

xingACE 사용자 매뉴얼

> **xingACE란?**

- 1) xingACE는 xingAPI Cyber Exchange 의 약자로 xingAPI 사용자를 위한 개인 가상 거래소입니다.
- 2) xingAPI를 이용하여 만든 프로그램이 xingACE에서 접속하여 24시간 가상체결을 받을 수 있습니다.

> **xingACE 는 언제 사용하나요?**

- 1) 24시간 가상체결을 받을 수 있습니다.**
장중에만 주문테스트가 가능한데 비해 xingACE를 사용하면 장중이 아니어도 주문테스트가 가능합니다.
- 2) 시세를 만들 수 있습니다.**
주문이 발생될 경우 해당 호가 시세도 같이 발생되며, 체결이 이루어질 경우 해당 체결 시세도 같이 발생합니다. (상대체결 참조)
- 3) 로직 시뮬레이션이 가능합니다.**
외부(이베스트투자증권 서버)로부터 xingACE로 시세를 전달해 줄 경우 거래소와 같이 주문체결을 발생시킬 수 있으며 그에 따른 시뮬레이션이 가능합니다.(시세체결 참조)

> 설치하기

- 1) xingACE는 xingAPI 패키지 프로그램에 같이 내장되어 있습니다.
- 2) 홈페이지에서 xingAPI 패키지 설치프로그램을 다운 받으시기 바랍니다.
- 3) 다운 받은 프로그램을 실행합니다.

> 접속하기

- 1) xingACE를 실행시키면 로그인 화면이 나타납니다.

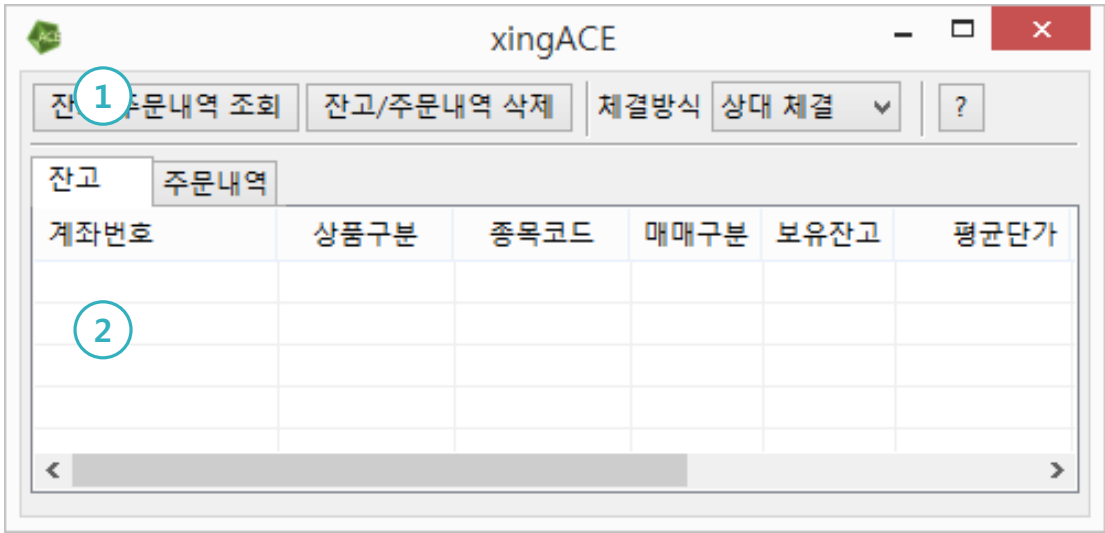


- 2) 아이디/비밀번호/공인인증비밀번호를 입력한 후에 접속을 누르시면 됩니다.

※ 주의

서버는 실서버만 가능하며 공인인증서가 있어야만 접속이 가능합니다.

> 화면 살펴보기



1) 톨바

메뉴		내용
잔고/주문내역 조회		잔고/주문내역을 조회합니다.
잔고/주문내역 삭제		잔고/주문내역을 삭제합니다.
체결방식	시세 체결	시세체결을 선택합니다.
	전량 체결	주문 즉시 전량 체결을 선택합니다.
	하프 체결	주문 즉시 1/2수량 체결을 선택합니다.
	상대 체결	상대 주문에 의한 체결을 선택합니다.
? (도움말)		도움말을 실행합니다.

2) 잔고/주문내역

- xingACE 에 접속하여 주문을 실행하면 잔고 및 주문내역 데이터가 표시됩니다.
- 잔고 및 주문내역 데이터는 PC의 DB 파일에 저장합니다.

> 사용자가 개발한 xingAPI 프로그램을 xingACE에 접속하기

Step 1 xingAPI 프로그램 실행

- ① 자신이 개발한 xingAPI 프로그램을 실행합니다.

여기서는 xingAPI 인스톨시에 배포되는 Sample_xingACE.exe 를 기준으로 설명합니다

Step 2 xingACE 서버 주소 입력

- ① xingACE가 설치된 PC의 IP를 입력하여 xingAPI 프로그램이 xingACE로 접속하도록 합니다.

같은 PC에서 실행중이라면 127.0.0.1 로 접속하면 됩니다.
127.0.0.1 은 어떤 PC이건 간에 자기 자산의 IP입니다.

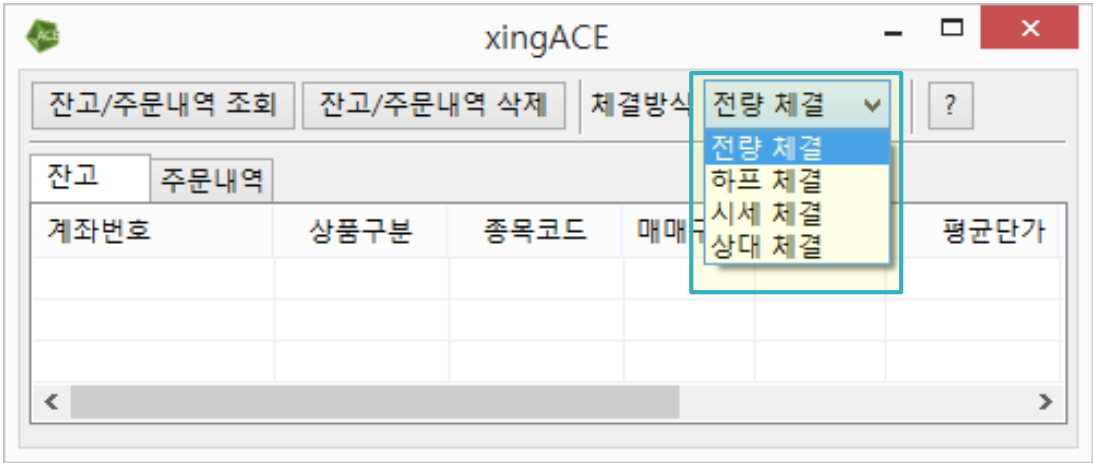
Step 3 아이디/비밀번호/공인인증 비밀번호 입력하고 로그인 하기

- ① 아이디/비밀번호/공인인증 비밀번호까지 입력을 해야 로그인이 가능합니다.

