資料)   \*依國內或海外市場，可能需搭配MarketNo  ＊須使用[SKQuoteLib\_EnterMonitorLONG](#_SKQuoteLib_EnterMonitorLONG)登入，該事件才會被觸發。  一張含有 文字, 螢幕擷取畫面, 字型, 電子藍 的圖片  自動產生的描述 | |

|  |
| --- |
| C#範例程式碼\_QuoteForm.cs |
| UI示意圖 |
| 拉出1個DataGridView，命名(Name)為「dataGridViewOnNotifyTicksLONG」。 |
| 全域變數 |
| // 宣告物件  SKCenterLib m\_pSKCenter = new SKCenterLib(); // 登入&環境設定物件  SKQuoteLib m\_pSKQuote = new SKQuoteLib(); // 國內報價物件 |
| Constructor |
| // dataGridViewOnNotifyTicksLONG  {  dataGridViewOnNotifyTicksLONG.Columns.Add("Column1", "交易日期");  dataGridViewOnNotifyTicksLONG.Columns.Add("Column2", "時：分：秒");  dataGridViewOnNotifyTicksLONG.Columns.Add("Column3", "毫秒’微秒");  dataGridViewOnNotifyTicksLONG.Columns.Add("Column4", "買價");  dataGridViewOnNotifyTicksLONG.Columns.Add("Column5", "賣價");  dataGridViewOnNotifyTicksLONG.Columns.Add("Column6", "成交價");  dataGridViewOnNotifyTicksLONG.Columns.Add("Column7", "成交量");  for (int i = 0; i < 1; i++)  dataGridViewOnNotifyTicksLONG.Rows.Add();  } |
| Current Block (可放在Form的Load事件裡) |
| //(LONG index)當有索取的個股成交明細有所異動，即透過向此註冊事件回傳所異動的個股成交明細  {  m\_pSKQuote.OnNotifyTicksLONG += new \_ISKQuoteLibEvents\_OnNotifyTicksLONGEventHandler(OnNotifyTicksLONG);  void OnNotifyTicksLONG(short sMarketNo, int nIndex, int nPtr, int nDate, int nTimehms, int nTimemillismicros, int nBid, int nAsk, int nClose, int nQty, int nSimulate)  {  if (dataGridViewOnNotifyTicksLONG.Rows.Count > 0)  {  DataGridViewRow Row1 = dataGridViewOnNotifyTicksLONG.Rows[0];  Row1.Cells[0].Value = nDate; // 交易日期。(YYYYMMDD)  Row1.Cells[1].Value = nTimehms; // 時間1。(時：分：秒)  Row1.Cells[2].Value = nTimemillismicros; // 時間2。(‘毫秒"微秒)  Row1.Cells[3].Value = nBid / 100.0; // 買價  Row1.Cells[4].Value = nAsk / 100.0; // 賣價  Row1.Cells[5].Value = nClose / 100.0; // 成交價  Row1.Cells[6].Value = nQty / 100.0; // 成交量  }  }  } |

|  |
| --- |
| 成功示意畫面 |
|  |

#### OnNotifyHistoryTicksLONG

|  |  |  |
| --- | --- | --- |
| (LONG index)當**首次**索取個股成交明細，此事件會**回補當天Tick**。 | | |
| **宣告** | void OnNotifyHistoryTicksLONG([in] SHORT sMarketNo, [in] LONG nIndex, [in] LONG nPtr, [in] LONG nDate, [in] LONG nTimehms, [in] LONG nTimemillismicros, [in] LONG nBid, [in] LONG nAsk, [in] LONG nClose, [in] LONG nQty, [in] LONG nSimulate); | |
| **參數** | sMarketNo | 報價有異動的商品市場別。 |
| nIndex | 系統所編的索引代碼。 |
| nPtr | 表示資料的位址(Key) |
| nDate | 交易日期。(YYYYMMDD) |
| nTimehms | 時間1。(時：分：秒’毫秒"微秒) |
| nTimemillismicros | 時間2。(’毫秒"微秒) |
| nBid | 買價。 |
| nAsk | 賣價。 |
| nClose | 成交價。 |
| nQty | 成交量。 |
| nSimulate | 0:一般揭示 1:試算揭示 |
| **備註** | * + 1. nTimehms提供單位：時分秒hh:mm:ss。     2. nTimemillismicros提供單位：毫秒微秒’millissecond’’micorsecond，預設提供毫秒微秒，開發人員自行決定是否顯示。(目前solace只提供證券商品)     3. 未進行揭示處理，開發人員需自行接手處理(判斷收的tick 為一般或試算揭示)。     4. T盤切換T+1盤，不保留前一盤資料，開發者需自行清除前一盤資料。     5. 避免在[SKQuoteLib\_OnNotifyHistoryTicksLONG](#_OnNotifyHistoryTicksLONG)通知事件裡進行[SKQuoteLib\_GetTickLONG](#_SKQuoteLib_GetTickLONG)()、[SKQuoteLib\_GetStockByIndexLONG](#_SKQuoteLib_GetStockByIndexLONG)(只可取得基本資料)     6. 盤中零股: 因盤中仍會進入試撮，請留意是否收取『試撮資料』，即試算揭示.   \*依國內或海外市場，可能需搭配MarketNo  ＊須使用[SKQuoteLib\_EnterMonitorLONG](#_SKQuoteLib_EnterMonitorLONG)登入，該事件才會被觸發。 | |

