

STage 주문함수 안내

1. 주문함수 안내

Buy

Sell

ExitLong

ExitShort

Stop 주문 및 기타

2. 주문 실행 방법

3. 유의사항

1. 주문함수 안내

STage에서 주문함수는 가장 핵심적인 함수라고 할 수 있습니다. 매매전략을 구현하는 단계에서 포함되어 실제 시스템 트레이딩을 수행할 수 있게 하는 함수입니다. STage의 주문함수는 매수/매도 주문실행의 기본적인 기능 과 시스템 트레이딩의 효율을 끌어올릴 수 있는 STage만의 This bar 기능을 갖고 있습니다. 사용자께서는 주문함수의 다양한 기능을 사용하여 고급전략을 작성하실 수 있습니다.

주문함수는 실제 매매에 직접적으로 영향을 미치는 함수이기 때문에 그 사용법을 정확히 숙지하셔야 합니다. 잘못된 사용 및 부주의한 사용은 예기치 않은 주문 수행 오류를 불러 일으킬 수 있으며, 이와 관련한 3. 유의사항 부분은 반드시 살펴보시기 바랍니다.

Buy

매수포지션을 취하는 주문함수입니다. 보유 중인 매도포지션이 있다면 청산 후 매수포지션을 취합니다. 아래의 매개변수를 지정하여 구체적인 주문 실행을 할 수 있습니다.

문법:

Buy [("주문 이름")] [해당종목] [주문 계약 수] [주문 실행 방법]

추가적인 매개변수 없이 Buy 만으로 주문함수 실행이 가능합니다. 다음의 문장은 완전한 문장입니다.

Buy

이 경우 아래의 문장을 실행하는 것과 같습니다.

Buy ("Buy") This Bar on Close

위의 주문은 주문계약수가 지정되어 있지 않습니다. 이 경우 SStage차트의 **[지표속성]-[매매신호]-[매매계약수]**의 지정된 값을 기본값으로 하여 주문계약이 실행됩니다.

[주문이름]

각각의 주문번호마다 고유의 이름을 붙일 수 있습니다. 다수의 매수주문을 실행할 경우 각각에 이름을 명명하면 포지션 식별 및 관리가 수월합니다. 또한 ExitLong주문으로 포지션을 청산할 때 주문 이름에 해당하는 포지션만 청산이 가능합니다.

Buy ("매수신호1")

위 문장은 현재 매수주문의 이름을 "매수신호1"로 명명하여 주문을 실행합니다. SStage차트 상에서 위의 매수주문이 발생한 곳은 "매수신호1"이라고 표시됩니다.

[해당종목]

SStage차트에서 복수종목을 참조할 경우 차례대로 Data1, Data2, Data3... 가 됩니다. **[해당종목]** 부분에 Data1, Data2를 표기함으로 해당 종목에 주문을 실행합니다. 생략 시에는 Data1으로 다루어집니다.

Buy ("매수신호1") Data1

Buy ("매수신호2") Data2

Buy ("매수신호3") Data3

[주문 계약 수]

주문 계약 수를 지정합니다. 아래와 같이 사용이 가능합니다.

Buy ("매수번호 1") **1 Contract**

Buy 10 Contracts

Buy Value1 Shares

Buy 1 Share

"수량 + **Contracts**"와 같이 지정하며 여기서 Contract, Contracts, Share, Shares는 동의어로 모두 동일한 효과를 갖습니다. 주문 계약 수를 지정하지 않을 경우 SStage차트의 **[지표속성]-[매매번호]-[매매계약수]**의 지정된 값을 기본값으로 하여 주문계약이 실행됩니다.

[주문 실행 방법]

Buy 주문에서 사용할 수 있는 주문 실행 방법은 총 7가지가 있습니다.

...this bar on close	백테스트 목적으로 사용되는 방법으로, 현재봉의 종가에 주문을 실행합니다.
...next bar at market	다음봉의 시가에 시장가 주문을 실행합니다.
...next bar at price Stop	다음봉의 가격이 기준가(price)보다 같거나 클 경우 주문을 실행합니다.
...next bar at price Limit	다음봉의 가격이 기준가(price)보다 같거나 작을 경우 주문을 실행합니다.
...this bar at market	현재봉에서 즉시 시장가 주문을 실행합니다.
...this bar at price Stop	현재봉에서 현재가(종가)가 기준가(price)보다 같거나 크게 되는 즉시 주문을 실행합니다.
...this bar at price Limit	현재봉에서 현재가(종가)가 기준가(price)보다 같거나 작게 되는 즉시 주문을 실행합니다.

아래와 같이 사용하실 수 있으며 자세한 설명은 본문서 **2. 주문 실행 방법**을 참고하시길 바랍니다.

Buy 100 Contracts This Bar on Close

Buy ("매수번호 1") **Next Bar at Market**

Buy 10 Shares Next Bar at High Stop

Buy ("매수번호 2") **5 Contracts This Bar at Market**

Sell

매도포지션을 취하는 주문함수입니다. 보유 중인 매수포지션이 있다면 청산 후 매도포지션을 취합니다. 아래의 매개변수를 지정하여 구체적인 주문 실행을 할 수 있습니다

문법:

Sell [("주문 이름")] [해당종목] [주문 계약 수] [주문 실행 방법]

추가적인 매개변수 없이 Sell 만으로 주문함수 실행이 가능합니다. 다음의 문장은 완전한 문장입니다.

Sell

이 경우 아래의 문장을 실행하는 것과 같습니다.

Sell ("Sell") This Bar on Close

위의 주문은 주문계약수가 지정되어 있지 않습니다. 이 경우 STage차트의 **[지표속성]-[매매신호]-[매매계약수]**의 지정된 값을 기본값으로 하여 주문계약이 실행됩니다.

[주문이름]

각각의 주문신호 마다 고유의 이름을 붙일 수 있습니다. 다수의 매도주문을 실행할 경우 각각에 이름을 명명하면 포지션 식별 및 관리가 수월합니다. 또한 ExitShort주문으로 포지션을 청산할 때 주문 이름에 해당하는 포지션만 청산이 가능합니다.

Sell ("매도신호1")

위 문장은 현재 매도주문의 이름을 "매도신호1"로 명명하여 주문을 실행합니다. STage차트 상에서 위의 매도주문이 발생한 곳은 "매도신호1"이라고 표시됩니다.

[해당종목]

STage차트에서 복수종목을 참조할 경우 차례대로 Data1, Data2, Data3... 가 됩니다. [해당종목] 부분에 Data1, Data2를 표기함으로 해당 종목에 주문을 실행합니다. 생략 시에는 Data1으로 다루어집니다.

Sell ("매도신호1") Data1

Sell ("매도신호2") Data2

Sell ("매도신호3") Data3

[주문 계약 수]

주문 계약 수를 지정합니다. 아래와 같이 사용이 가능합니다.

- Sell ("매도신호 1") 1 Contract**
- Sell 10 Contracts**
- Sell Value1 Shares**
- Sell 1 Share**

"수량 + **Contracts**"와 같이 지정하며 여기서 Contract, Contracts, Share, Shares는 동의어로 모두 동일한 효과를 갖습니다. 주문 계약 수를 지정하지 않을 경우 STage차트의 **[지표속성]-[매매신호]-[매매계약수]**의 지정된 값을 기본값으로 하여 주문계약이 실행됩니다.

