이트레이드증권 API(COM\_Ver)

C#

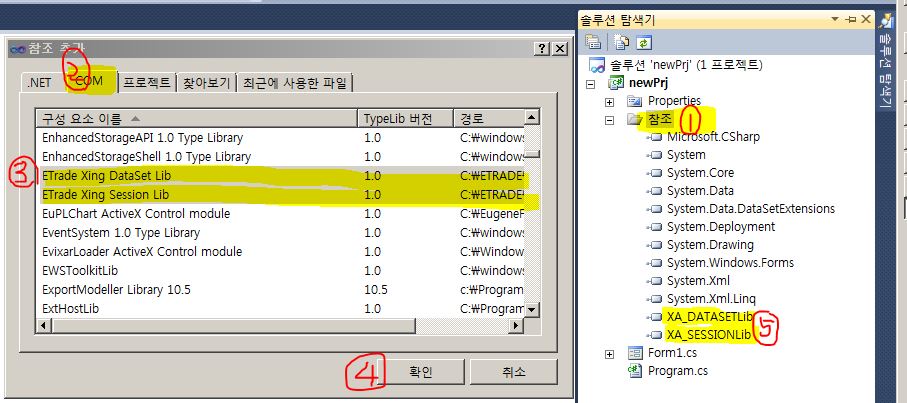
SampleCodeProject

(Kospi200지수 선물/옵션)

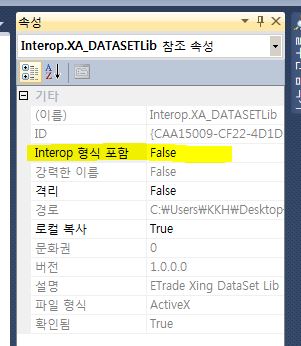
2013. 10. 22

목차

1. 환경설정
   1. 이트레이드 API-----------------------------3
   2. DevCenter for API--------------------------3
   3. 개발환경(Microsoft VisualStudio 2010)-----4
2. 화면구성 및 기능
   1. MainForm 및 Menu-------------------------5
   2. 로그인----------------------------------------6
   3. 계좌-------------------------------------------6
   4. 주문-------------------------------------------9
   5. 시세------------------------------------------11
3. Class구성 및 설명
   1. QueryClass-----------------------------------13
   2. RealClass-------------------------------------13
   3. FormClass------------------------------------14
4. 환경설정
   1. 이트레이드 API
      * + 이트레이드증권 홈페이지(<http://www.etrade.co.kr/>), 매매시스템, xingAPI 화면참조
        + 최신버전의 xingAPI 설치파일을 설치
        + C:\ETRADE\xingAPI 폴더에 관련파일 및 폴더 생성
        + C:\ETRADE\xingAPI \reg.bat 를 실행
        + C:\ETRADE\xingAPI\Res 폴더에 각Tr의 ‘\*.res’이 저장됨
        + xingAPI 사용등록
   2. DevCenter for API
      * + API 개발의 편의성을 극대화시킨 API 매매프로그램 개발의 백과사전
        + xingAPI 설치파일에 포함되어 자동 설치됨
        + 일반/실시간Tr 데이터 요청필드 및 수신필드에 대한 상세명세
   3. 개발환경(Microsoft VisualStudio 2010)
      * + <http://www.microsoft.com/visualstudio/kor/downloads#d-2010-express> 에서 ‘Visual C# 2010 Express – 한국어’ 설치
        + 새 프로젝트 생성후 솔루션 탐색기에서 참조추가
        + COM tab에서 관련 구성요소 선택후 확인



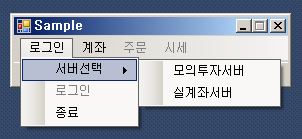
* + - * 참조 추가된 구성요소 항목(2개) 각각 오른클릭 하여 참조속성 변경 (Interop 형식 포함 : True 🡺 False)