## 技術分析

### OnNotifyMACDLONG

|  |  |  |
| --- | --- | --- |
| (LONG index)事件回傳證券市場－技術分析平滑異同平均線MACD數值。（日線－完整） | | |
| **宣告** | void OnNotifyMACDLONG([in] SHORT sMarketNo, [in] LONG nStockidx, [in] BSTR bstrMACD, [in] BSTR bstrDIF, [in] BSTR bstrOSC); | |
| **參數** | sMarketNo | 市場別代號 |
| nStockidx | 系統所編的索引代碼。 |
| bstrMACD | MACD平滑異同平均線 |
| bstrDIF | DIF |
| bstrOSC | DIF-MACD |
| **備註** | 目前只提供證券市場(不含盤中零股)  ＊須使用[SKQuoteLib\_EnterMonitorLONG](#_SKQuoteLib_EnterMonitorLONG)登入，該事件才會被觸發。 | |

|  |
| --- |
| C#範例程式碼\_QuoteForm.cs |
| UI示意圖 |
| 拉出1個richTextBox，命名(Name)為「richTextBoxMACD」。  一張含有 Rectangle, 白色, 螢幕擷取畫面, 文字 的圖片  自動產生的描述 |
| 全域變數 |
| // 宣告物件  SKCenterLib m\_pSKCenter = new SKCenterLib(); // 登入&環境設定物件  SKQuoteLib m\_pSKQuote = new SKQuoteLib(); // 國內報價物件 |

|  |
| --- |
| Current Block (可放在Form的Load事件裡) |
| //(LONG index)事件回傳證券市場－技術分析平滑異同平均線MACD數值。（日線－完整）  {  m\_pSKQuote.OnNotifyMACDLONG += new \_ISKQuoteLibEvents\_OnNotifyMACDLONGEventHandler(OnNotifyMACDLONG);  void OnNotifyMACDLONG(short sMarketNo, int nStockidx, string bstrMACD, string bstrDIF, string bstrOSC)  {  // (LONG index)取得商品技術指標MACD資訊。(平滑異同平均線)  SKMACD pSKMACD = new SKMACD();  pSKMACD.bstrStockNo = textBoxbstrStockNo.Text;  pSKMACD.bstrMACD = bstrMACD;  pSKMACD.bstrDIF = bstrDIF;  pSKMACD.bstrOSC = bstrOSC;  int nCode = m\_pSKQuote.SKQuoteLib\_GetMACDLONG(sMarketNo, nStockidx, ref pSKMACD);  // 取得回傳訊息  string msg = "【SKQuoteLib\_GetMACDLONG】" + m\_pSKCenter.SKCenterLib\_GetReturnCodeMessage(nCode);  richTextBoxMessage.AppendText(msg + "\n");  richTextBoxMACD.AppendText("【OnNotifyMACDLONG】" + "市場別代號:" + sMarketNo + "系統所編的索引代碼" + nStockidx + "MACD平滑異同平均線:" + pSKMACD.bstrMACD + "DIF:" + pSKMACD.bstrDIF + "OSC:" + pSKMACD.bstrOSC);  }  } |