[주문 실행 방법]

Sell 주문에서 사용할 수 있는 주문 실행 방법은 총 7 가지가 있습니다.

...this bar on close	백테스트 목적으로 사용되는 방법으로, 현재봉의 종가에 주문을 실행합니다.
...next bar at market	다음봉의 시가에 시장가 주문을 실행합니다.
...next bar at price Stop	다음봉의 가격이 기준가(price)보다 같거나 클 경우 주문을 실행합니다.
...next bar at price Limit	다음봉의 가격이 기준가(price)보다 같거나 작을 경우 주문을 실행합니다.
...this bar at market	현재봉에서 즉시 시장가 주문을 실행합니다.
...this bar at price Stop	현재봉에서 현재가(종가)가 기준가(price)보다 같거나 크게 되는 즉시 주문을 실행합니다.
...this bar at price Limit	현재봉에서 현재가(종가)가 기준가(price)보다 같거나 작게 되는 즉시 주문을 실행합니다.

아래와 같이 사용하실 수 있으며 자세한 설명은 본문서 **2. 주문 실행 방법**을 참고하시길 바랍니다.

- Sell 100 Contracts This Bar on Close**
- Sell ("매수신호 1") Next Bar at Market**
- Sell 10 Shares Next Bar at High Stop**
- Sell ("매수신호 2") 5 Contracts This Bar at Market**

ExitLong

매수포지션을 청산하는 주문함수입니다. 매개변수를 지정하여 구체적인 주문 실행을 할 수 있습니다.

문법:

ExitLong [("주문 이름")] [해당종목] [From Entry ("매수주문이름")] [주문 계약 수] [주문 실행 방법]

추가적인 매개변수 없이 ExitLong 만으로 주문함수 실행이 가능합니다. 다음의 문장은 완전한 문장입니다.

ExitLong

이 경우 아래의 문장을 실행하는 것과 같습니다.

ExitLong ("XL") This Bar on Close

위의 주문은 주문계약수가 지정되어 있지 않습니다. 이 경우 STage차트의 [지표속성]-[매매신호]-[매매계약수]의 지정된 값을 기본값으로 하여 주문계약이 실행됩니다.

[주문이름]

청산주문에도 이름을 붙일 수 있습니다. 주문 각각에 이름을 명명하면 포지션 식별 및 관리가 수월합니다.

ExitLong ("매수청산1")

위 문장은 현재 청산주문의 이름을 "매수청산1"로 명명하여 주문을 실행합니다. STage차트 상에서 위의 청산주문이 발생한 곳은 "매수청산1"이라고 표시됩니다.

[해당종목]

STage차트에서 복수종목을 참조할 경우 차례대로 Data1, Data2, Data3... 가 됩니다. [해당종목] 부분에 Data1, Data2를 표기함으로 해당 종목에 주문을 실행합니다. 생략 시에는 Data1으로 다루어집니다. 그렇기 때문에 STage차트에서 단일종목만 다룰 경우 생략이 가능합니다.

ExitLong ("매수청산1") Data1

ExitLong ("매수청산2") Data2

ExitLong ("매수청산3") Data3

[From Entry (“매수주문이름”)]

ExitLong 을 사용할 때 특정 매수주문의 청산만 수행할 수 있습니다. 아래와 같이 매수주문의 이름을 지정하여 ExitLong 을 실행하면 됩니다.

Buy (“매수번호 1”) 10 Contracts Next Bar at Market

Buy (“매수번호 2”) 30 Contracts Next Bar at High Stop

ExitLong (“청산 1”) From Entry (“매수번호 1”) Next Bar at Market

위에서 “매수번호 1” 매수주문은 10 계약, “매수번호 2” 매수주문은 30 계약, 총 40 계약의 매수포지션이 취해 집니다. 그리고 ExitLong 에서 [From Entry (“매수번호 1”)] 을 지정했기 때문에, 청산은 “매수번호 1” 10 계약만 이루어지며 30 개의 포지션은 그대로 유지가 됩니다.

[주문 계약 수]

주문 계약 수를 지정합니다. 아래와 같이 사용이 가능합니다.

ExitLong (“매도번호 1”) 1 Contract

ExitLong 10 Contracts

ExitLong Value1 Shares

ExitLong 1 Share

“수량 + Contracts”와 같이 지정하며 여기서 Contract, Contracts, Share, Shares는 동의어로 모두 동일한 효과를 갖습니다. 주문 계약 수를 지정하지 않을 경우 STage차트의 [지표속성]-[매매번호]-[매매계약수]의 지정된 값을 기본값으로 하여 주문계약이 실행됩니다.

[주문 실행 방법]

ExitLong 주문에서 사용할 수 있는 주문 실행 방법은 총 7 가지가 있습니다.

...this bar on close	백테스트 목적으로 사용되는 방법으로, 현재봉의 종가에 주문을 실행합니다.
...next bar at market	다음봉의 시가에 시장가 주문을 실행합니다.
...next bar at price Stop	다음봉의 가격이 기준가(price)보다 같거나 클 경우 주문을 실행합니다.
...next bar at price Limit	다음봉의 가격이 기준가(price)보다 같거나 작을 경우 주문을 실행합니다.
...this bar at market	현재봉에서 즉시 시장가 주문을 실행합니다.
...this bar at price Stop	현재봉에서 현재가(종가)가 기준가(price)보다 같거나 크게 되는 즉시 주문을 실행합니다.
...this bar at price Limit	현재봉에서 현재가(종가)가 기준가(price)보다 같거나 작게 되는 즉시 주문을 실행합니다.

아래와 같이 사용하실 수 있으며 자세한 설명은 본문서 **2. 주문 실행 방법**을 참고하시길 바랍니다.

ExitLong 100 Contracts This Bar on Close

ExitLong ("매수신호 1") Next Bar at Market

ExitLong 10 Shares Next Bar at High Stop

ExitLong ("매수신호 2") 5 Contracts This Bar at Market

ExitShort

매도포지션을 청산하는 주문함수입니다. 매개변수를 지정하여 구체적인 주문 실행을 할 수 있습니다.

문법:

ExitShort [("주문 이름")] [해당종목] [From Entry ("매수주문이름")] [주문 계약 수] [주문 실행 방법]

추가적인 매개변수 없이 ExitShort 만으로 주문함수 실행이 가능합니다. 다음의 문장은 완전한 문장입니다.

ExitShort

이 경우 아래의 문장을 실행하는 것과 같습니다.

ExitShort ("XS") This Bar on Close

위의 주문은 주문계약수가 지정되어 있지 않습니다. 이 경우 STage차트의 [지표속성]-[매매신호]-[매매계약수]의 지정된 값을 기본값으로 하여 주문계약이 실행됩니다.

[주문이름]

청산주문에도 이름을 붙일 수 있습니다. 주문 각각에 이름을 명명하면 포지션 식별 및 관리가 수월합니다.

ExitShort ("매도청산1")

위 문장은 현재 청산주문의 이름을 "매도청산1"로 명명하여 주문을 실행합니다. STage차트 상에서 위의 청산주문이 발생한 곳은 "매도청산1"이라고 표시됩니다.