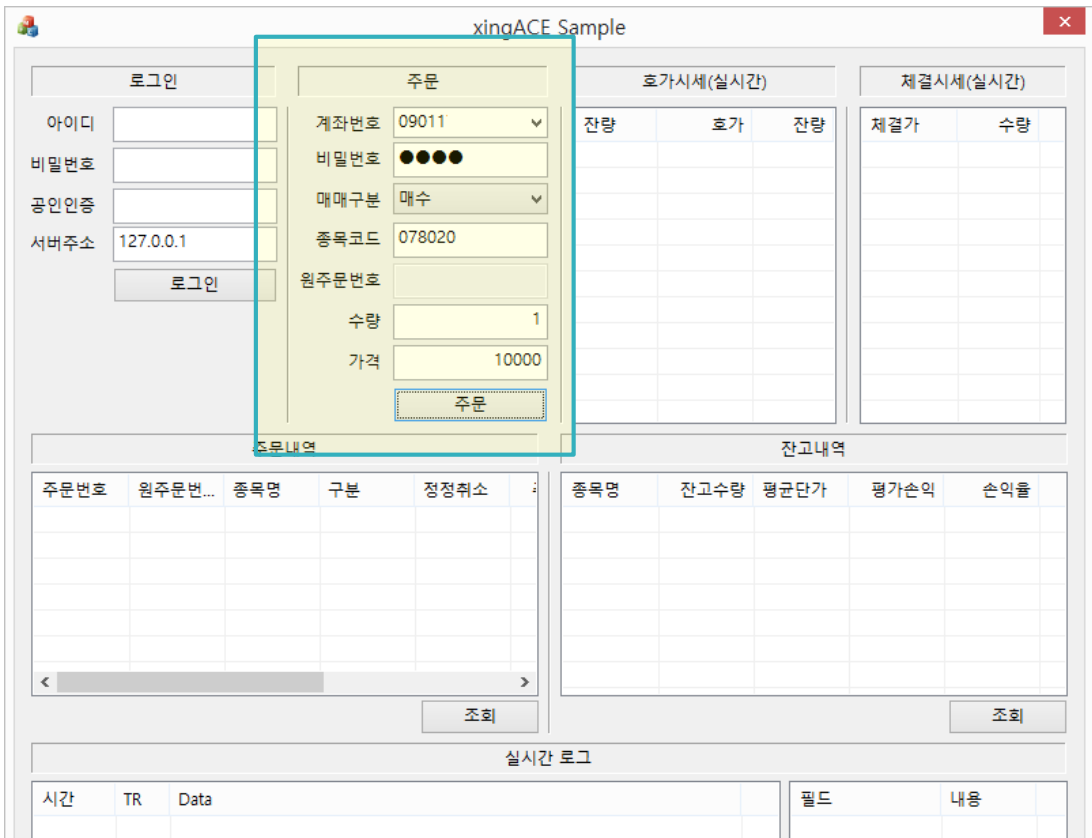
xingACE에서 로그인 한 ID로만 로그인이 가능하며 다른 아이디로는 로그인 할 수 없습니다.

> 가상체결 받아보기

Step 1 xingACE에서 체결방식을 “전량 체결”로 선택합니다.



Step 2 xingAPI를 이용해서 개발한 프로그램에서 주문을 실행합니다.



Step 3 xingAPI 프로그램에서 주문체결내역조회 및 잔고내역 조회에서 체결 및 잔고를 확인합니다.

로그인

아이디

비밀번호

공인인증

서버주소 127.0.0.1

주문

계좌번호 09011

비밀번호

주문

호가시세(실시간)

잔량

호가

잔량

체결시세(실시간)

체결가

수량

"조회" 버튼 클릭하여 주문
낸 내역을 확인합니다.
(주문내역은 실시간으로 동작
하지 않습니다.)

주문번호	원주문번호	종목명	구분	정정취소
1	0	이트레이...	매수신규	

주문낸 내역이 체결이 되었다
면 "조회" 버튼 클릭하여 잔고
내역을 확인합니다.
(잔고내역은 실시간으로 동작
하지 않습니다.)

종목명	잔고수량	평균단가	평가손익	손익율
이트레이...	1	10000.0	-200	

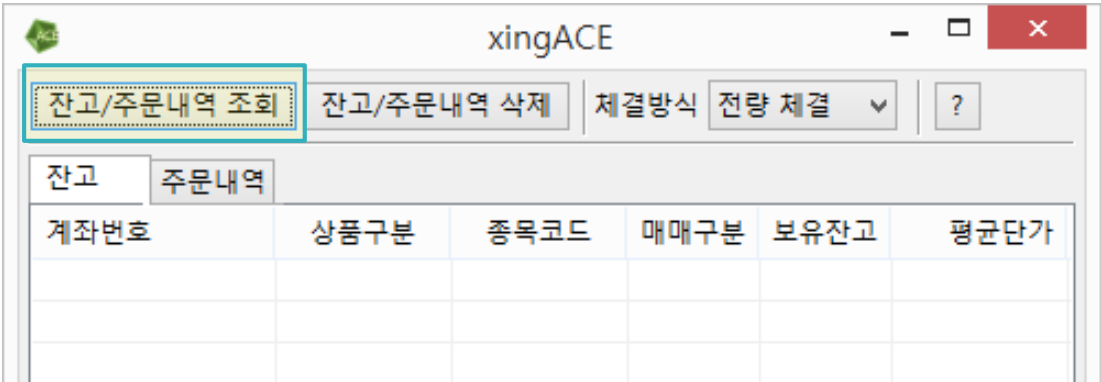
시간	TR	Data	필드	내용
15:32:15	SC1	000000000109011	B	063 HT ...
15:32:15	SC0	000000000109011	B	SONAT000063 ...

실시간 로그는 실시간 주문접수, 실시간 주문체결 등 실시간 데이터를 받았을때 실시간으로 표시하여 줍니다.

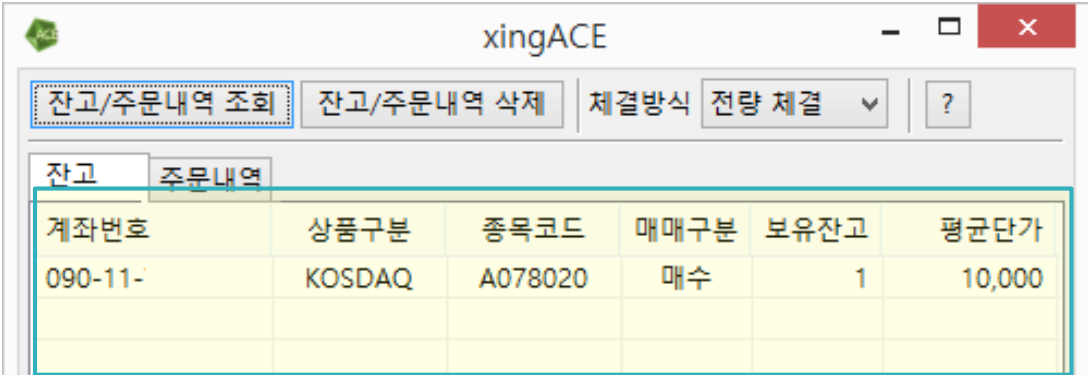
주문이 발생되면 자동으로 주문체결이 발생되어 해당 그리드에 표시됩니다. (자동체결은 xingACE에서 다른 옵션으로 변경이 가능합니다.)

Step 4 xingACE의 “잔고/주문내역 조회” 버튼을 눌러 해당 잔고 및 주문이 들어왔는지 확인합니다.

① 잔고/주문내역 조회를 클릭합니다.



② 잔고 탭을 클릭하면 잔고내역이 들어온 것을 확인할 수 있습니다.



③ 주문내역 탭을 클릭하면 잔고내역이 들어온 것을 확인할 수 있습니다.



> xingACE의 기능

1) 가상체결

- xingACE 는 [III.둘러보기]에서 보신 것처럼
- 전송된 주문에 대해 가상으로 주문을 체결시키고
 - 실시간 주문접수, 실시간 주문체결 등의 주문이 발생되어 체결이 될때까지의 데이터를
 - xingAPI를 이용하여 개발된 프로그램으로 전송해 줍니다.