|  |
| --- |
| 成功示意畫面 |
| 一張含有 文字, 螢幕擷取畫面, 字型 的圖片  自動產生的描述 |

### OnNotifyBoolTunelLONG

|  |  |  |
| --- | --- | --- |
| (LONG index)事件回傳技術分析－布林通道。（日線－完整） | | |
| **宣告** | void OnNotifyBoolTunelLONG([in] SHORT sMarketNo, [in] LONG nStockidx, [in] BSTR bstrAVG, [in] BSTR bstrUBT, [in] BSTR bstrLBT); | |
| **參數** | sMarketNo | 市場別代號 |
| nStockidx | 系統所編的索引代碼。 |
| bstrAVG | 均線 |
| bstrUBT | 通道上端 |
| bstrLBT | 通道下端 |
| **備註** | 目前只提供證券市場(不含盤中零股)  ＊須使用[SKQuoteLib\_EnterMonitorLONG](#_SKQuoteLib_EnterMonitorLONG)登入，該事件才會被觸發。 | |

|  |
| --- |
| C#範例程式碼\_QuoteForm.cs |
| UI示意圖 |
| 拉出1個richTextBox，命名(Name)為「richTextBoxBoolTunel」。  一張含有 Rectangle, 白色, 螢幕擷取畫面, 文字 的圖片  自動產生的描述 |
| 全域變數 |
| // 宣告物件  SKCenterLib m\_pSKCenter = new SKCenterLib(); // 登入&環境設定物件  SKQuoteLib m\_pSKQuote = new SKQuoteLib(); // 國內報價物件 |

|  |
| --- |
| Current Block (可放在Form的Load事件裡) |
| //(LONG index)事件回傳技術分析－布林通道。（日線－完整）  {  m\_pSKQuote.OnNotifyBoolTunelLONG += new \_ISKQuoteLibEvents\_OnNotifyBoolTunelLONGEventHandler(OnNotifyBoolTunelLONG);  void OnNotifyBoolTunelLONG(short sMarketNo, int nStockidx, string bstrAVG, string bstrUBT, string bstrLBT)  {  // (LONG index)取得商品BoolTunel資訊  SKBoolTunel pBoolTunel = new SKBoolTunel();  pBoolTunel.bstrStockNo = textBoxbstrStockNo.Text;  pBoolTunel.bstrAVG = bstrAVG;  pBoolTunel.bstrUBT = bstrUBT;  pBoolTunel.bstrLBT = bstrLBT;  int nCode = m\_pSKQuote.SKQuoteLib\_GetBoolTunelLONG(sMarketNo, nStockidx, ref pBoolTunel);  // 取得回傳訊息  string msg = "【SKQuoteLib\_GetBoolTunelLONG】" + m\_pSKCenter.SKCenterLib\_GetReturnCodeMessage(nCode);  richTextBoxMessage.AppendText(msg + "\n");  richTextBoxBoolTunel.AppendText("【OnNotifyBoolTunelLONG】" + "市場別代號:" + sMarketNo + "系統所編的索引代碼" + nStockidx + "均線:" + pBoolTunel.bstrAVG + "通道上端:" + pBoolTunel.bstrUBT + "通道下端:" + pBoolTunel.bstrLBT);  }  } |

|  |
| --- |
| 成功示意畫面 |
| 一張含有 文字, 螢幕擷取畫面, 字型 的圖片  自動產生的描述 |

### OnNotifyKLineData

|  |  |  |
| --- | --- | --- |
| 事件回傳技術分析資訊。 | | |
| **宣告** | void OnNotifyKLineData([in] BSTR bstrStockNo, [in] BSTR bstrData); | |
| **參數** | bstrStockNo | 商品代號。 |
| bstrData | 回傳字串，技術分析資料。傳回字串格式分成四種:  (1.) 舊版輸出格式，1分鐘線。以逗號分開資料。  ( 月/日/年, 時:分, 開盤價, 最高價, 最低價, 收盤價, 成交量 )  例如，06/18/2009, 09:05, 1365, 1385, 1365, 1380, 656  (2.) 舊版輸出格式，日線288天，完整日線，週線，月線。以逗號分開資料。  ( 月/日/年, 開盤價, 最高價, 最低價, 收盤價, 成交量 )  例如，06/18/2009, 1365, 1385, 1365, 1380, 656  舊版輸出函式中所取回的價格都未經過小數點處理，例如1101 價格為「36.50」，則函式所傳回的價格為「3650」，因此使用者所開發的程式則必須接手處理。  特別注意舊版輸出格式中的期匯率商品(TypeNo=209)小位數有四位須特別再處理，例如RH1704價格為「6.8712」，則函式所傳回的價格為「68712」  (3.) 新版輸出格式，1分鐘線。以逗號分開資料。  (年/月/日, 時:分, 開盤價, 最高價, 最低價, 收盤價, 成交量)  例如，商品代號(1108)  2021/8/2 09:01 12.95,12.95,12.95.12.95,5  (4.) 新版輸出格式，日線288天，完整日線，週線，月線。以逗號分開資料。  (年/月/日, 開盤價, 最高價, 最低價, 收盤價, 成交量 )  例如，商品代號(1108)  2021/8/2, 12.95,13.00,12.80,12.95,291  新版輸出函式中所取回的價格已進行過小數點處理，且由Solace K線主機提供． |
| **備註** | 證券-日K:成交量為整股交易，不含【鉅額交易、盤中零股、盤後零股】 | |