[해당종목]

STage차트에서 복수종목을 참조할 경우 차례대로 Data1, Data2, Data3... 가 됩니다. [해당종목] 부분에 Data1, Data2를 표기함으로 해당 종목에 주문을 실행합니다. 생략 시에는 Data1으로 다루어집니다. 그렇기 때문에 STage차트에서 단일종목만 다룰 경우 생략이 가능합니다.

ExitShort ("매도청산1") Data1

ExitShort ("매도청산2") Data2

ExitShort ("매도청산3") Data3

[From Entry (“매수주문이름”)]

ExitShort 을 사용할 때 특정 매도주문의 청산만 수행할 수 있습니다. 아래와 같이 매도주문의 이름을 지정하여 ExitShort 을 실행하면 됩니다.

Sell (“매도신호 1”) 10 Contracts Next Bar at Market

Sell (“매도신호 2”) 30 Contracts Next Bar at High Stop

ExitShort (“청산 1”) From Entry (“매도신호 1”) Next Bar at Market

위에서 “매도신호 1” 매도주문은 10 계약, “매도신호 2” 매도주문은 30 계약, 총 40 계약의 매도포지션이 취해 집니다. 그리고 ExitShort 에서 [From Entry (“매도신호 1”)] 을 지정했기 때문에, 청산은 “매도신호 1” 10 계약만 이루어지며 30 개의 포지션은 그대로 유지가 됩니다.

[주문 계약 수]

주문 계약 수를 지정합니다. 아래와 같이 사용이 가능합니다.

ExitShort (“매도신호 1”) 1 Contract

ExitShort 10 Contracts

ExitShort Value1 Shares

ExitShort 1 Share

“수량 + Contracts”와 같이 지정하며 여기서 Contract, Contracts, Share, Shares는 동의어로 모두 동일한 효과를 갖습니다. 주문 계약 수를 지정하지 않을 경우 STage차트의 [지표속성]-[매매신호]-[매매계약수]의 지정된 값을 기본값으로 하여 주문계약이 실행됩니다.

[주문 실행 방법]

ExitShort 주문에서 사용할 수 있는 주문 실행 방법은 총 7 가지가 있습니다.

...this bar on close	백테스트 목적으로 사용되는 방법으로, 현재봉의 종가에 주문을 실행합니다.
...next bar at market	다음봉의 시가에 시장가 주문을 실행합니다.
...next bar at price Stop	다음봉의 가격이 기준가(price)보다 같거나 클 경우 주문을 실행합니다.
...next bar at price Limit	다음봉의 가격이 기준가(price)보다 같거나 작을 경우 주문을 실행합니다.
...this bar at market	현재봉에서 즉시 시장가 주문을 실행합니다.
...this bar at price Stop	현재봉에서 현재가(종가)가 기준가(price)보다 같거나 크게 되는 즉시 주문을 실행합니다.
...this bar at price Limit	현재봉에서 현재가(종가)가 기준가(price)보다 같거나 작게 되는 즉시 주문을 실행합니다.

아래와 같이 사용하실 수 있으며 자세한 설명은 본문서 **2. 주문 실행 방법**을 참고하시길 바랍니다.

ExitShort 100 Contracts This Bar on Close

ExitShort ("매도신호 1") Next Bar at Market

ExitShort 10 Shares Next Bar at High Stop

ExitShort ("매도신호 2") 5 Contracts This Bar at Market

SetBreakEven

일정한 이익에 도달 후, 손익분기점에 청산주문 실행을 지정하는 함수입니다.

문법:

SetBreakEven([이익수준설정], [실행구분], [해당종목])

[이익수준설정]

진입평균가로부터 이익수준을 설정합니다. [실행구분]에 따라 설정이 가능합니다.

[실행구분]

이익수준을 설정할 때, 어떠한 방법을 사용할 지 정할 수 있습니다.

Percent, Price, Money, Point 4 가지의 사용이 가능합니다.

Percent	이익수준설정을 진입평균가 대비 백분율단위로 지정합니다.
Price	이익수준설정을 진입평균가 대비 가격단위로 지정합니다.
Money	이익수준설정을 진입평균가 대비 금액단위로 지정합니다.
Point	이익수준설정을 진입평균가 대비 포인트단위로 지정합니다.

[해당종목]

SStage 차트에서 복수종목을 참조할 경우 차례대로 Data1, Data2, Data3... 가 됩니다. [해당종목] 부분에 Data1, Data2 의 Data ID 를 표기함으로 해당 종목에 주문을 실행합니다. Data1 의 Data ID 는 1, Data2 의 Data ID 는 2 와 같습니다. 생략 시에는 Data1 으로 다루어집니다.

사용예

SetBreakEven(0.1, Percent, 1)

// Data1 종목에 대해 포지션 진입 후, 현재가가 진입평균가 대비 0.1% 가격에 도달하면 진입평균가에 청산주문을 지정합니다.

SetBreakEven(3, Price, 1)

// Data1 종목에 대해 포지션 진입 후, 현재가가 진입평균가 대비 3 만원 높은 가격에 도달하면 진입평균가에 청산주문을 지정합니다.

SetBreakEven(150000, money, 1)

// Data1 종목에 대해 포지션 진입 후, 현재가가 진입평균가 대비 금액으로 150000 원만큼 높은 가격에 도달하면 진입평균가에 청산주문을 지정합니다.

SetExitOnClose

장종료시 청산주문을 실행하는 함수입니다. 백테스트 목적으로 고안된 함수이며 실제 매매에서는 적용되지 않습니다.

문법:

SetExitOnClose

당일 종가는 증가봉이 완성되는 시점에 확인이 가능합니다. 하지만 증가봉이 완성되는 시점은 장종료인 상태이기 때문에 당일 증가를 기준으로 주문 실행이 불가능 합니다.

그렇기 때문에 실제 상황에서 당일 증가청산을 구현하기 위해서는 IF 문과 Time 을 활용하여 직접 코드로 구현하시거나 STage 차트의 지표속성 설정 -> [매매신호]탭 -> [당일청산] 설정에서 "당일 청산" 옵션 체크 한 후, 시간 설정을 하는 방법이 있습니다.

첫번째 방법으로는 사용자께서 직접 코드로 구현하시는 것입니다. IF 문과 Time 을 활용하면 특정 시각을 지정하여 포지션 청산을 수행할 수 있습니다.

If Time = 145000 Then ExitLong

두번째 방법으로는 STage 차트 상의 기능을 이용하는 것입니다. STage 차트의 지표속성 설정에서 당일 청산 기능을 지원합니다.

1) STage 차트에서 지표명을 더블클릭하거나 마우스 우클릭 후 "지표 속성" 선택

2) [매매신호] 탭 선택 후 나오는 화면을 보시면 비용, 매매계약수, 당일청산, 손절매 설정 항목이 있습니다.

3) 그 중 [당일청산] 설정을 선택하시면 세가지 항목이 있는데 "당일 청산" 옵션을 체크 한 후, 시간을 설정하시면 지정 시간에 포지션이 있을 경우 청산을 하게 되고 이 후 주문은 실행되지 않게 됩니다.

[매매신호]탭 -> [당일청산] -> [당일청산] 시간설정을 통해 원하시는 기능을 구현할 수 있습니다.

