

آموزش برنامه نویسی به زبان MQL4
دوست خوبمون MDA7 عزیز زحمت تایپ رو کشیدند و منم تصمیم گرفتم مطالب رو جمع
آوری و ویرایش کنم تا به عنوان یه رفرنس باشه.
امیدوارم برا بقیه دوستان مفید واقع بشه

پست شماره 1 - مقدمه

دوستان با اجازتون اولین پست رو در خصوص تایپ رو شروع میکنم
در ابتدا بیاید کمی در مورد تفاوت های ماشین و انسان فکر کنیم ، انسان موجودی خارق العاده هست که همه
کار ازش برمیاد حتی کارهایی که قبلها به نظر انسانهای اون زمان چیزی غیر قابل انجام بوده!
در قبال این همه قابلیت ذهنی و فکری قابلیت های مثل دوست داشتن ، ترسیدن ، شادی ، فداکاری و هم
داریم.

در مورد بازاری که توش داریم فعالیت میکنیم بعضی از این خصوصیات به نفعمون هستند ولی متأسفانه
بیشترشون به ضررمون تموم میشند.

ولی ماشین برخلاف این همه توانایی های انسانی هیچ توانایی از خودش نداره
ماشین فقط قادر هست که تنها دستوری که از ما میگیره رو اجرا کنه. ولی در عوض اون احساساتی که ما ها
داریم و بیشتر باعث ضررهامون میشن رو ماشین هیچکدومشون رو حالیش نیست.
نه عصبانی میشه ، نه ناراحت ، نه خوشحال ، نه سرخورده ، نه میدونه پولدار شدن یعنی چی و نه پول از
دست دادن....

همون موجود خارق العاده ای در موردش صحبت کردیم یا یک نگاه به چارت شاید بیش از چند هزار دستور رو به
طور نا خود آگاه توی ذهنمونش اجرا میکنه.

فردی رو در نظر بگیرید که داره تعلیم رانندگی میبینه ، اول باید صندلیشو درست کنه بعد آینه ها رو بعد کلاچ رو
پیدا کنه و و با هزار مصیبت بزنه دنده یک بعد ماشین خاموش شه دوباره روز از نو و روزی از نو تا اینکه
بالاخره شانسی ماشین راه بیفته.

این فرد داره همه چیز رو اصولی و همون طور که بهش یاد داده شده رو یکی یکی اجرا میکنه تا ماشین رو
حرکت بده، ولی یک کسی که چندین سال هست رانندگی میکنه از یک طرف کشور تا طرف دیگش میره در
حالی که داره صحبت میکنه ، چایی میخوره ، تخمه میشکنه ، بعضی وقتها هم پشت رول چرت میزنه ...
این فرد دوم داره دنده ها رو یکی یکی عوض میکنه ولی بدون اینکه خودش دقیقاً متوجه نحوه کار کردنش باشه
، در واقع داره از ذهن ناخود آگاهش استفاده میکنه.

خلاصه سرتون رو درد نیارم ، برای اینکه کد نویس خوبی بشید اول باید سعی کنید مثل ماشین فکر کنید. کاری
رو که دارید بصورت نا خود آگاه روی چارت انجام میدید رو به صورت تک تک اول برای خودتون مشخص کنید که
مثلاً من که شروع میکنم به ترید اولین کاری که انجام میدم این هست دوم ... سوم ... و در آخر با همه این کار

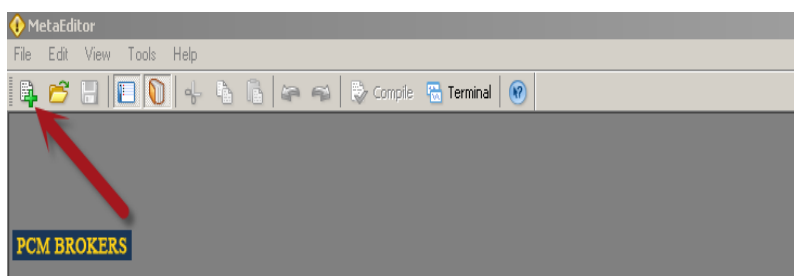
ها خواهیم رسید به یک سیگنال بای یا سل یا بستن پوزیشن.
پس به نظر من مهمترین کار توی نوشتن کد این هست که بتونیم کاری رو که انجام میدیم رو به صورت خط به خط و به صورت اولویت برای خودمون مشخص کنیم.