比較表:

|  |  |  |
| --- | --- | --- |
| 修改前 | 說明 | V2.13.45 |
| 上市股票(商品)代號。 | bstrStockNo | 商品代號 |
| N/A | 備註 | 日K:成交量為整股交易，不含【鉅額交易、盤中零股、盤後零股】 |

|  |
| --- |
| C#範例程式碼\_QuoteForm.cs |
| UI示意圖 |
| 拉出1個DataGridView，命名(Name)為「dataGridViewKLine」。 |
| 全域變數 |
| // 宣告物件  SKCenterLib m\_pSKCenter = new SKCenterLib(); // 登入&環境設定物件  SKQuoteLib m\_pSKQuote = new SKQuoteLib(); // 國內報價物件 |
| Constructor |
| // dataGridViewKLine  {  dataGridViewKLine.Columns.Add("Column1", "月/日/年 時:分");  dataGridViewKLine.Columns.Add("Column2", "開盤價");  dataGridViewKLine.Columns.Add("Column3", "最高價");  dataGridViewKLine.Columns.Add("Column4", "最低價");  dataGridViewKLine.Columns.Add("Column5", "收盤價");  dataGridViewKLine.Columns.Add("Column6", "成交量");  } |
| Current Block (可放在Form的Load事件裡) |
| //事件回傳技術分析資訊  {  m\_pSKQuote.OnNotifyKLineData += new \_ISKQuoteLibEvents\_OnNotifyKLineDataEventHandler(OnNotifyKLineData);  void OnNotifyKLineData(string bstrStockNo, string bstrData)  {  // 回傳字串，技術分析資料  string[] values = new string[6];  values = bstrData.Split(',');  dataGridViewKLine.Rows.Add(values[0], values[1], values[2], values[3], values[4], values[5]);  }  } |

|  |
| --- |
| 成功示意畫面 |
|  |

## 交易資訊(期選)

### OnNotifyFutureTradeInfoLONG

|  |  |  |
| --- | --- | --- |
| (LONG index)事件回傳接收**期貨**商品的交易資訊。 | | |
| **宣告** | void OnNotifyFutureTradeInfoLONG([in]BSTR bstrStockNo, [in] SHORT sMarketNo, [in] LONG nStockidx, [in] LONG nBuyTotalCount, [in] LONG nSellTotalCount, [in] LONG nBuyTotalQty, [in] LONG nSellTotalQty, [in] LONG nBuyDealTotalCount, [in] LONG nSellDealTotalCount  ); | |
| **參數** | bstrStockNo | 商品代號 |
| sMarketNo | 市場別代號 |
| nStockidx | 系統所編的索引代碼。 |
| nBuyTotalCount | 總委託買進筆數 |
| nSellTotalCount | 總委託賣出筆數 |
| nBuyTotalQty | 總委託買進口數 |
| nSellTotalQty | 總委託賣出口數 |
| nBuyDealTotalCount | 總成交買進筆數 |
| nSellDealTotalCount | 總成交賣出筆數 |
| **備註** | nStockidx系統所編的索引代碼，透過MarketNo與 Stockidx ，可以使用 [SKQuoteLib\_GetStockByIndexLONG](#_SKQuoteLib_GetStockByIndexLONG)取得商品基本資訊  ＊須使用[SKQuoteLib\_EnterMonitorLONG](#_SKQuoteLib_EnterMonitorLONG)登入，該事件才會被觸發。 | |