참고로 다른 항목의 기능은 아래와 같으며, [당일 증가 청산] / [당일 청산] 의 사용에 혼동이 없도록 주의하시길 바랍니다. (두 설정의 차이는 청산 후 주문 실행의 여부입니다) "당일 진입 금지"는 지정 시간 이후 주문 실행이 되지 않게 하는 설정이며 "당일 증가 청산"의 경우 지정 시간에 포지션이 있을 경우 청산하지만 이 후 주문 명령이 있을 경우 실행됩니다.

SetProfitTarget

설정된 목표이익에 도달하면 청산하는 주문입니다.

문법:

SetProfitTarget([이익수준설정], [실행구분], [해당종목])

[이익수준설정]

진입평균가로부터 이익수준을 설정합니다. [실행구분]에 따라 설정이 가능합니다.

[실행구분]

이익수준을 설정할 때, 어떠한 방법을 사용할 지 정할 수 있습니다.

Percent, Price, Money, Point 4 가지의 사용이 가능합니다.

Percent	이익수준설정을 진입평균가 대비 백분율단위로 지정합니다.
Price	이익수준설정을 진입평균가 대비 가격단위로 지정합니다.
Money	이익수준설정을 진입평균가 대비 금액단위로 지정합니다.
Point	이익수준설정을 진입평균가 대비 포인트단위로 지정합니다.

[해당종목]

SStage 차트에서 복수종목을 참조할 경우 차례대로 Data1, Data2, Data3... 가 됩니다. [해당종목] 부분에 Data1, Data2 의 Data ID 를 표기함으로 해당 종목에 주문을 실행합니다. Data1 의 Data ID 는 1, Data2 의 Data ID 는 2 와 같습니다. 생략 시에는 Data1 으로 다루어집니다.

사용예

SetProfitTarget(1.0, Percent, 1)

// Data1 종목에 대해 포지션 진입 후, 현재가가 진입평균가 대비 1.0% 가격에 도달하면 청산주문을 실행합니다.

SetProfitTarget (5, Price, 1)

// Data1 종목에 대해 포지션 진입 후, 현재가가 진입평균가 대비 5 만원의 가격에 도달하면 청산주문을 실행합니다.

SetProfitTarget (300000, money, 1)

// Data1 종목에 대해 포지션 진입 후, 현재가가 진입평균가 대비 금액으로 300000 원만큼의 가격에 도달하면 청산주문을 실행합니다.

SetStopLoss

설정된 손실에 도달하면 청산하는 주문입니다.

문법:

SetStopLoss([손실수준설정], [실행구분], [해당종목])

[손실수준설정]

진입평균가로부터 손실수준을 설정합니다. [실행구분]에 따라 설정이 가능합니다.

[실행구분]

손실수준을 설정할 때, 어떠한 방법을 사용할 지 정할 수 있습니다.

Percent, Price, Money, Point 4 가지의 사용이 가능합니다.

Percent	손실수준설정을 진입평균가 대비 백분율단위로 지정합니다.
Price	손실수준설정을 진입평균가 대비 가격단위로 지정합니다.
Money	손실수준설정을 진입평균가 대비 금액단위로 지정합니다.
Point	손실수준설정을 진입평균가 대비 포인트(틱)단위로 지정합니다.

[해당종목]

STage 차트에서 복수종목을 참조할 경우 차례대로 Data1, Data2, Data3... 가 됩니다. [해당종목] 부분에 Data1, Data2 의 Data ID 를 표기함으로 해당 종목에 주문을 실행합니다. Data1 의 Data ID 는 1, Data2 의 Data ID 는 2 와 같습니다. 생략 시에는 Data1 으로 다루어집니다.

사용예

SetStopLoss(1.0, Percent, 1)

// Data1 종목에 대해 포지션 진입 후, 현재가가 포지션 반대방향으로 진입평균가 대비 1.0% 가격에 도달하면 청산주문을 실행합니다.

SetStopLoss (5, Price, 1)

// Data1 종목에 대해 포지션 진입 후, 현재가가 포지션 반대방향으로 진입평균가 대비 5 만원의 가격에 도달하면 청산주문을 실행합니다.

SetStopLoss (300000, money, 1)

// Data1 종목에 대해 포지션 진입 후, 현재가가 포지션 반대방향으로 진입평균가 대비 금액으로 300000 원만큼의 가격에 도달하면 청산주문을 실행합니다.

SetTrailingStop

포지션의 이익이 특정 수준에 도달한 후 이익이 설정한 만큼 감소하면 청산하는 주문입니다.

문법:

SetTrailingStop([이익수준설정], [실행구분], [감소수준설정], [실행구분], [해당종목])

[이익수준설정]

TrailingStop을 동작시킬 특정 이익 수준을 설정합니다. [실행구분]에 따라 설정이 가능합니다.

[감소수준설정]

이익수준대비 감소수준을 설정합니다. [실행구분]에 따라 설정이 가능합니다.

[실행구분]

이익수준, 감소수준을 설정할 때, 어떠한 방법을 사용할 지 정할 수 있습니다.
Percent, Price, Money, Point 4 가지의 사용이 가능합니다.

Percent	가격설정을 기준가 대비 백분율단위로 지정합니다.
Price	가격설정을 기준가 대비 가격단위로 지정합니다.
Money	가격설정을 기준가 대비 금액단위로 지정합니다.
Point	가격설정을 기준가 대비 포인트(틱)단위로 지정합니다.

[해당종목]

STage 차트에서 복수종목을 참조할 경우 차례대로 Data1, Data2, Data3... 가 됩니다. [해당종목] 부분에 Data1, Data2 의 Data ID 를 표기함으로 해당 종목에 주문을 실행합니다. Data1 의 Data ID 는 1, Data2 의 Data ID 는 2 와 같습니다. 생략 시에는 Data1 으로 다루어집니다.

사용예

SetTrailingStop(1.0, Percent, 0.2, Percent, 1)

// Data1 종목에 대해 포지션 진입 후, 1% 이익에 도달하면 0.2% 만큼 이익이 감소할 경우 청산주문을 실행합니다.

SetTrailingStop (5, Price, 1, Price, 1)

// Data1 종목에 대해 포지션 진입 후, 현재가가 포지션 이익방향으로 5 만금 상승하면 이후 1 만금 가격이 하락할 경우 청산주문을 실행합니다.

SetTrailingStop (300000, money, 100000, money, 1)

// Data1 종목에 대해 포지션 진입 후, 현재가가 포지션 이익방향으로 금액으로 300000 원만큼의 가격에 도달하면 이후 금액으로 100000 원 만큼 가격이 하락할 경우 청산주문을 실행합니다.