پست 2 - آشنایی با محیط MetaEditor - قسمت اول

از قسمت بالای متاتریدرتون روی گزینه MetaEditor کلیک کنید تا وارد قسمت کدنویسی متا شوید



برای ایجاد فایل جدید روی New کلیک کنید



از این قسمت باید Expert یا indicator یا Script یا که میخواهید بنویسید رو انتخاب کنید



پس از ایجاد اسم برای فایل صفحه زیر باز میشه که در واقع شامل 4 قسمت هست که مشخص کردم

```
//+-----+
//|                                     MDA7_Expert EX1.mq4
//|                                     MDA7
//|                                     FX.MDA7@YAHOO.CO.UK
//+-----+
#property copyright "MDA7"
#property link      "FX.MDA7@YAHOO.CO.UK"
1
//+-----+
//| expert initialization function
//+-----+
2
int init()
{
//----
//----
return(0);
}
//+-----+
//| expert deinitialization function
//+-----+
3
int deinit()
{
//----
//----
return(0);
}
//+-----+
//| expert start function
//+-----+
4
int start()
{
//----
//----
return(0);
}
PCM BROKERS
```

پست 3 - آشنایی با محیط MetaEditor - قسمت دوم

در این پست میخوام همون 4 تا ناحیه ای رو که در آخرین عکس شکل قبل بود رو با هم بررسی کنیم
ناحیه 1 قسمتی هست که در اونجا متغیر ها رو تعریف میکنیم و یا فایل یا یک کتابخانه رو فراخونی میکنیم.
ناحیه های بعدی کلا شامل 3 تا ناحیه هست که در واقع هر کدام یک تابع هستند که کار خاصی را در برنامه انجام میدن

قبل از توضیح دادنشون بگذارید توضیح بدم که یک برنامه چطور کار میکنه!

همونطور که قبلا گفتم ماشین هیچ کاری از دستش بر نیامد و هیچ چیزی رو نمیشناسه مگر اینکه ما بهش بفهمونیم

در ناحیه 1 هر چیزی رو که میخوایم در برنامه استفاده کنیم رو باید براش تعریف کنیم .

بعد از این که برنامه اجرا رو یا همون متغیر ها رو شناخت ، میره سرغ قسمتهای بعدی (برنامه همیشه از بالاترین سطر شروع میشه و خط به خط میره تا برسه به انتهای آخرین سطر)

ولی در اینجا فکر بعضی از مسایل رو قبلا کردند ، مثلا یک چیزی که از برنامه میخوایم ، لازم هست که فقط 1 بار در ابتدا اجرا بشه و لازم نیست که هر لحظه اون اطلاعات محاسبه بشن.

فرض کنید میخواید به برنامه بگید که اسم کمپانی یا همون بروکرتون رو چک کنه.

اسم کمپانیتون همیشه ثابت هست و نمیتونه که بعدا تغییر کنه و اگر این کد رو در جایی قرار بدید که هر لحظه بخواید چک بشه فقط سرعت کدتون رو پایین میارید.

ناحیه 2 یا همون `int init()` قسمتی هست که برای کدهایی ساخته شده که فقط یک بار لازمه که محاسبه بشن و وقتی روی چارت اجرا بشه فقط همون لحظه اول اجرا میشه و اعدادش تا آخر کار برنامه همون چیزی خواهد بود که یک بار محاسبه شدند.

ناحیه 3 یا همون `int deinit()` قسمتی هست که اون هم مثل قبلی فقط یک بار اجرا خواهد شد و اون هم در پایان کار برنامه.