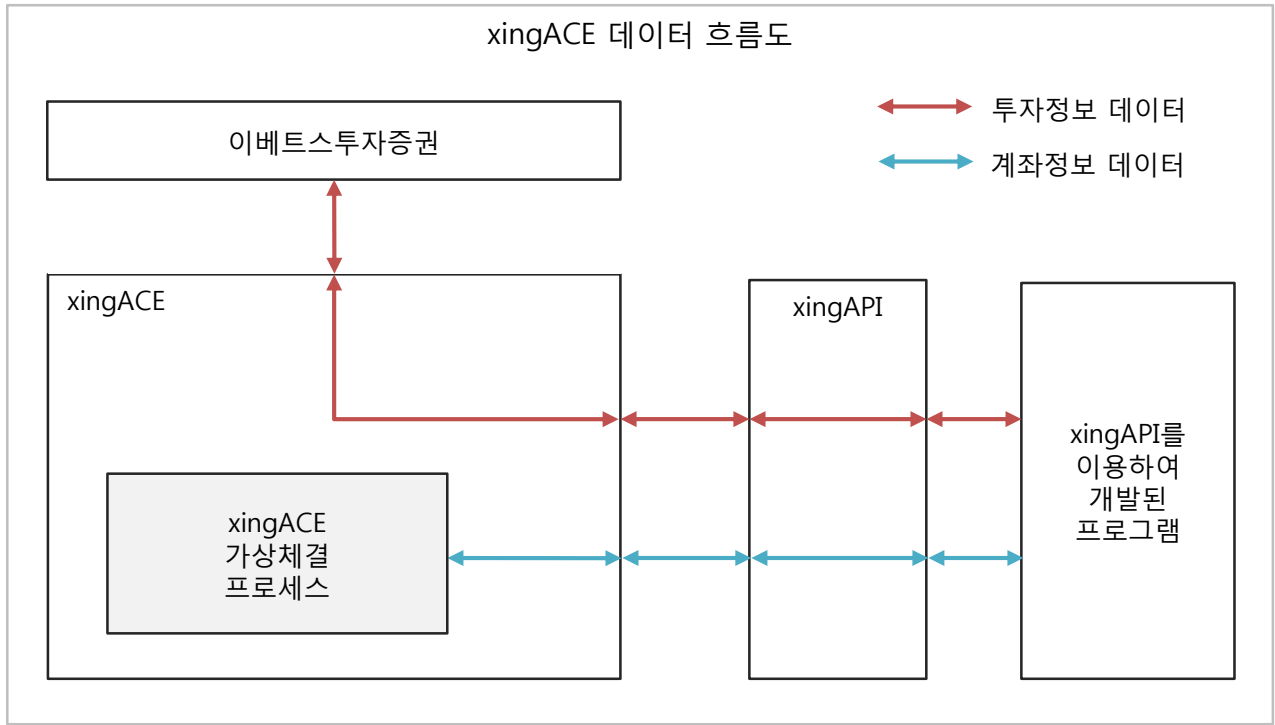
2) 주문내역 관리

xingACE는 xingAPI를 이용하여 개발된 프로그램에서 발생한 주문을 관리합니다.

3) 잔고내역 관리

xingACE는 xingAPI를 이용하여 개발된 프로그램에서 발생한 체결에 대해서 관리합니다.

> xingACE의 구조



1) 위의 Diagram에서 보여지는 바와 같이 xingACE는 xingAPI에서 들어오는 투자정보 데이터는 재가공 없이 바로 서버로 전송을 하고 서버로부터 받은 투자정보 데이터를 xingAPI로 재전송을 합니다.

> xingACE의 한계

1) 예수금

xingACE에서는 계좌정보에 대한 데이터를 이베스트투자증권 서버에서 받지 않으므로 예수금 정보가 없으며 예수금관리를 하지 않고 주문에 대한 증거금 체크도 하지 않습니다.

2) 주문가능수량

예수금이 전혀 없으므로 신규주문에 대한 주문가능수량에 대한 처리를 하지 않습니다.

3) TR에 대한 정합성 체크

TR에 대한 정합성 체크는 실서버에서 체크하는 수준으로 하지 않으므로 xingACE를 이용한 TR 정합성 체크는 불가능합니다.

4) TR 수신 데이터 정보

실서버 수준의 모든 데이터를 제공하지 않으며 가상체결에 필요한 일부 데이터만 제공합니다.

5) 에러메시지

실서버와는 다른 에러메시지를 전송합니다.

※ 유의

xingACE는 주문체결을 테스트하기 위한 시스템이며 그 외의 다른 기능은 제공하지 않거나 잘못 제공될 수 있으므로 유의하시기 바랍니다.

> 다양한 가상체결 방식 지원

1) 주문 테스트를 위해 다양한 방식의 가상체결방식을 지원합니다.

체결방식	설명
전량 체결	주문을 전송하면 무조건 주문수량 전체가 자동체결 됩니다. 개발시에 주문체결 실시간 TR을 받기 위해서 사용합니다.
하프 체결	주문을 전송하면 무조건 주문수량의 반이 자동체결되고 남은 수량은 미체결로 남게 됩니다. 주문 수량이 단위수량일 경우에는 전량 체결이 됩니다. 개발시에 정정/취소 실시간 TR을 받기 위해서 사용합니다.
시세 체결	모의투자 체결방식처럼 장중에 거래소로부터 들어오는 시세 중에 같은 가격의 체결이 있을 경우 체결됩니다. 전략을 점검하기 위해 사용합니다.
상대 체결	거래소 방식처럼 상대 호가 잔량이 있어야 체결됩니다. 같은 계좌 혹은 다른 계좌로 미리 주문을 전송한 후에 상대 가격으로 주문을 전송해야 체결이 발생합니다. 주문 혹은 체결이 발생하면 해당 시세를 받을 수 있습니다. 특정 시세를 만들거나 거래소 같은 체결을 원할 때 사용합니다.

xingACE Sample

1) xingACE Sample은 xingACE를 설명하기 위해 xingAPI를 이용해 개발된 프로그램이며 일반적인 기능 보다는 xingACE를 위한 기능만을 탑재하였습니다.

로그인

아이디

비밀번호

공인인증

로그인 영역
xingACE와의 연결을 위해 서버주소는 127.0.0.1을 입력해야 합니다.

주문

계좌번호09011

비밀번호

매매구분매수

종목코드078020

원주문번호

수량1

주문 영역

주문

호가시세(실시간)

잔량

호가

잔량

체결시세(실시간)

체결가

수량

호가시세 및 체결시세
조회는 제공되지 않으며 실시간만 표시됩니다.
주문이 전송되면 해당 종목의 실시간 호가 시세가 표시됩니다.
가상체결 중 상대체결일 경우 xingACE에서 전송되는 시세를 표시하기 위해 사용됩니다.

주문내역

주문번호

원주문번호

종목명

구분

정정취소

잔고내역

종목명

잔고수량

평균단가

평가손익

손익율

주문/잔고내역
실시간은 지원되지 않습니다.
xingACE로 전송한 주문의 내역을 보기위해 사용됩니다.

조회

조회

실시간 로그

시간

TR

Data

필드

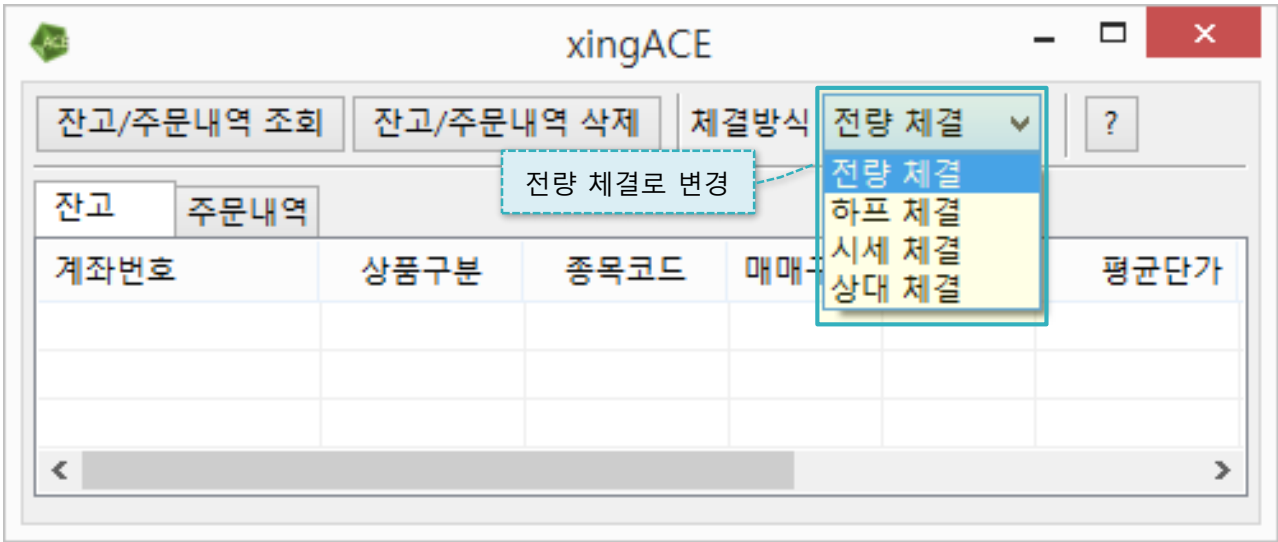
내용

실시간 로그
실시간 주문과 실시간 시세를 받은 내역을 표시합니다.