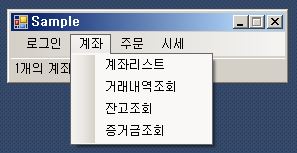
1. 화면구성 및 기능
   1. MainForm 및 Menu
      * + MianForm 화면



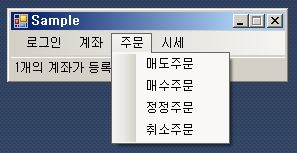
* + - * 로그인 메뉴



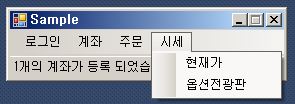
* + - * 계좌 메뉴



* + - * 주문 메뉴



* + - * 시세 메뉴



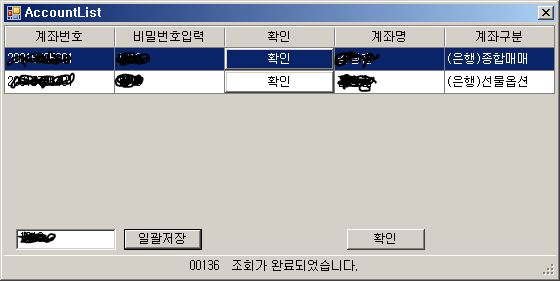
* 1. 로그인
     + - 서버선택 : 모의투자서버 및 실계좌서버의 선택에 따라 myXASessionClass.ConnectServer()호출
       - Server선택 즉시 해당 로그인Menu 실행
       - 로그인 Menu 실행시 기존에 열린 폼은 모두 닫힘



* + - * 로그인EventHandler에서 로그인이 성공적으로 수행되면 계좌Menu의 계좌리스트Menu 실행
      * 서버선택과 로그인 Menu는 이미 실행된 항목(서버 및 ID)을 바꿀수 있슴
      * 종료Menu : 프로그램을 종료 함
  1. 계좌
     + - 계좌리스트Menu : 계좌 비밀번호 설정

. 관련Tr : CACBQ21900(개별계좌정보 조회)

. 서버 선택 🡺 로그인 성공 🡺 계좌리스트Menu 실행의 과정이 일괄적으로 수행됨



. MainForm의 static Dictionary<string, List<string>> AccList

에 선물옵션 계좌정보를 등록

. 일괄 및 개별 등록가능

. 선물옵션 계좌만 관리됨

. 모의투자에서는 CACBQ21900(개별계좌정보 조회)기능이 제공되지 않으므로 절차를 무시함

* + - * 거래내역조회 Menu : HTS 6503/6603 화면참조

. 관련Tr : t0434(선물/옵션 체결/미체결)

C01(선물주문체결)

H01(선물주문정정취소)

O01(선물접수)

. 전체,체결,미체결 Vs. 조회,실시간,조회/실시간 3\*3=9종류 선택가능

. Request(t0434(선물/옵션 체결/미체결))는 1회만, 실시간 조회시 RealTr로만 갱신작업 수행함

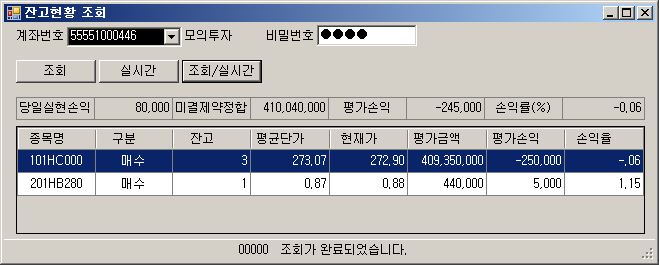
* + - * 잔고조회 Menu : HTS 6502 화면참조

. 관련Tr : t0441(선물/옵션 잔고평가(이동평균))

C01(선물주문체결)

FC0(KOSPI200선물체결)

OC0(KOSPI200옵션체결)