2. 주문 실행 방법

STage의 주문 실행 방법은 아래와 같습니다. (총 7가지 실행 방법)

	Next bar	This bar
On close		This bar on close
Market	Next bar at Market	This bar at Market
Stop	Next bar at price Stop	This bar at price Stop
Limit	Next bar at price Limit	This bar at price Limit

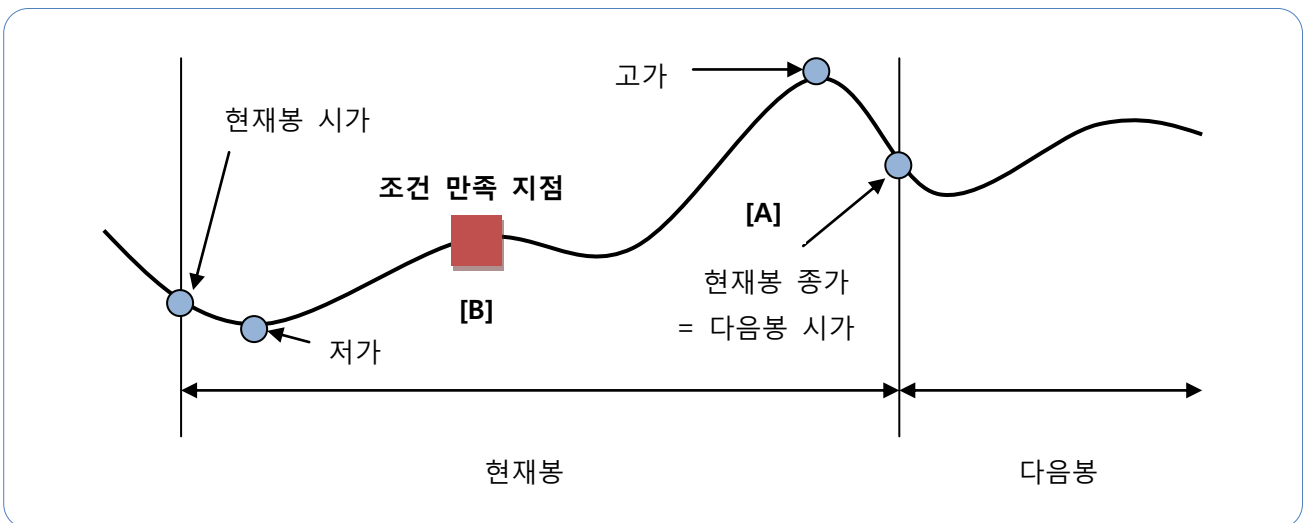
< Next bar 와 This bar의 차이 >

Next bar와 This bar의 차이는 주문이 발생하는 시점입니다. Next bar 주문은 현재봉이 완성된 후 다음 봉 시가가 형성 되었을 때 주문을 발생시키는 실행 방법입니다. 반면 This bar 주문은 현재봉의 완성을 기다리지 않고 현재 봉에서 현재가 기준으로 주문을 실행합니다. 다음과 같이 사용이 가능합니다.

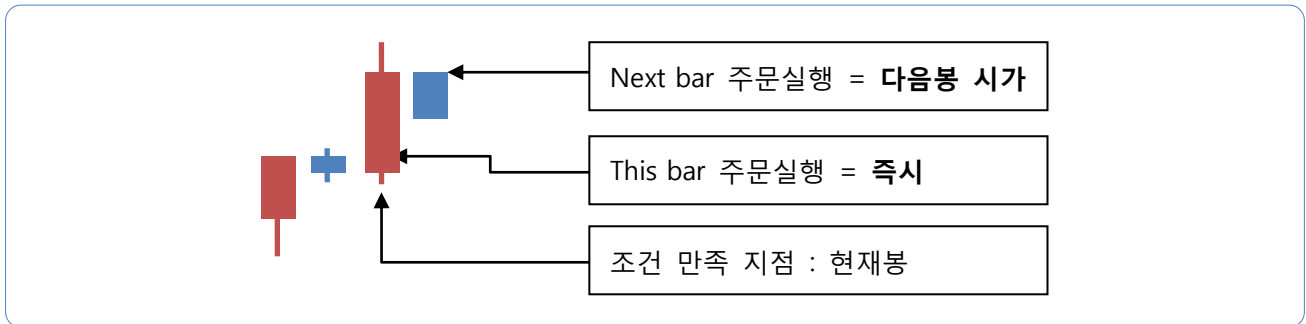
- Buy 10 Shares Next Bar at market
- Buy ("매수신호1") Data1 Next Bar at arket
- Buy ("매수") Data1 5 Contracts Next Bar at market
- Sell 10 Contracts Next Bar at market

- [1] IF Condition = True Then Buy 1 contract **next bar at market**
- [2] IF Condition = True Then Buy 1 contract **this bar at market**

그림과 같이 가격이 진행되는 상황에서 빨간 사각형 부분에서 조건이 만족되어 주문이 실행 될 경우 [1] next bar의 경우 주문 실행이 다음봉 시가 [A] 지점에서 이루어지게 됩니다. 반면 [2] this bar 주문의 경우 조건 만족 지점 [B]에서 즉시 주문이 실행됩니다.



Next bar 와 This bar 의 차이를 봉차트에서 살펴보면 아래와 같습니다.



This bar 주문은 STage 만의 특별한 기능으로 시세가 급변하는 상황에서도 매매기회를 포착할 수 있도록 고안된 것입니다. 주문실행이 즉시 이루어진다는 것이 특징이며, 조건문 및 Stop, Limit 주문과 같이 쓰일 경우 고급전략의 수행이 가능합니다. This bar 주문의 경우, 반드시 유념하셔야 하는 주의사항이 몇 가지 있으며 본문 후반부 주의사항을 반드시 살펴보시길 바랍니다.

< This bar on close >

This bar on close 주문은 이전의 설명한 This bar 주문과는 다른 주문으로 "This bar on close"를 모두 명시해야 기능을 합니다. 이 주문은 현재봉의 증가로 주문을 수행하는 것으로, 백테스트 목적으로 고안된 기능입니다. 주문함수 사용시 [주문 실행 방법]을 생략할 경우 기본값으로 This bar on close 가 적용됩니다.

실시간에서 현재봉의 증가를 파악할 수 있는 시점은 다음봉의 시가가 형성된 이후이기 때문에 실시간에서 This bar on close 는 next bar at market 과 같게 됩니다. 백테스트 경우 갭의 발생 등으로 인해 현재봉의 증가와 다음봉의 시가가 다를 경우 This bar on close 와 next bar at market 의 주문실행결과는 다르게 됩니다.

< Market / Stop / Limit 의 차이점 >

1) Market

Buy 1 contract next bar at market 에서 Market 은 주문을 시장가 주문으로 발생시킵니다. Next bar at market 의 경우 다음봉 생성 직후(시가 형성 직후) 시장가 주문을 발생시키며, This bar at market 의 경우 주문함수 실행 즉시 시장가 주문을 발생시킵니다.

2) Stop

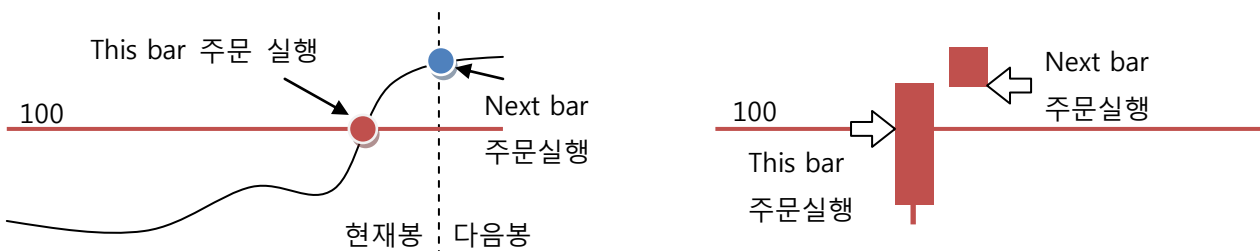
Stop 과 Limit 은 기준가격의 돌파 여부에 따라 주문을 실행합니다. 각 주문 별로 예를 들어보겠습니다.