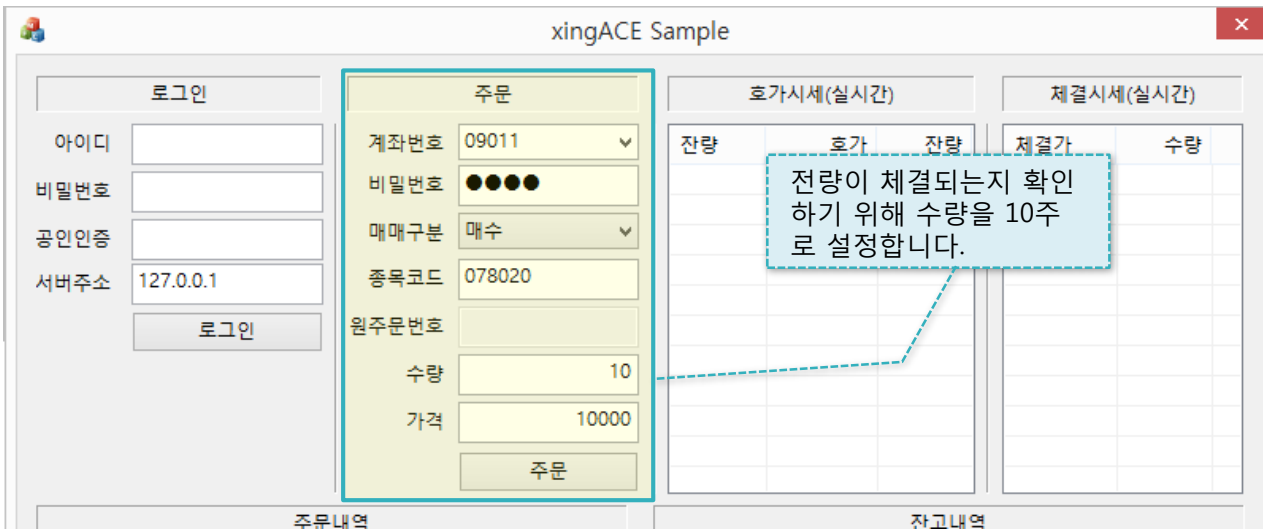
> 전량 체결

주문을 실행한 즉시 주문수량 전체가 체결되는 방식입니다.

Step 1 xingACE 에서 체결방식을 “전량 체결”로 변경합니다.



Step 2 xingAPI 로 개발한 프로그램에서 주문을 실행합니다.



Step 3 주문수량이 전량 체결된것을 확인합니다.

로그인

아이디

비밀번호

공인인증

서버주소

로그인

주문

계좌번호

비밀번호

매매구분

종목코드

원주문번호

수량

가격

주문

호가시세(실시간)

잔량

호가

잔량

체결시세(실시간)

체결가

수량

주문내역

종목명	구분	주문수량	주문가격	체결수량	체결가
이트레이...	매수신규	10	100.00	10	100.00

10 주 체결되었습니다.

잔고내역

종목명	잔고수량	평균단가	평가손익	손익율
이트레이...	10	10000.0	-2000	

잔고가 10 주 되었습니다.

실시간 로그

시간	TR	Data
18:01:34	SC1	0000000001090 B 063 HT ...
18:01:34	SC0	0000000001090 B SONAT000063 ...

1. 실시간 주문접수(SC0)
2. 실시간 주문체결(SC1)
순으로 받은 것을 확인 할 수 있습니다.

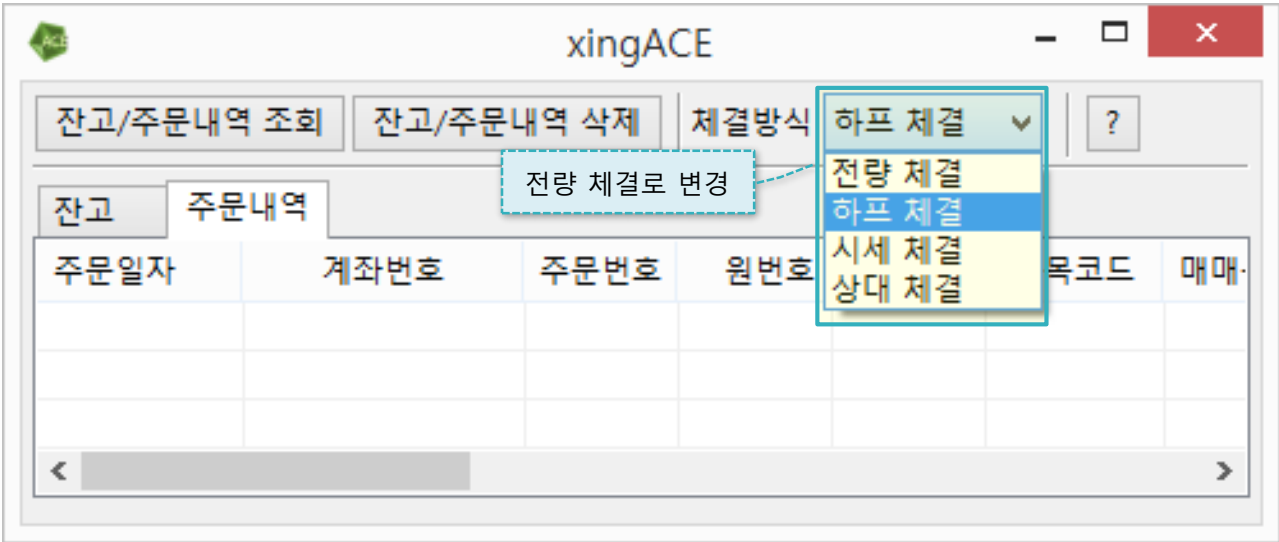
필드	내용
----	----

실시간 주문내역, 실시간 잔고내역 등 실시간 주문접수 및 주문체결을 받아 계산하는 화면을 테스트하는 데 사용하시면 편리한 기능입니다.

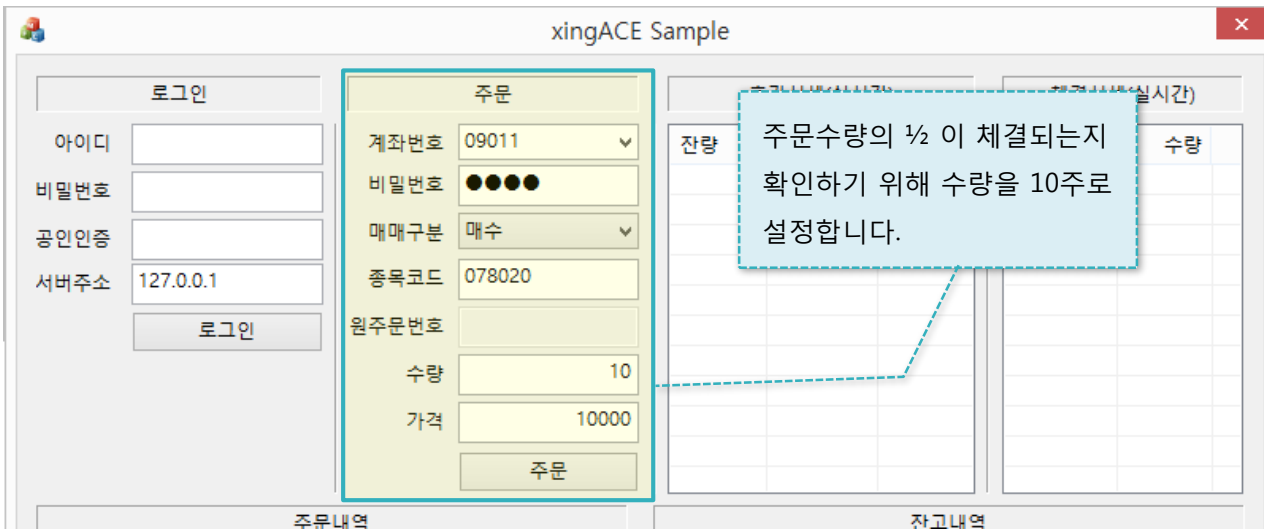
> 하프 체결

주문을 실행한 즉시 주문수량의 ½ 이 체결되는 방식입니다.