. 조회,실시간,조회/실시간 3종류 선택가능

. Request(t0441(선물/옵션 잔고평가(이동평균)))는 1회만, 실시간 조회시 RealTr로만 갱신작업 수행함

. 각종목의 현재가를 수신하여 평가손익 계산

. 종목 청산 체결시 실현손익 계산하여 반영

* + - * 증거금조회 Menu : HTS 6604 화면참조

. 관련Tr : CFOBQ10500 (선물옵션 계좌예탁금증거금조회) 

. 실시간 갱신기능 없슴

* 1. 주문 : HTS 8750 화면참조
     + - 관련Tr : CFOAT00100(선물옵션 정상주문) 매도/매수주문 Menu

CFOAT00200(선물옵션 정정주문) 정정주문 Menu

CFOAT00300(선물옵션 취소주문) 정정주문 Menu



* + - * 지정가 주문만 수행토록 함
      * 종목선택창(CodeSelector) : 주문대상 종목의 코드를 선택 생성

. 관련Tr : t9943 (지수선물마스터조회)

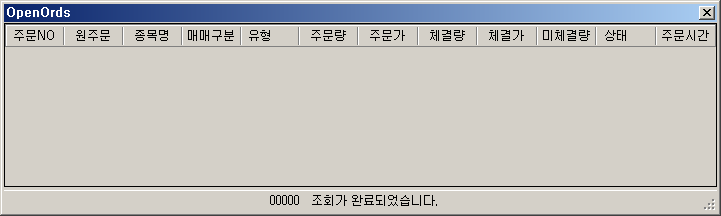
t9944 (지수옵션마스터조회)

. MainForm의 static SortedDictionary<string, List<string>> 종목마스터를 통하여 Singleton Instance로 구현



* + - * 미체결주문 선택 창(OpenOrds) : 정정/취소주문시 미체결주문 선택

. 관련Tr : t0434(선물/옵션 체결/미체결)



* 1. 시세
     + - 현재가 Menu : HTS 2106 화면참조

. 관련Tr : t2105(선물/옵션현재가호가조회)

FH0(KOSPI200선물호가)

FC0(KOSPI200선물체결)

OH0(KOSPI200옵션호가)

OC0(KOSPI200옵션체결)



* + - * 옵션전광판 Menu : HTS 2301 화면참조

. 관련Tr : t2301()

OC0(KOSPI200옵션체결)

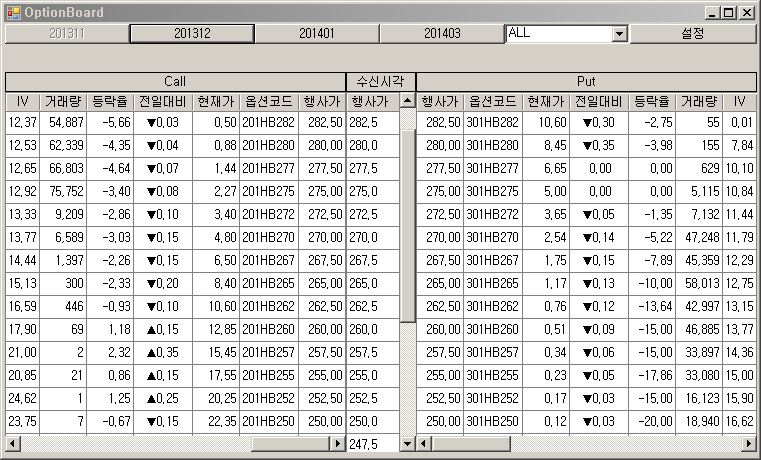
OH0(KOSPI200옵션호가)

OMG(KOSPI200옵션민감도)

.만기월별 행사가별 콜/풋 정보표시

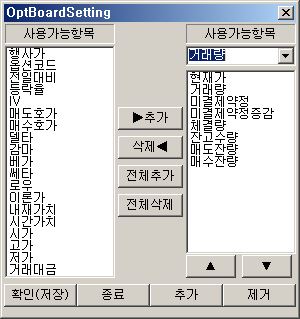
. ‘OptBoardCategories.txt’파일에 저장하며, static SortedDictionary<string, List<string>> Categories에