Buy 주문에서 Stop 의 사용은 기준가보다 가격이 같거나 높을 때 Buy 주문을 실행시킵니다.

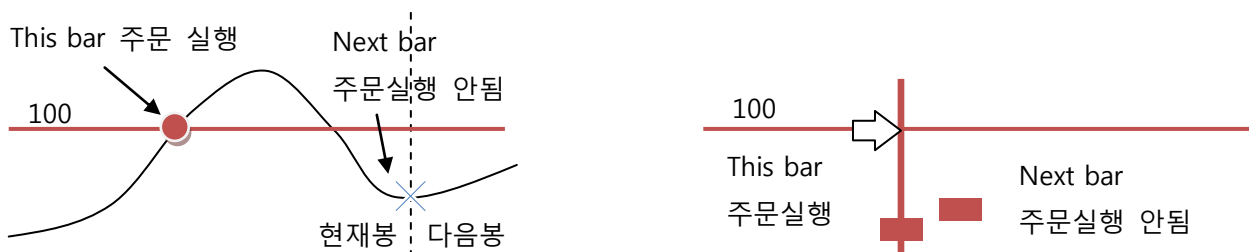
- 1) Buy 1 contract next bar at 100 Stop
- 2) Buy 1 contract this bar at 100 Stop

1)의 경우 Next bar 주문이므로 다음봉 가격이 판단의 대상가격이 되며, 이 가격이 100 보다 같거나 클 경우 매수주문을 실행하게 됩니다. 반면 2)의 경우 This bar 주문이므로 현재가가 판단의 대상가격이 되며 현재가가 100 보다 같거나 큰 경우 즉시 매수주문을 실행합니다.

< 현재가와 다음봉 시가 모두 기준가를 돌파하는 경우 >



< 현재가는 기준가 이상이지만, 다음봉 시가는 기준가 미만인 경우 >



3) Limit

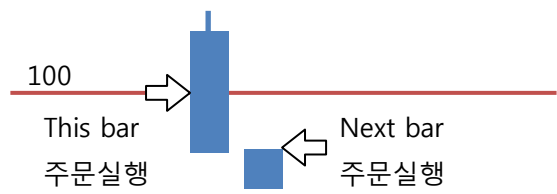
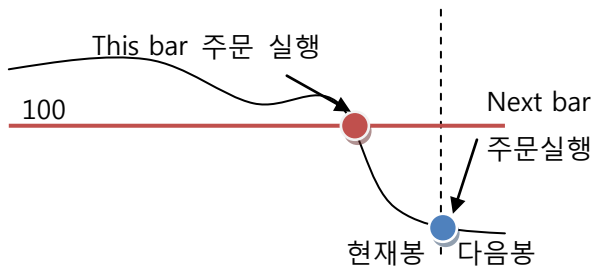
Buy 주문에서 Limit 의 사용은 기준가보다 가격이 같거나 낮을 때 Buy 주문을 실행시킵니다.

3) Buy 1 contract next bar at 100 Limit

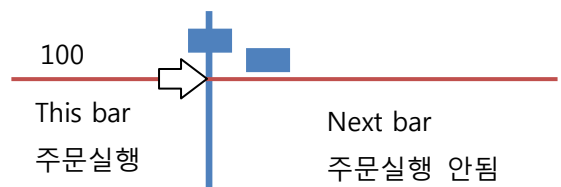
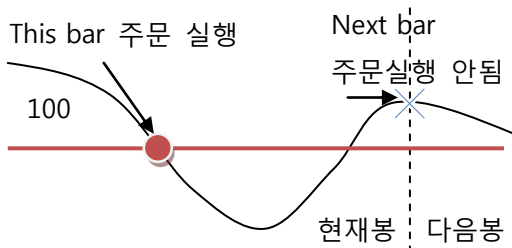
4) Buy 1 contract this bar at 100 Limit

3)의 경우 Next bar 주문이므로 다음봉 가격이 판단의 대상가격이 되며, 이 가격이 100 보다 같거나 낮을 경우 매수주문을 실행하게 됩니다. 반면 4)의 경우 This bar 주문이므로 현재가가 판단의 대상가격이 되며 현재가가 100 보다 같거나 낮은 경우 즉시 매수주문을 실행합니다.

< 현재가와 다음봉 시가 모두 기준가 이하인 경우 >



< 현재가는 기준가 이하이지만, 다음봉 시가는 기준가 보다 높은 경우 >



위의 예시에서는 Buy 주문에 대해 Stop 과 Limit 주문을 다루었습니다. Sell 의 경우에는 이와 반대로 작용합니다.

Sell 주문에서 Stop 의 사용은 기준가보다 가격이 같거나 낮을 때 Sell 주문을 실행시킵니다.
 Sell 주문에서 Limit 의 사용은 기준가보다 가격이 같거나 높을 때 Sell 주문을 실행시킵니다.

ExitLong 과 ExitShort 의 경우까지 포함하여 이를 표로 나타내면 아래와 같습니다.

주문구분	Stop	Limit
Buy	가격이 같거나 높을 때	가격이 같거나 낮을 때
Sell	가격이 같거나 낮을 때	가격이 같거나 높을 때
ExitLong	가격이 같거나 낮을 때	가격이 같거나 높을 때
ExitShort	가격이 같거나 높을 때	가격이 같거나 낮을 때

3. 유의사항

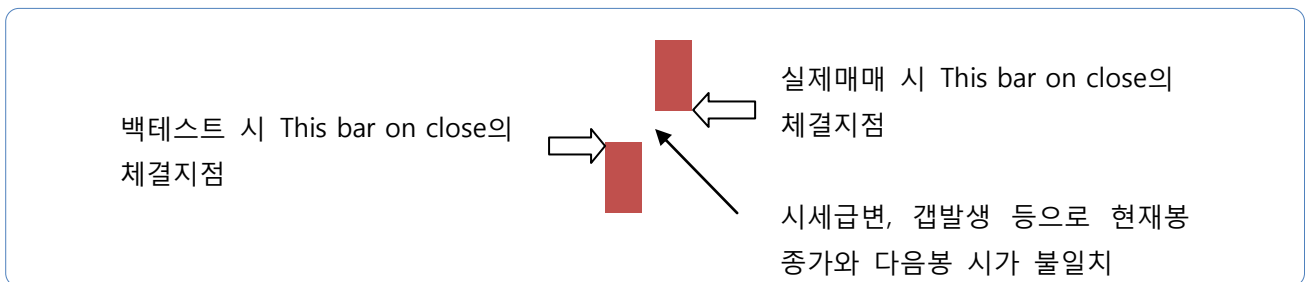
(1) 백테스트 관련 유의사항

This bar on close, This bar 주문은 주문의 특성 상, 실제 매매할 경우의 주문 결과와 백테스트의 주문 결과에 차이가 있을 수 있습니다. 그러한 현상이 발생하는 원인과 대처 방법을 살펴보겠습니다.