Step 1 xingACE 에서 체결방식을 “전량 체결”로 변경합니다.



Step 2 xingAPI 로 개발한 프로그램에서 주문을 실행합니다.



Step 3

주문수량이 1/2만 체결된것을 확인합니다.

로그인

아이디

비밀번호

공인인증

서버주소

로그인

주문

계좌번호

비밀번호

매매구분

종목코드

원주문번호

수량

가격

주문

호가시세(실시간)

잔량

호가

잔량

체결시세(실시간)

체결가

수량

주문내역

종목명	구분	주문수량	주문가격	체결수량	체결가
이트레...	매수신규	10	100.00	10	100.00

주문수량 10주 중 5주가 체결되었습니다.

잔고내역

종목명	잔고수량	평균단가	평가손익	손익율
이트레이...	10	10000.0	-2000	

잔고는 5 주 되었습니다.

실시간 로그

시간	TR	Data
18:01:34	SC1	0000000001090 B 063 HT ...
18:01:34	SC0	0000000001090 B SONAT000063 ...

1. 실시간 주문접수(SC0)
2. 실시간 주문체결(SC1)
순으로 받은 것을 확인 할 수 있습니다.

필드	내용
----	----

Step 4 직전 주문에 대한 정정주문을 실행하고 체결을 확인합니다.

로그인

아이디
kaienht

비밀번호
●●●●●●●●

공인인증
●●●●●●●●

서버주소
127.0.0.1

로그인

주문

계좌번호
09011

비밀번호

매매구분
정정

종목코드
078020

원주문번호
1

수량
5

가격
10100

주문

호가시세(실시간)

잔량 호가 잔량

체결시세(실시간)

체결가 수량

주문내역

주문번호	종목명	구분	주문수량	주문가격	체결수량
1	이트레...	매수신규	10	10000.00	5
2	이트레...	매수정정	5	10100.00	2

주문수량 5주 중 2주가 체결되었습니다.

잔고내역

종목명	잔고수량	평균단가	평가손익	손익율
이트레이...	7	10028.00	-2510	-3.000000

잔고는 7 주 되었습니다.

실시간 로그

시간	TR	Data	필드	내용
18:55:10	SC1	000000000109011	B 063 HT ...	
18:55:10	SC2	000000000109011	B SONAS100063	
18:55:10	SC0	000000000109011	B SONAT001063	

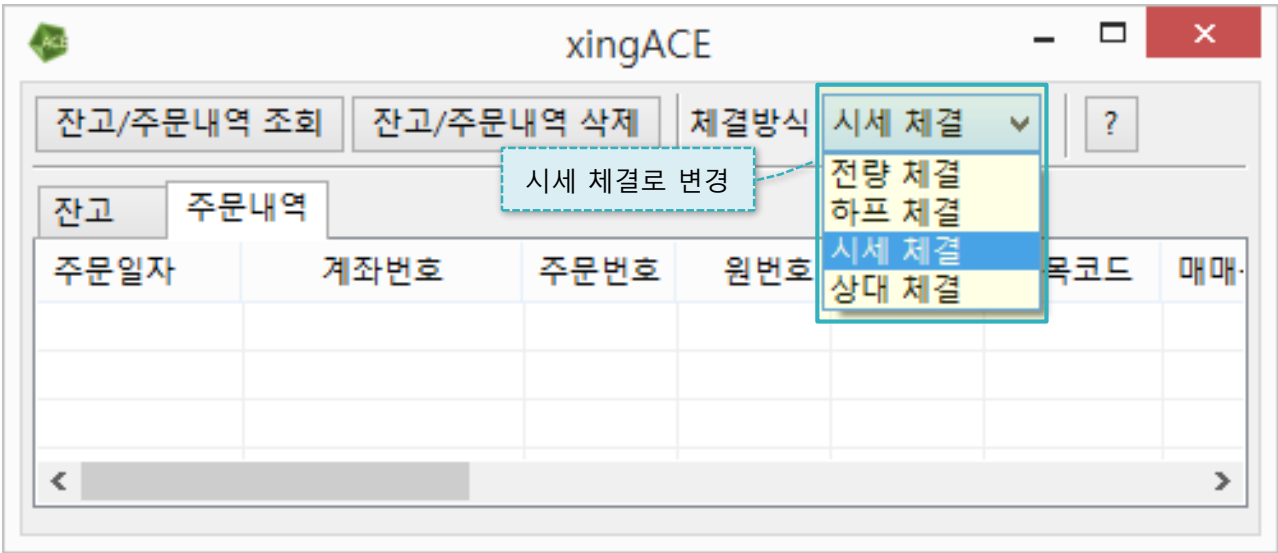
1. 실시간 주문접수(SC0)
2. 실시간 주문정정/취소(SC2)
3. 실시간 주문체결(SC1)
순으로 받은 것을 확인 할 수 있습니다.

1. 실시간 주문내역의 정정/취소 주문을 테스트하는데 사용하시면 편리한 기능입니다.
2. 주문수량이 홀수인 경우엔 반올림 하지 않고 버립니다.
예를 들면, 주문수량이 5일 경우엔 체결수량은 2 입니다.
3. 체결수량이 0일 경우에는 전량체결 됩니다.

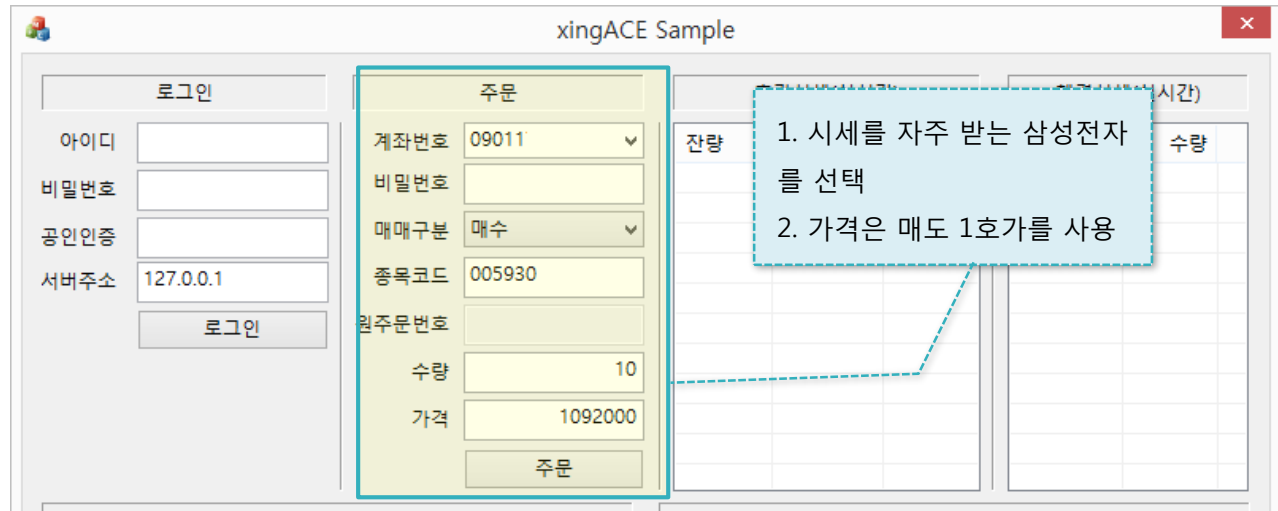
> 시세 체결

시세체결은 장중에 거래소에서 발생한 체결시세가 주문낸 가격과 같을 경우 체결시키는 방식을 말합니다. 당사 모의투자 시스템에서 사용하는 방식입니다.