اگر در اندیکاتوهایی که تا بحال استفاده کردید دقت کرده باشید وقتی اون رو از چارت حذف میکنید ، تمام خط و خطوطی که روی چارت انداخته بود باحذف اندیکاتور پاک میشندن.

شما هم کدهایی رو که مایلید در زمان حذف برنامه از چارت اجرا بشند رو در این قسمت باید بنویسید مثلا پاک کردن هر خطی که تا بحال ایجاد شده!

ناحیه 4 یا همون `int start()` قسمتی هست که با ایجاد شدن هر تیک در بازار از ابتدا تا انتها یک بار اجرا خواهد شد (وقتی یک قیمت جدید در چارت ظاهر میشه در واقع یک تیک به بازار اومده ، وقتی Volume یک کندل برابر 820 هست یعنی در این کندل 820 تا تیک داشتیم پس Start هشتصد و بیست بار در این کندل اجرا شده (820 بار از ابتدای start تا انتهای اون)

پس با توجه به چیزایی که نوشتیم در داخل این قسمت محاسباتی انجام میشن که نیاز داریم دائما اعداد جدیدی رو بدست بیاریم

فرض کنید های یا لوی کندل حال.

پست 4 - انواع داده - قسمت اول

بیا ببینیم کم کم شروع کنیم و سعی کنیم که مثل ماشین فکر کنیم تا بتونیم ایده هامون رو تبدیل به کدی بکنیم که ماشین بتونه اون رو بخونه و انجام بده.

ما چه چیزهایی رو میتونیم با یک کیبورد به ماشین بفهمونیم و چند نوی میتونن باشن؟

عدد ، نوشته ، آره یا نه ، رنگ و زمان که اصطلاحاتشون به این صورت هستن:

پست 5 - انواع داده - قسمت دوم

در پست قبلی انواع `int` و `double` رو باهم چک کردیم ، در این پست انواع دیگه رو خیلی سریع باهم جلو میریم تا اینکه از پست بعدی بقیه مطالب رو با نوشتن یک کد کامل جلو ببریم

البته نوشتن یک استراتژی بسته به قدرت برنامه نویسی فرد و تجربش خیلی کمتر از مدت زمانی خواهد بود که ما صرف آموزش این استراتژی خواهیم کرد

و در حین نوشتن پله پله توضیح خواهیم داد بخاطر همین کمی طولانی به نظر خواهد رسید ولی صبور باشید

چون بعد از نوشتن یک استراتژی و فهم دقیق اون شاید بعدش با کمی حوصله و پشت کار بتونید خودتون

کدهای مشابه رو بدون کمک گرفتن از دیگران بنویسید

بریم سراغ داده ها

3- Boolean یا همون داده های منطقی برای داده هایی مثل آره و نه استفاده میشه یا همون `true / false`

بعنوان مثال :

```
Bool show_lines=true;
```

همون طور که قبلا عرض کردم در ابتدا بایستی نوع داده رو به برنامه بشناسونیم در این جا هم `bool` نوشتیم تا

برنامه بفهمه که این داده از نوع Boolean هست

`Show_Lines` یک اسم هست که به این داده گذاشتیم و مقدار اون رو هم `true` (میتونست `false` باشه)

اندیکاتورهایی رو به خاطر بیارید که به چیز مشابه این تنظیم رو در ورودی دارند و با ترو فالس کردن خطوطی به

چارت اضافه یا کم میشه

بعدا در مثال عملی بیشتر کار خواهیم کرد

4- String داده های خطی هستند که به اینصورت نوشته میشه:

```
String my_name_is="mda7";
```

برای دادن مقدار به داده های خطی حتما باید اونها رو داخل دو تا کوتیشن (" ") بنویسیم

5- Color داده های مربوط به رنگ هست ، به این صورت :

```
color pivot_color=white;
```

متغیر رنگی به نام `Pivot_Color` تعریف کردیم و به اون مقدار سفید دادیم

* : البته برای رنگ به جای `White` میتونیم عدد مربوط به سفید رو هم قزاز بدیم ولی من توصیه نمیکنم چون

همینی که اسم رنگ رو بنویسیم راحت تر از حفظ کردن شماره عددی رنگ هست.