	Next bar	This bar
On close		This bar on close
Market	Next bar at Market	This bar at Market
Stop	Next bar at price Stop	This bar at price Stop
Limit	Next bar at price Limit	This bar at price Limit
백테스트와 실제매매	차이 없음	차이가 있는 경우 발생

< This bar on close >

This bar on close 주문은 실시간에서는 Next bar at market 으로 작용하나, 백테스트 시에는 현재봉 증가에 주문이 체결된 것으로 나타냅니다. 백테스트 시에는 완성된 봉 데이터를 미리 보유하고 있기 때문에 현재봉 증가 체결이 가능한 것으로 보이지만, 실제 매매에서는 다음봉 시가에 주문이 발생하는 것입니다. 그러므로 현재봉의 증가와 다음봉의 시가가 다른 봉 (시세급변, 갭 등으로 인한) 의 경우 This bar on close 주문의 백테스트 결과와 실제 매매 결과와는 차이를 보이게 됩니다.

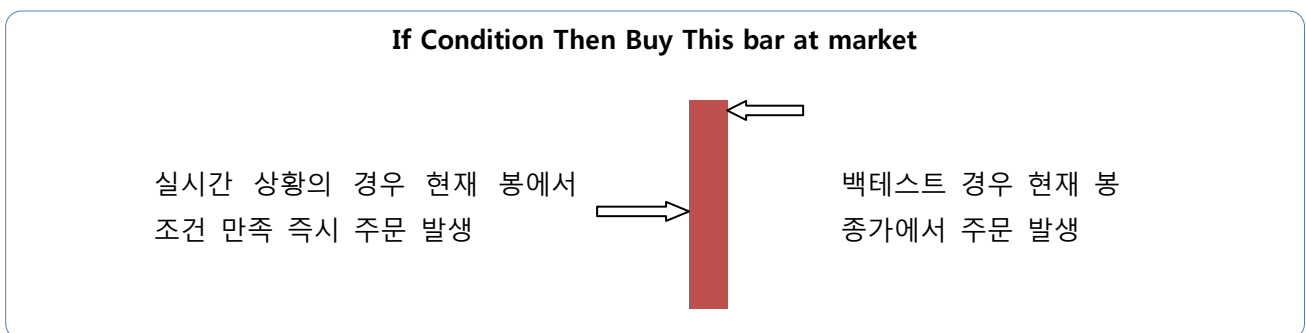


그러므로 일반적인 경우 전략을 검증하고자 하실 때 Next bar 주문을 사용하시는 것이 바람직하며, This bar on close 의 경우 전략 및 시장의 특성을 검증하실 경우에 한해 제한적으로 사용하시는 것이 좋습니다.

< This bar 주문 : This bar at market >

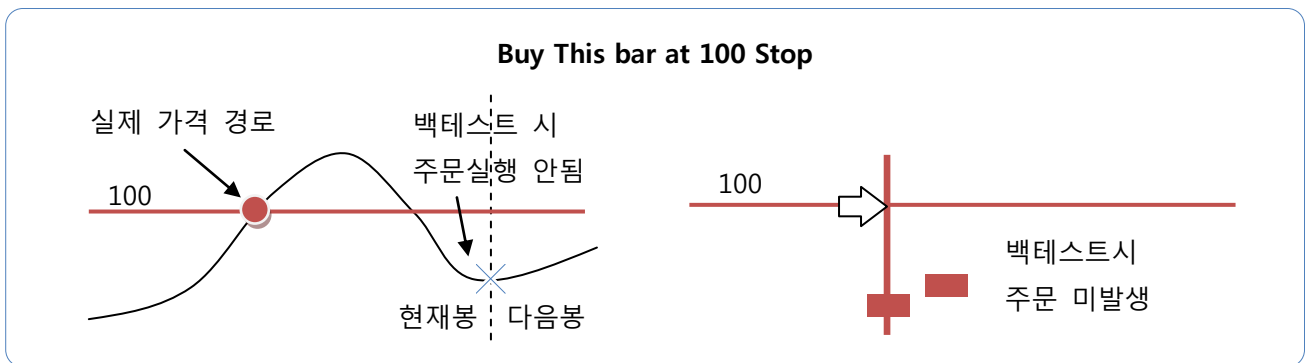
This bar at market 주문은 실시간 상황에서 주문 실행이 즉시 시장가로 이루어집니다. 하지만 백테스트의 경우, 주어진 데이터가 시가, 고가, 저가, 종가의 완성된 봉이고 가격이 어떤 경로를 통해 형성되었는지에 대한 정확한 정보 파악은 어렵습니다. Stage에서는 This bar at market 주문의 백테스트 경우 현재봉의 증가로 주문체결을 가정하며 이는 실시간 상황과 차이가 있습니다.

구체적인 예를 들어 살펴보겠습니다. 다음은 Condition 이 True 일 경우 Buy 주문을 실행하는 것입니다.



< Stop / Limit 주문 사용시 This bar 주문 >

앞서 Stop / Limit 주문에서 살펴보셨듯이 현재봉 내에서 가격 움직임에 따라 This bar at price Stop / This bar at price Limit 의 조건이 만족될 경우 즉시 주문이 발생하게 됩니다. 하지만 백테스트의 경우 This bar 주문은 현재봉의 종가와 기준가를 비교하여 판단한 후 기준가로 주문을 발생시키게 됩니다. 판단기준이 현재봉의 종가와 기준가이기 때문에 특정 상황에서는 백테스트와 실제상황의 불일치가 발생할 수 있습니다. 아래의 경우를 살펴보겠습니다.



이 경우 실제 가격 경로대로라면 기준가 100 을 넘어서면서 실제 주문이 발생하게 되지만, 백테스트의 경우 우측 그림과 같은 시가, 고가, 저가, 종가의 정보만이 있으며 종가 기준으로 판단하게 되기 때문에 조건을 만족하지 않아 주문이 발생하지 않습니다.

< 백테스트에서의 This bar 사용 유의 사항 및 보완법 >

지금까지 살펴보셨듯이, This bar on close 및 This bar 주문은 백테스트와 실제 매매의 결과에 차이가 있을 수 있습니다. 이러한 현상의 가장 근본적인 원인은 과거데이터에 있습니다. 우리가 보통 사용하는 가격데이터는 시가-고가-저가-종가의 정보를 담고 있는데, 이는 가격이 어떠한 경로로 움직였는지에 대한 정보를 모두 포함하고 있지 않습니다. 백테스트는 이러한 봉 데이터 기준(봉단위)로 이루어지고 실제 가격 경로를 완전하게 반영하지 않기 때문에, 실제 가격 경로에 적합한 주문인 This bar 주문 사용시에는 결과에 차이가 생기게 됩니다. 그렇기 때문에 백테스트에서 This bar 사용 시 다음을 유의하셔야 합니다.

1) 성과분석 결과에 왜곡이 있을 수 있다는 점을 유의

반복적으로 드리는 말씀이지만, This bar 주문을 사용하여 백테스트를 할 경우에는 실제 매매 상황과 다른 결과를 나타낼 수 있습니다.