Step 1 xingACE 에서 체결방식을 “전량 체결”로 변경합니다.



Step 2 xingAPI 로 개발한 프로그램에서 주문을 실행합니다.



- 1. 실시간 호가시세 및 체결시세는 주문이 발생한 직후부터 받도록 설정되어 있습니다.
- 2. 테스트를 위해 너무 많이 발생하는 호가시세는 제외하였습니다.

Step 3 시세가 들어오면서 체결이 되는 것을 확인합니다.

로그인

아이디
비밀번호
공인인증
서버주소 127.0.0.1
로그인

주문

계좌번호 09011701768
비밀번호
매매구분 매수
종목코드 005930
원주문번호
수량
가격
주문

호가시세(실시간)

잔량
호가
잔량

체결시세(실시간)

체결가
수량
1090000 1
1090000 3
1091000 1
1091000 1
1091000 1
1090000 1
1090000 7
1090000 8
1090000 3
1091000 5

주문내역

종목명	구분	주문수량	주문가격	체결수량	체결가
삼성전자	매수...	10	1092000...	10	1090500...

3번째 체결시세가 들어오면서 전
량 체결되었습니다.

조회

잔고내역

종목명

실시간 로그

시간	TR	Data
11:18:38	SC1	000000000109011 B 063
11:18:38	S3_	111841 5?0005000?00046?1090000?90000 01090000?90006
11:18:37	SC1	000000000109011 B 063
11:18:37	S3_	111840 5?0005000?00046?1090000?90000 01090000?90006 0109700...
11:18:36	SC1	000000000109011 B 063 HT ...
11:18:36	S3_	111840 5?0004000?00037?1091000?90000 01090000?90006 0109700...
11:18:34	SC0	000000000109011 B SONAT000063

체결시세 5주, 3주, 8주, 7주
가 순서대로 들어온것을 확
인 할 수 있습니다.

1. 실시간 주문접수(SC0)
2. 실시간 시세체결(S3_) => 5주
3. 실시간 주문체결(SC1) => 5주 체결
4. 실시간 시세체결(S3_) => 3주
5. 실시간 주문체결(SC1) => 3주 체결
6. 실시간 시세체결(S3_) => 8주
7. 실시간 주문체결(SC1) => 2주 체결

시세체결이 들어온 수량 만큼 체결이 되
는 것것을 확인할 수가 있습니다.

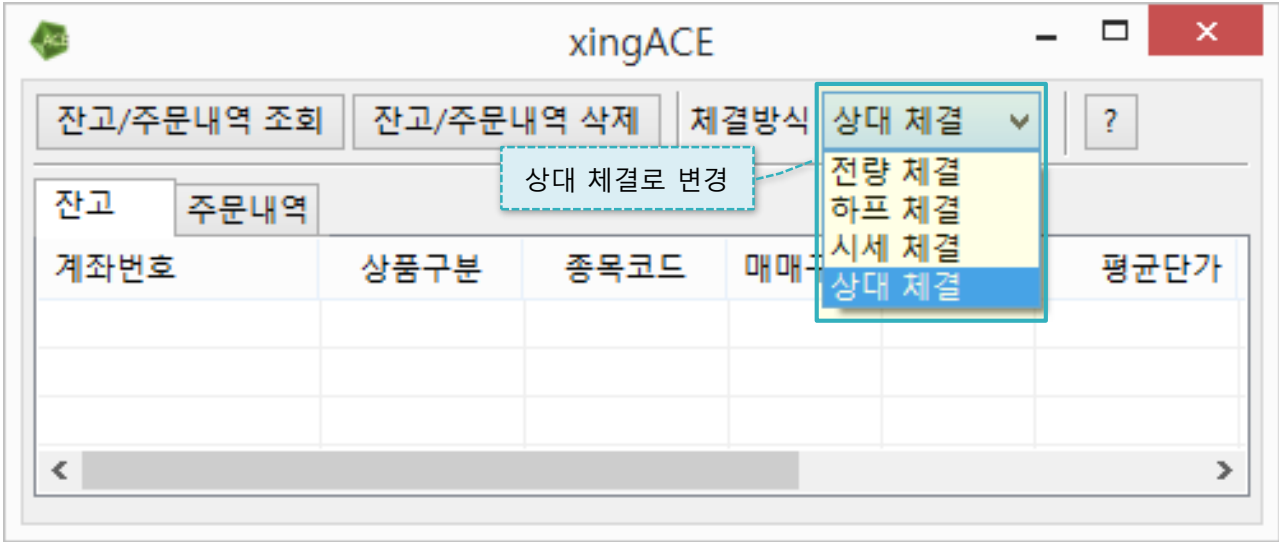
- 1) 1,091,000원에 5주 시세 발생 => 1,091,000원에 5주 주문 체결
- 2) 1,090,000원에 3주 시세 발생 => 1,090,000원에 3주 주문 체결
- 3) 1,090,000원에 8주 시세 발생 => 1,090,000원에 2주 주문 체결
- 총 10주 체결이 되었고 평균체결가 1,090,500원에 10주 체결된것을 볼 수 있습니다.

시세체결은 실제 장에서와 같이 체결되므로 프로그램을 시뮬레이션 하기에 좋습니다.

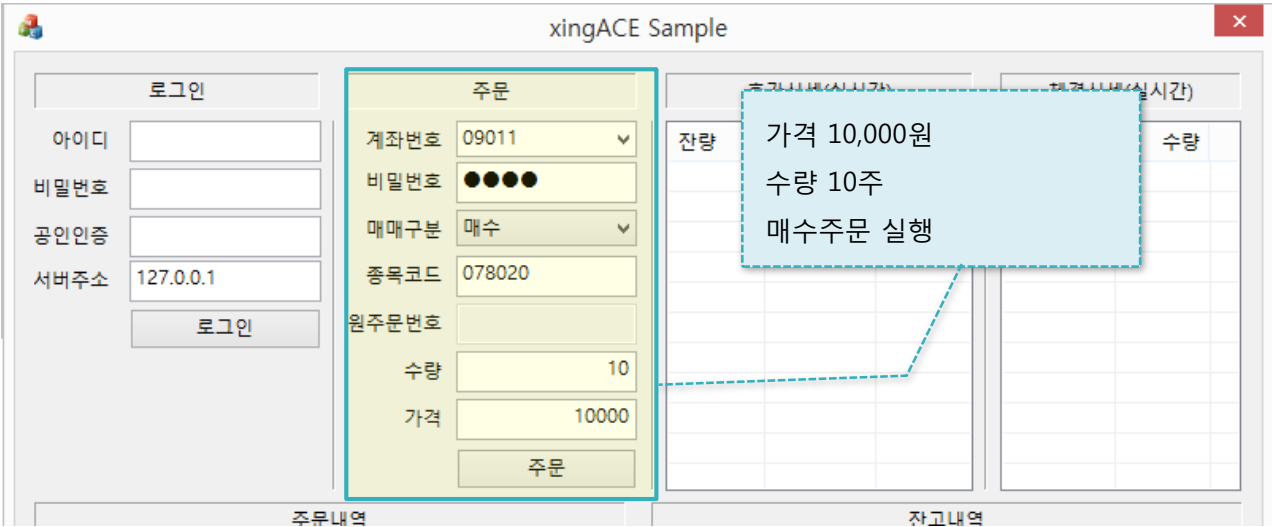
> 상대 체결

주문을 실행했을때 상대호가에 잔량이 있어야 체결이 되는 방식입니다.
상대호가의 잔량도 주문에 의해 만들어지게 됩니다. 시세로 받은 잔량을 의미하는게 아닌 직접 주문을
실행해야 합니다.
실제 거래소와 같이 주문이 발생하면 실시간 시세호가가 생성되고 체결이 발생하면 실시간 시세체결이
발생합니다.