6- `DateTime` داده زمان هست در واقع یک عدد طولانی هست که از ابتدای سال 1970 به هر یک ثانیه یکی به

اون اضافه شده و در این لحظه که من تایپ میکنم به عدد 1389353082 رسیده و این کار تا آخر سال 2037

ادامه خواهد داشت

datetime Time_Candle = iTime(Symbol(),0,10);

در اینجا ما یک متغیر از نوع datetime. با نام Time_Candle تعریف کردیم و اون رو برابر کندل دهم قرار دادیم. از طولانی بودن مقدار نترسید باز بعدا در موردش صحبت خواهیم کرد ولی مختصرا میگم iTime خودش یک تابع هست که به صورت پیشفرض در متا قرارداده شده و همون عدد طولانی که در موردش صحبت کردیم رو برمیگردونه

وقتی جلوی یک عبارتی () میبینید اصولا خودش باید که یک تابع باشه و میبینید که جلوی iTime هم پرانتز باز و بسته داریم و اما عبارات داخل پرانتز :

Symbol باز یک تابع هست که نام جفت ارزی رو که اکسپرت روی اون هست رو برمیگردونه (به علامت پرانتز جلوی Symbol دقت کنید)

0 : یعنی هر تایم فریمی که ران باشه ، حالا اگر این مثلا میگذاشتیم 240 اون وقت میرفت و تایم H4 رو میخوند.

10: یعنی کندل دهم از سمت راست به چپ چارت

همیشه کندلی که در حال بازی کردن هست و هنوز بسته نشده رو بعنوان کندل 0 میشناسیم کندل بعدیش میشه 1 و الی تا آخر.

پست 6 - استراتژی میانگین متحرک و macd قسمت اول

ترکیب میانگین متحرک و MACD

Moving Average MACD Combo

در تئوری, معامله کردن بر اساس تعقیب روند بسیار ساده است . تنها کاری که باید انجام دهید این است « وقتی قیمت در حال صعودی است خرید داشته باشید و وقتی قیمت در حال نزول است معامله فروش». ولی در عمل چنین معاملاتی بسیار مشکل است وقتی به دنبال یک فرصت معاملاتی برای معامله براساس روند هستید ممکن است این سوالات به ذهن شما خطور کند :

جهت روند چیست ؟

آیا الان باید با روند همراه شوم یا منتظر یک حرکت بازگشتی بمانم ؟

چه موقع این روند تمام می شود ؟

بزرگترین واهمه یک معامله گر براساس روند این است که بسیار دیر با روند همراه می شود.

یعنی در نقطه ای که روند رو به افول است . با وجود این مشکلات معامله گری بر اساس روند

یکی از مورد توجه ترین سبک های معاملاتی است . چرا که وقتی یک روند کوتاه مدت یا بلند مدت

شکل می گیرد می تواند برای ساعت ها روز ها و حتی ماه ها ادامه داشته باشد .

ما برای این منظور شیوه ای طراحی کردیم که ضمن پاسخگو به سولات فوق می تواند نقاط ورود و خروج را نیز

دقیقا مشخص نماید. این استراتژی استراتژی ترکیبی میانگین متحرک و MACD نام دارد . ما دو میانگین را در استراتژی خود به کار گرفته ایم : میانگین متحرک ساده 50 دوره ای یا SMA50 و میانگین متحرک ساده 100 دوره ای یا SMA100 . عملا دوره مناسب برای میانگین بستگی تام به چارچوب زمانی مورد استفاده دارد. ما این استراتژی را برای چارچوب های زمانی یک ساعته و روزانه طراحی کرده ایم . SMA50 به عنوان ماشه برای تعیین نقطه ورود به کار می رود . در حالی که SMA100 به ما جهت صحیح روند را نشان می دهد در این استراتژی ما فقط وقتی وارد معامله خرید یا فروش می شویم که قیمت میانگین های متحرک را در جهت روند قطع کند . با وجود اینکه این استراتژی از منطقی مشابه استراتژی Momo بهره می برد ولی بسیار هوشیارانه تر از آن است و از میانگین متحرک های بلند مدت در چارچوب زمانی یک ساعته و روزانه برای کسب سود های بلند مدت تر بهره می گیرد.