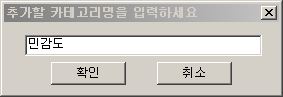
항목을 등록하여 선택하여 사용



. Categories설정(OptBoardSetting) : Category별로 사용항목을

설정 등록



. Category명 추가(AddCategory) : 추가할 Category 입력창

1. Class구성 및 설명
   1. QueryClass
      * + ‘class Query’ 로부터 상속을 받아 각 Tr별로(class Tr명)구현하여

‘QueryClass’폴더에 집결

* + - * 단일블럭의 SetFieldData를 필요로하는 Tr만 상속가능 (예외 : t1104(주식 현재가 시세메모)등)
      * delegate를 이용하여 ‘CallBackMethod’ 및 ‘CallBackMsg’로 사용class에서 ReceiveData 및 ReceiveMessage 수신/처리
      * 반드시’myXAQueryClass.ResFileName();’ 호출후에 ‘GetInBlockName();’ 호출 해야함
      * 자식class의 생성자 및 QueryExcute overload는 Thread분기시 필요
      * 상속의필요성 : override구현으로 Tr별입력항목을 intellisense로 표시하고, Data의 연속적(myXAQueryClass.IsNext) 호출시(myXAQueryClass.Request(true);) Tr별로 구현, 또한 Tr고유의 특별한 처리(예 : class t9943, t9944)를 위함
  1. RealClass
     + - ‘class Real’ 로부터 상속을 받아 각 Tr별로(class Tr명)구현하여

‘RealClass’폴더에 집결

* + - * delegate를 이용하여 ‘CallBackMethod’로 사용class에서 ReceiveRealData 수신/처리
      * 상속의필요성 : override구현으로 Tr별입력항목을 intellisense로 표시하고, Tr고유의 특별한 처리를 위함
      * ‘UnadviseRealData();’ 호출 관련하여 이전에 반드시 ‘CallBackMethod = null;’을 해주도록 함
      * QueryClass와 달리 overload구현은 없슴
      * SetFieldData의 InBlockName은 "InBlock"임을 전제로 함
  1. FormClass
     + - 각 기능별 UI의 구현
       - 역할에 따라 ‘Show();’ 와 ‘ShowDialog()’ 로 호출
       - 필요Tr은 대체적으로 QueryTr은 지역변수로, RealTr은 Private변수로 사용
       - DataGridView 컨트롤은 DataTable로써 DataSource지정하여 Data핸들링함
       - QueryClass 생성 사용시 ‘CallBackMethod’ 및 ‘CallBackMsg’에 해당 Method를 등록,

매개변수포함 생성자 또는 QueryExcute() 호출시 intellisense의 순서와 Key에 맞도록 Dictionary<> 생성전달

* + - * RealClass 생성 사용시 ‘CallBackMethod’에 해당 Method를 등록, ‘UnadviseRealData();’ 호출 관련하여 이전에 반드시 ‘CallBackMethod = null;’을 해주도록 함,

매개변수포함 AdviseExcute() 호출시 intellisense의 순서와 Key에 맞도록 Dictionary<> 생성전달

* + - * 주요 static변수

. MainForm. AccList : 비밀번호 입력 확인된 선물옵션 계좌목록

. MainForm.종목마스터 : 종목코드 및 만기, 상/하한가 관리

. MainForm. LoadedForm : 프로그램내 모든 열린(Show()) Form목록, 새로운 로그인시 모두닫기 용도

. CodeSelector. Instance : Singleton\_Instance의 유일한 접근자

. OptionBoard. Categories : 옵션전광판의 설정항목Category 관리

* + - * 로그인 성공 이후에 모든 Menu항목이 활성화 됨

새로운 로그인시 모든 열린창 닫음