Step 1 xingACE 에서 체결방식을 “상대 체결”로 변경합니다.



Step 2 상대호가의 잔량을 만들기 위해 매수주문을 실행합니다.



Step 3 주문이 접수만 된 상태와 호가시세가 들어온것을 확인합니다.

로그인

아이디
비밀번호
공인인증
서버주소

주문

계좌번호
비밀번호
매매구분
수량
가격

호가시세(실시간)

잔량
호가
잔량

체결시세(실시간)

체결가
수량

주문내역

종목명
구분
주문수량
주문가격
체결수량
체결가

잔고내역

종목명
잔고수량
평균단가
평가손익
손익율

실시간 로그

시간
TR
Data

필드
내용

매수주문에 의해 매수1호가
가 생성이 되었습니다.

매수주문에 의한 실시간 주문접수(SC0)
와 실시간 호가시세(HA_)가 발생되었습
니다.

아직 체결이 되지 않았습니다.

상대호가에 잔량이 존재하지 않기 때문에 아직 체결이 발생하지 않았습니다.
상대체결의 경우에만 주문이 발생할 경우 실제 거래소처럼 실시간 시세호가 발생합니다.

Step 4 매수1호가(10,000원), 10주에 매도주문을 실행하고 체결여부를 확인합니다.

로그인

아이디

비밀번호

공인인증

서버주소

주문

계좌번호

비밀번호

매매구분

수량

가격

주문

호가시세(실시간)

잔량	호가	잔량
0	0	
0	0	
0	0	
0	0	
0	0	
	0	0
	0	0
	0	0
	0	0
	0	0

체결시세(실시간)

체결가	수량
10000	10

주문내역

종목명	구분	주문수량	주문가격	체결수량	체결가
이트레...	매수...	10	10000.00	10	10000.00
이트레...	매도...	10	10000.00	10	10000.00

잔고내역

종목명	잔고수량	평균단가	평가손익	손익율
-----	------	------	------	-----

실시간 로그

시간	TR	Data
09:55:29	HA_	095529 0000000 0000000 0000000000 000000000 000000
09:55:29	K3_	095529 2 00000330 0000000 00010000 00010000
09:55:29	SC1	0000000000109011 B 063
09:55:29	SC1	0000000000109011 B 063
09:55:29	SC0	0000000000109011 B SONAT000063
09:54:57	HA_	095457 0000000 0010000 0000000000 0000000010 0000000 0000000 0...
09:54:57	SC0	0000000000109011 B SONAT000063

매도주문에 의해 매수호가 잔량이 사라졌습니다.

10,000원, 10주 체결시세 발생

Step3의 매수주문과 Step4의 매도주문이 체결되었습니다.

매수주문과 매도주문이 동시에 체결되었으므로 잔고는 없습니다.

매도주문에 의한 주문접수(SC0)가 발생한 후에 매도주문 및 매수주문의 주문체결(SC1)이 발생하였고 시세체결(K3_)와 시세호가(HA_)가 발생한 것을 확인할 수 있습니다.

xingACE에서는 매도주문에 대한 잔고 적합성을 체크하지 않기 때문에 잔고가 없어도 위와같이 매도주문이 거부되지 않습니다.

Stop/Limit/Trailing Stop 주문을 테스트 하기 위해 특정 시세가 필요합니다. 실제 장에서는 해당 시세를 기다리기가 까다롭습니다. 이럴때 상대체결을 이용하면 시세가 발생하므로 Stop 주문등 특정 시세가 필요한 테스트를 하기에 용이합니다.

※ 주의

상대체결일 경우에만 실시간 시세가 발생합니다.
시세조회를 하게 되면 조회는 실서버에서 가져오므로 xingACE내의 호가는 가져오지 못합니다.

> 실시간 Data 전송

xingACE에는 외부에서 실시간 Data를 받아 xingAPI 프로그램으로 전송을 해주는 기능이 추가되어 있습니다. 외부에서 xingACE로 실시간 Data를 보낼 수 있는 API를 제공합니다.
그러므로, 사용자는 실시간 데이터를 만들어 xingACE를 통하여 자신의 xingAPI 프로그램으로 실시간 데이터를 전송할 수 있습니다.

> 실시간 Data 전송 API

시세재전송 API는 몇 개의 함수를 제공하는 1개의 DLL로 구성되어 있습니다.

1) 사용 파일

xingACERDSender.dll 은 xingACE로 실시간 데이터를 전송할 수 있는 API DLL 입니다.
Visual C++ 2012 로 개발되어졌으며 관련 MFC 파일이 필요합니다.

2) 함수

함수명	설명
xingACERDSend	실시간 데이터를 xingACE로 전송합니다. TR Code, InBlock Data 와 크기, OutBlock Data 와 크기 의 Parameter를 필요로 합니다.
xingACERDOpen	전송을 위해 API를 Open합니다. Open을 하지 않고 Send를 할 경우 Send 내에서 자동적으로 Open을 실행하기 때문에 Open을 꼭 할 필요는 없습니다.
xingACERDClose	전송이 끝난 후에 Close 를 합니다. Close를 하지 않고 프로그램을 종료하면 프로그램이 비정상종료될 수 있습니다.

3) 실시간 Data 전송 Sample

시세재전송 API 사용을 쉽게 하기 위해 Visual C++로 개발된 Sample을 제공합니다.
Sample 은 xingACE가 설치된 폴더에 Sample_xingACE.exe 를 실행하시면 됩니다.

Step 1 실시간 TR 리스트에서 전송할 TR을 선택합니다.

(1) 실시간 TR 리스트

코드	설명	파일
PM_	KOSPI 프로그램매매전체집계 (PM)	D:\ProgramSource\xing...
S2_	KOSPI우선호가(S2)	D:\ProgramSource\xing...
S2_	KOSPI우선호가(S2)	D:\ProgramSource\xing...
S3_	KOSPI체결(S3)	D:\ProgramSource\xing...
S3_	KOSPI체결(S3)	D:\ProgramSource\xing...
S4_	KOSPI기세(S4)	D:\ProgramSource\xing...

(2) InBlock 리스트

이름	타입	길이	설명	데이터
shcode	char	6	단축코드	

(3) OutBlock 리스트

이름	타입	길이	설명	데이터
chetime	char	6	체결시간	
sign	char	1	전일대비구분	
change	long	8	전일대비	
drate	float	6.2	등락률	
price	long	8	현재가	
opentime	char	6	시가시간	
open	long	8	시가	
hightime	char	6	고가시간	
high	long	8	고가	
lowtime	char	6	저가시간	
low	long	8	저가	
cgubun	char	1	체결구분	
cvolume	long	8	체결량	

(5) 전송리스트

TR	InBlock	OutBlock
----	---------	----------

(7) 전송주기

사용법

(1) 실시간 TR 리스트에서 전송할 TR을 선택합니다.
(2) InBlock 리스트의 필드에 전송할 데이터를 입력합니다.
(3) OutBlock 리스트의 필드에 전송할 데이터를 입력합니다.
(4) 입력된 데이터를 "추가" 버튼을 눌러 전송리스트에 추가합니다.
(1)~(4)를 반복하며 전송할 데이터를 추가합니다.
(5) 전송리스트에서 전송할 순서를 설정합니다.
(6) 전송할 주기를 설정합니다.
전송주기는 1/1000 초로 설정이 가능합니다.
(7) 전송주기 설정

- 실시간 TR 리스트에서 전송할 실시간 TR을 선택합니다.
(실시간 TR 은 Sample이 설치된 폴더 밑에 있는 RES 폴더에서 .res 파일을 읽어와서 표시합니다.)
- 실시간 TR을 선택하면 InBlock 리스트와 OutBlock 리스트에 해당 TR의 필드가 표시됩니다.