|  |
| --- |
| C#範例程式碼\_QuoteForm.cs |
| UI示意圖 |
| 拉出1個DataGridView，命名(Name)為「dataGridViewOnNotifyFutureTradeInfoLONG」。  一張含有 螢幕擷取畫面, Rectangle 的圖片  自動產生的描述 |
| 全域變數 |
| // 宣告物件  SKCenterLib m\_pSKCenter = new SKCenterLib(); // 登入&環境設定物件  SKQuoteLib m\_pSKQuote = new SKQuoteLib(); // 國內報價物件 |
| Constructor |
| // OnNotifyFutureTradeInfoLONG  {  dataGridViewOnNotifyFutureTradeInfoLONG.Columns.Add("Column1", "總委託買進筆數");  dataGridViewOnNotifyFutureTradeInfoLONG.Columns.Add("Column2", "總委託賣出筆數");  dataGridViewOnNotifyFutureTradeInfoLONG.Columns.Add("Column3", "總委託買進口數");  dataGridViewOnNotifyFutureTradeInfoLONG.Columns.Add("Column4", "總委託賣出口數");  dataGridViewOnNotifyFutureTradeInfoLONG.Columns.Add("Column5", "總成交買進筆數");  dataGridViewOnNotifyFutureTradeInfoLONG.Columns.Add("Column6", "總成交賣出筆數");  for (int i = 0; i < 1; i++)  {  dataGridViewOnNotifyFutureTradeInfoLONG.Rows.Add();  }  } |

|  |
| --- |
| Current Block (可放在Form的Load事件裡) |
| //(LONG index)事件回傳接收期貨商品的交易資訊  {  m\_pSKQuote.OnNotifyFutureTradeInfoLONG += new \_ISKQuoteLibEvents\_OnNotifyFutureTradeInfoLONGEventHandler(OnNotifyFutureTradeInfoLONG);  void OnNotifyFutureTradeInfoLONG(string bstrStockNo, short sMarketNo, int nStockidx, int nBuyTotalCount, int nSellTotalCount, int nBuyTotalQty, int nSellTotalQty, int nBuyDealTotalCount, int nSellDealTotalCount)  {  if (dataGridViewOnNotifyFutureTradeInfoLONG.Rows.Count > 0)  {  DataGridViewRow Row1 = dataGridViewOnNotifyFutureTradeInfoLONG.Rows[0];  Row1.Cells[0].Value = nBuyTotalCount; //總委託買進筆數  Row1.Cells[1].Value = nSellTotalCount; //總委託賣出筆數  Row1.Cells[2].Value = nBuyTotalQty; //總委託買進口數  Row1.Cells[3].Value = nSellTotalQty; //總委託賣出口數  Row1.Cells[4].Value = nBuyDealTotalCount; //總成交買進筆數  Row1.Cells[5].Value = nSellDealTotalCount; //總成交賣出筆數  }  }  } |

|  |
| --- |
| 成功示意畫面 |
|  |

### OnNotifyStrikePrices

|  |  |  |
| --- | --- | --- |
| 選擇權資訊。透過呼叫[SKQuoteLib\_GetStrikePrices](#_SKQuoteLib_GetStrikePrices)後，資訊由該事件回傳。 | | |
| **宣告** | void OnNotifyStrikePrices([in] BSTR bstrOptionData); | |
| **參數** | bstrData | 回傳的字串內容，以「,」分隔每一個欄位，欄位依序為：  [選擇權商品代碼],[選擇權中文名],[Call商品買賣權代碼],[Put商品買賣權代碼],**[履約價]**,[年+月],[最後交易日]  ex: TXO,台選,TXO11000H8,TXO11000T8,**1100000**,201808,20180815   |  |  | | --- | --- | | 0 | Product商品代碼。  例如，台選商品代碼為 TXO，也就是選擇權代號前三碼。 | | 1 | Name中文名稱。例如，台選。 | | 2 | Call Call 商品買賣權代碼。例如，TXO08100C7 | | 3 | Put Put 商品買賣權代碼。例如，TXO08100O7 | | 4 | StrikePrice 履約價。  例如，StrikePrice資料為810000，使用時要除一百，即StrikePrice/100。 | | 5 | YearMonth 年＋月。  例如，2017年3月，YearMonth資料為201703 | | 6 | LastTradeDay 最後交易日  例如，2018年7月19日，資料為20180719  當為0時代表沒有最後交易日或為不可交易商品。 | |
| **備註** | 當全部資料已經全部回傳完畢，將回傳一筆以「##」開頭的內容，表示查詢結束。 | |