2) 백테스트 시에는 This bar 주문 대신 Next bar 주문을 사용

전략을 작성할 때 백테스트 시에는 Next bar 주문을 사용하도록 작성할 수 있습니다.

예약어	설명
Realtime	현재 SStage 차트의 데이터가 실시간데이터 인지 과거데이터(백테스트)인지를 판별하여, 실시간일 경우 TRUE 값을 갖습니다.

Realtime 예약어를 활용하여 아래와 같이 한가지 지표에 실시간, 백테스트 시 주문을 구분할 수 있습니다.

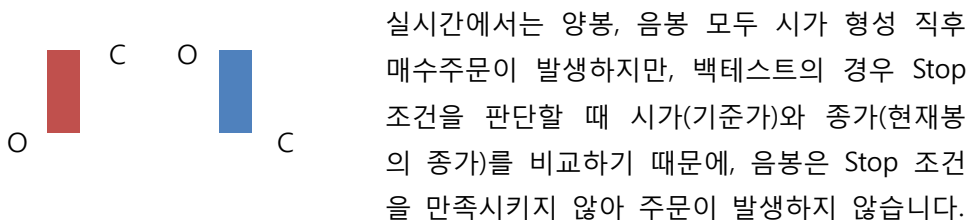
```

If Realtime Then
    Buy this bar at market
Else
    Buy next bar at market
End If
    
```

위의 구문은 SStage 차트가 실시간 상황일 경우 Buy this bar at market 을 실행하고 백테스트 시에는 Buy next bar at market 을 실행합니다. If Realtime Then ~~ 을 활용하면 실제 매매 경우와 백테스트 시 This bar 주문의 사용으로 인한 차이를 수정할 수 있습니다.

이번에는 Stop 과 Limit 의 경우를 살펴보겠습니다. Buy This bar at open stop 을 예로 들어보겠습니다. 현재봉의 시가보다 같거나 높으면 매수주문이 발생합니다. 실시간에서는 현재봉 생성 직후 시가 형성과 동시에 현재가(종가)도 시가와 같으므로 즉시 매수주문이 발생하게 됩니다. 실시간에서 위 구문은 현재봉 생성과 동시에 매수주문을 발생하게 됩니다.

하지만 백테스트의 경우는 기준가와 현재봉의 증가로 판단하기 때문에 "증가 < 시가"인 봉에서는 위의 Buy This bar at open stop 주문이 실행되지 않습니다.



이 경우, 실제상황에서는 항상 매수주문을 발생시키는 것이 목적이므로 다음과 같이 전략을 구현할 수 있습니다.

```
If Realtime or C > O Then
    Buy This bar at open stop
Else
    Buy This bar at open limit
End If
```

위의 구문은 STage 차트가 실시간이거나 증가 > 시가인 경우 Buy This bar at open stop 을 실행시키며, 백테스트 상황에서 C > O 가 아닌 경우에는 Buy This bar at open limit 을 실행시킵니다. 백테스트 상황에서 C > O 가 아닌 경우에는 Stop 조건은 만족시키지 못하고 Limit 조건은 만족이 됩니다. 지금 전략의 목적은 항상 매수주문이 발생하는 것이므로 백테스트 음봉의 상황에서 Limit 를 사용하여 백테스트 시 주문실행이 일어나게 할 수 있습니다.

이러한 방법을 활용하면 This bar 주문 사용시 발생할 수 있는 백테스트와 실시간의 결과 차이를 해결할 수 있습니다.

(2) 실시간에서 주문함수 사용시 유의사항

매수주문, 매도주문이 동시에 발생하지 않도록 반드시 확인

< This bar 주문의 경우 각별한 주의 요함 >

많은 전략에서 주문함수는 조건문과 함께 쓰입니다. 조건문, Stop, Limit 과 함께 주문함수를 실행할 경우, 매수와 매도주문이 동시에 발생하지 않도록 전략작성에 주의를 해야 합니다.

예를 들어 아래와 같은 경우에는 매수/매도주문의 동시 발생이 일어나지 않습니다.

```
If Condition Then
    Buy next bar at market
Else
    Sell next bar at market
End
```

여기서 Condition 은 Boolean 변수로 True 또는 False 의 값을 가지며 시장상황에 따라 Condition 의 값은 True, False 중 한가지이기 때문에 Buy 와 Sell 중 한가지만 수행되게 됩니다.

다음의 경우에도 매수/매도주문의 동시 발생이 일어나지 않습니다.

```
bCrossUp = MA( C, 5 ) Cross Over MA( C, 20 )
bCrossDn = MA( C, 5 ) Cross Under MA( C, 20 )
If bCrossUp Then
    Buy next bar at market
End If
If bCrossDn Then
    Sell next bar at market
End If
```

하지만 다음 예의 경우에는 매수/매도주문의 동시 발생이 일어날 수 있습니다.

```
bCrossUp = MA( C, 5 ) Cross Over MA( C, 20 )
bCrossDn = MA( C, 5 ) Cross Under MA( C, 60 )
If bCrossUp Then
    Buy next bar at market
End If
If bCrossDn Then
    Sell next bar at market
End If
```

위의 구문의 bCrossUp 은 5 일 이평과 20 일 이평을 기준으로 삼지만, bCrossDn 은 5 일 이평과 60 일 이평을 기준으로 삼습니다. 시장의 상황에 따라서 5 일 이평은 20 일 이평을 Cross Over 하지만 60 일 이평은 Cross Under 일 수도 있습니다. 이렇게 될 경우 두 가지 IF 문이 모두 실행되게 되며, Buy 주문 실행과 동시에 Sell 주문이 실행됩니다. 위의 경우는 next bar 주문이기 때문에 다음봉 시가에서 매수/매도 동시주문이 발생하게 됩니다.

만약 위의 구문이 next bar 가 아닌 This bar 일 경우, 매우 심각한 결과를 초래하게 됩니다. This bar 주문은 현재봉에서 조건 만족 즉시 주문을 실행하기 때문에 계속해서 Buy 주문과 Sell 주문을 실행하게 됩니다.

전략을 작성할 때, 매수주문과 매도주문이 동시에 발생하는 일이 없도록 반드시 주의해야 합니다. 특히 This bar 주문을 사용할 경우, 작은 실수가 치명적인 결과로 이어질 수 있습니다.

이와 관련하여 매수/매도주문이 동시 발생하는지 확인하는 방법은 몇 가지가 있습니다. 첫 번째로 PlotPoint 함수를 사용하여 STage 차트 상에서 눈으로 직접 확인하는 것입니다. 앞의 예문을 아래와 같이 수정해보겠습니다.

```
bCrossUp = MA( C, 5 ) Cross Over MA( C, 20 )
bCrossDn = MA( C, 5 ) Cross Under MA( C, 20 )
If bCrossUp Then
    PlotPoint1(BottomSide, "B1")
// Buy next bar at market
End If
If bCrossDn Then
    PlotPoint2(TopSide, "S1")
// Sell next bar at market
End If
```

이렇게 지표를 작성하게 되면, 주문함수는 실행하지 않은 채 조건문이 만족되는 부분에 PlotPoint 함수로 표시를 하게 됩니다. 만약 "B1" 표시와 "S1" 표시가 동시에 발생하는 붕이 존재한다면 매수/매도주문이 동시 발생하는 것입니다.