قوائد ورود به معامله خرید :

(1) در موقعی که قیمت کمتر از هر دو میانگین متحرک 50 و 100 دوره ای است بازار را زیر نر بگیرید .
(2) هرگاه قیمت یکی از میانگین متحرک های را به سمت بالا قطع کرده و از آن 10 پیپ فاصله گرفت به اندیکاتور MACD نگاه کنید . اگر این اندیکاتور طی میله اخیر از منفی به مثبت تغییر جهت داده است وارد معامله خرید شوید.

(3) حد زیان را در نقطه حداقل 5 میله قبلی قیمت قرار دهید .
(4) نیمی از معامله را با سود دو برابر مخاطره ای که پذیرفته اید ببندید و در نیم دیگر حد توقف را به نقطه ورود منتقل فرمایید.

(5) نیمه دوم معامله را وقتی ببندید که قیمت SMA50 را به سمت پایین شکسته و 10 پیپ با آن فاصله گرفته باشد .
قواعد ورود به معامله فروش :

(1) در موقعی که قیمت بیش از هر دو میانگین متحرک 50 و 100 دوره های است بازار را زیر نظر بگیرید.
(2) هرگاه قیمت یکی از میانگین متحرک های راست به سمت بالا قطع کرد و از آن 10 پیپ فاصله گرفت به اندیکاتور MACD نگاه کنید اگر این اندیکاتور طی 5 میله اخیر از مثبت به منفی تغییر جهت داده اسن وارد معامله فروش شوید.

(3) حد زیان را در نقطه حداکثر 5 میله قبلی قیمت قرار دهید.
(4) نیمی از معامله را با سود دو برابر مخاطره ای پذیرفته اید ببندید و در نیم دیگر حد توقف را به نقطه ورود منتقل فرمایید.

(5) نیمه دوم معامله را وقتی ببندیم که قیمت SMA50 را به سمت بالا شکسته و 10 پیپ با آن فاصله گرفته باشد .

توجه : اگر قیمت بین دو میانگین متحرک 50 و 10 دوره‌ای است وارد معامله نشوید.

همونطور که قبلا گفته بودم باید در ابتدا همه چیزهایی رو که لازم داریم در ابتدای برنامه و قبل از تابع `init int` تعریف و نامگذاری کنیم

همونطور که از نوشته معلوم هست ما 2 تا موینگ احتیاج داریم (`SMA50` و `SMA100`) برای راحتی کار همین اسمهارو استفاده میکنیم

و یک `MACD` هم نیاز داریم.

خوب حالا باید نوع خروجی اینها رو بدینیم که آیا عدد صحیح هست یا اعشاری یا رنگ یا
هر اندیکاتوری معمولا یک یا چند عدد رو در خروجی به ما میده حالا اگر روی چارتتون موینگ بندازید و `MACD` وقتی روی مثلا کندل دهم روی موینگ موس رو نگه دارید یک عدد اعشاری خواهید دید که این عدد در واقع مقدار موینگ همون کندل دهم هست پس موینگ ما باید از نوع `double` باشه

همین کار رو روی `MACD` انجام بدید باز هم عدد اعشاری خواهیم دید که این عدد یا زیر صفر (منفی) خواهد بود و یا بالای صفر (مثبت).

پس تا اینجا ما 3 تا متغیر به نامهای `SMA100` - `SMA50` و `MACD` که هر سه از نوع `double` هستند رو تعریف خواهیم کرد ولی بحث اینجاست که مقادیر اونها رو چه باید بدیم!!!!

با کمی فکر در مورد نوشته های قبلی در مود تابع `init` و `deinit` و `start` یادمان خواهد افتاد که محاسباتی که لازم هست با حرکت چارت از نو حساب بشند رو باید داخل تابع `start` محاسبه کنیم.