|  |
| --- |
| C#範例程式碼\_QuoteForm.cs |
| UI示意圖 |
| 拉出1個DataGridView，命名(Name)為「dataGridViewOnNotifyStrikePrices」。 |
| 全域變數 |
| // 宣告物件  SKCenterLib m\_pSKCenter = new SKCenterLib(); // 登入&環境設定物件  SKQuoteLib m\_pSKQuote = new SKQuoteLib(); // 國內報價物件 |
| Constructor |
| //dataGridViewOnNotifyStrikePrices  {  {  dataGridViewOnNotifyStrikePrices.Columns.Add("Column1", "選擇權商品代碼");  dataGridViewOnNotifyStrikePrices.Columns.Add("Column2", "中文名稱");  dataGridViewOnNotifyStrikePrices.Columns.Add("Column3", "Call 商品買賣權代碼");  dataGridViewOnNotifyStrikePrices.Columns.Add("Column4", "Put 商品買賣權代碼");  dataGridViewOnNotifyStrikePrices.Columns.Add("Column5", "履約價");  dataGridViewOnNotifyStrikePrices.Columns.Add("Column6", "年＋月");  dataGridViewOnNotifyStrikePrices.Columns.Add("Column7", "最後交易日");  }  } |
| Current Block (可放在Form的Load事件裡) |
| //選擇權資訊。透過呼叫 GetStrikePrices 後，資訊由該事件回傳  {  m\_pSKQuote.OnNotifyStrikePrices += new \_ISKQuoteLibEvents\_OnNotifyStrikePricesEventHandler(OnNotifyStrikePrices);  void OnNotifyStrikePrices(string bstrOptionData)  {  string[] values = bstrOptionData.Split(',');  dataGridViewOnNotifyStrikePrices.Rows.Add(values[0], values[1], values[2], values[3], values[4], values[5], values[6]);  }  } |

|  |
| --- |
| 成功示意畫面 |
|  |

# Struct結構物件

## SKSTOCKLONG (國內報價商品物件 LONG index)

struct SKSTOCKLONG

{

LONG nStockidx; // 商品自定索引代號

SHORT sDecimal; // 小數位數

SHORT sTypeNo; // EX: (證券)類股別 1 水泥 , 2 食品…etc.

BSTR bstrMarketNo; // 市埸代碼

BSTR bstrStockNo; // 商品代碼EX: 1101 台泥, TX12 台指期12月…etc.

BSTR bstrStockName; // 商品名稱

LONG nHigh; // 最高價

LONG nOpen; // 開盤價

LONG nLow; // 最低價

LONG nClose; // 成交價

LONG nTickQty; // 單量

LONG nRef; // 昨收、參考價

LONG nBid; // 買價

LONG nBc; // 買量

LONG nAsk; // 賣價

LONG nAc; // 賣量

LONG nTBc; // 買盤量(即外盤量)

LONG nTAc; // 賣盤量(即內盤量)

LONG nFutureOI; // 期貨未平倉 OI

LONG nTQty; // 總量

LONG nYQty; // 昨量

LONG nUp; // 漲停價

LONG nDown; // 跌停價

LONG nSimulate; // 揭示 0:一般 1:試算　＊

　　　　　　　　// [**證券逐筆**]盤中出現『1:試算』代表行情劇動，

　　　　　　　　　　　　　觸發價格穩定措施狀態

LONG nDayTrade // [**限證券整股商品**]可否當沖 0:一般 1:可先買後賣現股當沖

　　　　　　　　　　　　　2:可先買後賣和先賣後買現股當沖

LONG nTradingDay // 交易日(YYYYMMDD)

備註 :當日非交易日時, 資料為前一交易日

LONG nDealTime //成交時間 (hhmmss)

};

### 盤中零股

＊因盤中仍會進入試撮，請留意是否收取『試撮資料』，即試算揭示.

struct SKSTOCKLONG

{

LONG nStockidx; // 商品自定索引代號

SHORT sDecimal; // 小數位數

SHORT sTypeNo; // EX: (證券)類股別 1 水泥 , 2 食品…etc.

BSTR bstrMarketNo; // 市埸代碼(0:上市, 1:上櫃, 2:期貨, 3:選擇權, 4:盤中零股-上市, 6:盤中零股-上櫃)

BSTR bstrStockNo; // 商品代碼EX: 1101 台泥, TX12 台指期12月…etc.