Step 2 InBlock리스트의 필드값과 OutBlock 리스트의 필드값을 채워줍니다.

Sample_xingAceRdSender

(1) 실시간 TR 리스트

코드	설명
PM_	KOSPI프로그램매매전체집계(PM)
S2_	KOSPI우선호가(S2)
S2_	KOSPI우선호가(S2)
S3_	KOSPI체결(S3)
S3_	KOSPI체결(S3)
S4	KOSPI기세(S4)

(2) InBlock 리스트

이름	타입	길이	설명	데이터
shcode	char	6	단축코드	005930

005930

입력

(3) OutBlock 리스트

이름	타입	길이	설명	데이터
chetime	char	6	체결시간	
sign	char	1	전일대비구분	
change	long	8	전일대비	
drate	float	6.2	등락율	
price	long	8	현재가	
opentime	char	6	시가시간	
open	long	8	시가	
hightime	char	6	고가시간	
high	long	8	고가	
lowtime	char	6	저가시간	
low	long	8	저가	
cgubun	char	1	체결구분	
cvolume	long	8	체결량	
...

입력

(4) 추가

<

↑위로

↓아래로

(5) 전송리스트

(6) 전송간격 설정

/1000 초 간격

사용법

(1) 실시간 TR 리스트

(2) InBlock 리스트의

(3) OutBlock 리스트의

(4) 입력된 데이터를

(1)~(4)를 반복하며 전송

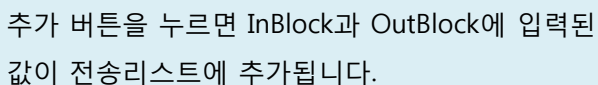
(5) 전송리스트에서 2

(6) 전송할 주기를 설

전송주기는 1/100

(7) ListView32

Step 3



Step 4 “전송” 버튼을 눌러 전송리스트에 추가된 데이터를 xingACE로 전송합니다.

Sample_xingAceRdSender

(5) 전송리스트

TR	InBlock	OutBlock
S3_	005930	090000 1 00010000 000050 01050000 00000000 0000..

시간을 지정하면 지정된 시간마다 1건씩 자동으로 전송합니다.

(4) 추가

↑ 위로 ↓ 아래로 전송제거 선택삭제 전체삭제 저장하기 읽어오기

(6) 전송간격 설정

100 /1000 초 간격으로 전송

(7) 전송

사용법

(1) 실시간 TR 리스트에서 전송할 TR을 선택합니다.

(2) InBlock 리스트의 필드에 전송할 데이터를 입력합니다.

(3) OutBlock 리스트의 필드에 전송할 데이터를 입력합니다.

(4) 입력된 데이터를 “추가” 버튼을 눌러 전송리스트에 추가합니다.

(1)~(4)를 반복하여 전송할 데이터를 추가합니다.

(5) 전송리스트에서 전송할 순서를 설정합니다.

(6) 전송할 주기를 설정합니다.

전송주기는 1/1000 초로 설정이 가능합니다.

(7) 전송

전송 버튼을 누르면 xingACE로 전송리스트의 데이터를 전송합니다.

저장하기/읽어오기 버튼을 이용하면 설정된 전송리스트 데이터를 파일로 저장하고 다음에 다시 읽어올 수 있습니다.

4) 실시간 Data 전송 Sample 소스 분석하기

실시간 Data 전송 Sample 은 xingAPI 가 설치된 폴더의 Sample 폴더에서 찾을 수 있습니다.
소스는 Dialog Base 로 구성되어 있으며 주된 기능은 Sample_xingACESenderDlg.cpp 에 있으며
Utility 함수는 stdafx.cpp 에서 찾을 수 있습니다.

주요 함수 리스트입니다.

```
//=====
// 컨트롤 초기화
//=====
void CSample_xingACERdSenderDlg::InitControl()

//=====
// TR List에 TR 목록을 표시한다.
//-----
// .res 폴더에는 TR 파일들이 있다.
// 이 파일의 목록을 읽어와서 표시한다.
//=====
void CSample_xingACERdSenderDlg::InitTrData()

//=====
// TR res 파일을 읽어서 InBlock, OutBlock 의 필드를 각 List에 추가한다.
//=====
void CSample_xingACERdSenderDlg::ProcessTR( int nSelect )

//=====
// InBlock 입력 버튼을 눌렀을때 InBlock Edit 의 값을 컨트롤에 넣어준다.
//=====
void CSample_xingACERdSenderDlg::OnClickedButtonUpdateInblock()

//=====
// OutBlock 입력 버튼을 눌렀을때 OutBlock Edit 의 값을 컨트롤에 넣어준다.
//=====
void CSample_xingACERdSenderDlg::OnClickedButtonUpdateOutblock()

//=====
// 전송 버튼 클릭시 Timer를 기동시킨다.
//=====
void CSample_xingACERdSenderDlg::OnClickedButtonSend()

//=====
// 화면이 종료될때 Uninit 를 한다.
//=====
void CSample_xingACERdSenderDlg::OnClose()

//=====
// 데이터를 전송한다.
//=====
void CSample_xingACERdSenderDlg::SendRealData()
```

함수들 중에 xingACERdSender API를 사용한 함수는 SendRealData() 와 OnClose() 함수이며 이 중
SendReadlData() 함수에 대해 분석을 하도록 하겠습니다.

SendRealData()

이 함수는 Time 이벤트가 발생하였을때 호출되는 함수이며, 전송리스트에서 데이터를 가져와 xingACE 로 데이터를 전송하는 역할을 합니다. 이때 중요한것은 OutBlock Data를 만들때 Attribute 처리를 해주 야 한다는 것입니다. (Attribute는 xingAPI DLL 가이드 참조)

```
void CSample_xingACERdSenderDlg::SendRealData()
{
    // 전송리스트의 갯수보다 Index가 더 크면 타이머를 중지하고 전송전 상태로 만든다
    int nCount = m_ctrlSendList.GetItemCount();
    if( m_nSendIndex >= nCount )
    {
        // Timer 를 종료한다.
        KillTimer( TID_SEND );

        // 버튼의 Caption 을 전송으로 설정한다.
        SetDlgItemText( IDC_BUTTON_SEND, BTN_TEXT_SEND );

        return;
    }

    // 전송할 데이터를 구하고
    CString strTr      = m_ctrlSendList.GetItemText( m_nSendIndex, 0 );
    CString strInBlock = m_ctrlSendList.GetItemText( m_nSendIndex, 1 );
    CString strOutBlock = m_ctrlSendList.GetItemText( m_nSendIndex, 2 );

    // 상태를 전송중으로 바꾸고
    m_ctrlSendList.SetItemText( m_nSendIndex, 3, "전송중" );

    // 데이터를 전송하고
    BOOL bResult = m_iSender.Send(
        strTr,                // 실시간 TR Code
        strInBlock,           // InBlock Data
        strInBlock.GetLength(), // InBlock 의 크기
        strOutBlock,         // OutBlock 의 Data
        strOutBlock.GetLength() // OutBlock 의 크기
    );

    // 결과를 상태에 입력한다.
    m_ctrlSendList.SetItemText( m_nSendIndex, 3, bResult ? "전송" : "실패" );

    // 전송이 완료되었으므로 Index를 하나 늘린다.
    m_nSendIndex++;

    // 다음에 보낼 것이 없다면 여기서 중단한다.
    if( m_nSendIndex >= nCount )
    {
        // Timer 를 종료한다.
        KillTimer( TID_SEND );

        // 버튼의 Caption 을 전송으로 설정한다.
        SetDlgItemText( IDC_BUTTON_SEND, BTN_TEXT_SEND );
    }
}
```

리스트의
첫번째 컬럼은 TR Code
두번째 컬럼은 InBlock Data,
세번째 컬럼은 OutBlock Data
이고 이것을 컬럼별로 가져온다.

데이터 전송

전송 데이터 위치를 하나 증가시켜
서 다음 Timer 이벤트를 대비한다.

감사합니다