BSTR bstrStockName; // 商品名稱

LONG nHigh; // 最高價

LONG nOpen; // 開盤價

LONG nLow; // 最低價

LONG nClose; // 成交價

LONG nTickQty; // 單量

LONG nRef; // 昨收、參考價

LONG nBid; // 買價

LONG nBc; // 買量

LONG nAsk; // 賣價

LONG nAc; // 賣量

LONG nTBc; // 買盤量(即外盤量)

LONG nTAc; // 賣盤量(即內盤量)

LONG nTQty; // 總量

LONG nYQty; // 昨量

LONG nUp; // 漲停價

LONG nDown; // 跌停價

LONG nSimulate; // 揭示 0:一般 1:試算(試撮)　＊

　　　　　　　　// [**證券逐筆**]盤中出現『1:試算』代表行情劇動，

　　　　　　　　　　　　　觸發價格穩定措施狀態;

//[證券盤中零股] 盤中會進入試撮，請留意是否收取『試撮資料』

LONG nDayTrade // [**限證券商品**]可否當沖 0:一般

LONG nTradingDay // 交易日(YYYYMMDD)

備註 :當日非交易日時, 資料為前一交易日

LONG nTradingLotFlag //[**證券]** 整股、盤中零股揭示註記 0:現股 ; 1:盤中零股

};

## SKTICK(國內報價TICK物件)

struct SKTICK

{

LONG nPtr; //第幾筆成交明細, 由0開始

LONG nTimehms; //時間(時分秒hh:mm:ss)

LONG nTimemillismicros; //時間(毫秒微秒“ms”μs)

LONG nBid; //買價

LONG nAsk; //賣價

LONG nClose; //成交價

LONG nQty; //量

LONG nSimulate; //揭示 0:一般 1:試算（試撮）＊

LONG nDate; //交易日(YYYYMMDD)

};

＊盤中零股: 因盤中仍會進入試撮，請留意是否收取『試撮資料』，即試算揭示。

＊時間單位：毫秒（ms）：10-3秒 ；微秒（µs）：10-6秒，當nTimemillismicros資料是996886，前三位數字為（ms）10-3秒，後三位數為　（µs）10-6秒。

## SKBEST5(五檔價格物件)

typedef struct BEST5

{

LONG nBid1; // 第一檔買價

LONG nBidQty1; // 第一檔買量

LONG nBid2; // 第二檔買價

LONG nBidQty2; // 第二檔買量

LONG nBid3; // 第三檔買價

LONG nBidQty3; // 第三檔買量

LONG nBid4; // 第四檔買價

LONG nBidQty4; // 第四檔買量

LONG nBid5; // 第五檔買價

LONG nBidQty5; // 第五檔買量

LONG nExtendBid; // 衍生一檔買價{海外無衍生一檔}

LONG nExtendBidQty; // 衍生一檔買量{海外無衍生一檔}

LONG nAsk1; // 第一檔賣價

LONG nAskQty1; // 第一檔賣量

LONG nAsk2; // 第二檔賣價

LONG nAskQty2; // 第二檔賣量

LONG nAsk3; // 第三檔賣價

LONG nAskQty3; // 第三檔賣量

LONG nAsk4; // 第四檔賣價

LONG nAskQty4; // 第四檔買量

LONG nAsk5; // 第五檔賣價

LONG nAskQty5; // 第五檔賣量

LONG nExtendAsk; // 衍生一檔賣價{海外無衍生一檔}

LONG nExtendAskQty; // 衍生一檔賣量{海外無衍生一檔}

LONG nSimulate; // 揭示 0:一般 1:試算(試撮\*)

}

**\*國內行情—盤中零股：盤中仍有試撮行情，請自行判斷及揭示。**

## SKMACD(國內報價SKMACD物件)

typedef struct SKMACD

{

BSTR bstrStockNo; // 商品代碼

BSTR bstrMACD; // 技術指標MACD平滑異同平均線

BSTR bstrDIF; // 技術指標DIF

BSTR bstrOSC // 技術指標OSC

}

## SKBoolTunel (國內報價SKBoolTunel物件)

typedef struct SKBoolTunel

{

BSTR bstrStockNo; // 商品代碼

BSTR bstrAVG; // 技術指標布林通道 AVG均線

BSTR bstrUBT; // 技術指標 UBT通道上端

BSTR bstr LBT; // 技術指標 LBT通道下端

}