پس در ابتدای برنامه این 3 تا رو تعریف میکنیم بدون اینکه مقدار دهی کنیم و بعدا در داخل تابع `start` مقدار دهی خواهیم کرد.

```
double SMA50;
```

```
double SMA100;
```

```
double MACD;
```

همون طور که میبینید از علامت = استفاد ه نکردیم تا بعدا در تابع start مقدار دهی کنیم

```
//+-----+-----+-----+-----+-----+-----+-----+-----+-----+
//|                                                                 MDA7- Moving & MACD.mq4 |
//|                                                                 MDA7 |
//|                                                                 FX.MDA7@YAHOO.CO.UK |
//+-----+-----+-----+-----+-----+-----+-----+-----+-----+
#property copyright "MDA7"
#property link      "FX.MDA7@YAHOO.CO.UK"

double SMA50 :
double SMA100 :
double MACD :

//+-----+-----+-----+-----+-----+-----+-----+-----+-----+
//| expert initialization function |
//+-----+-----+-----+-----+-----+-----+-----+-----+-----+
int init()
{
//-----
//-----
return(0);
}
//+-----+-----+-----+-----+-----+-----+-----+-----+-----+
//| expert deinitialization function |
//+-----+-----+-----+-----+-----+-----+-----+-----+-----+
int deinit()
{
//-----
//-----
return(0);
}
//+-----+-----+-----+-----+-----+-----+-----+-----+-----+
//| expert start function |
//+-----+-----+-----+-----+-----+-----+-----+-----+-----+
int start()
{
//-----
//-----
return(0);
}
PCM BROKERS
```

پست 7 - استراتژی میانگین متحرک و macd قسمت دوم

حالا باید بریم سراغ تابع start و متغیرهامون رو مقدار دهی کنیم

در واقع ما خروجی 3 تا اندیکاتور رو نیاز داریم , حال باید بدونیم که چطور میشه یک اندیکاتور رو داخل یک

اکسپرت فراخونی کرد تا بعدا از عدد اون برای محاسبات بعدی استفاده کرد!!

ما یک سری اندیکاتورها رو داریم که شرکت سازنده متاتریدر به صورت دیفالت اونها رو داره و بعضی از اندیکاتورها

نوشته کسان دیگه هستند (Custom Indicators) خوشبختانه این 3 تا عددی که ما لازم داریم همشون از نوع

اندیکاتورهای دیفالت متاتریدر هستند

یک لیستی از توابع فراخونی اندیکاتورهای متا رو براتون مینویسم:

iAC

iAD

iAlligator

iADX

iATR

iAO

iBearsPower

iBands

iBandsOnArray

iBullsPower

iCCI

iCCIONArray

iCustom

iDeMarker

iEnvelopes

iEnvelopesOnArray

iForce

iFractals

iGator

iIchimoku

iBWMFI

iMomentum

iMomentumOnArray

iMFI

iMA *****

iMAOnArray

iOsMA

iMACD *****

iOBV

iSAR

iRSI

iRSIONArray

iRVI

iStdDev

iStdDevOnArray

iStochastic

iWPR

از لیست فوق ما به دو تابع iMA (برای بدست آوردن مقدار موینگ اوریج) و iMACD (برای بدست آوردن مقدار MACD) نیاز خواهیم داشت

1- تابع iMA :

double iMA(string symbol, int timeframe, int period, int ma_shift, int ma_method, int applied_price, int shift)

مقدار موینگ کندل (shift) رو بهمون میده و پارامترهای مختلفی رو داره که باید به دقت پر بشن

String symbol

با مقدار دهی این قسمت شما به تابع میگوید که کدوم جفت ارز رو بررسی کنه! اگر دقت کنید خودش از نوع خطی هست برای اینکه مقدار مربوط به همون ارزی رو که اکسپرت روش ران خواهد بود رو برگردونه کافی Symbol یا بنویسید NULL() هست که در این قسمت بنویسید

رو براتون میخوند. AUDUSD" اونوقت مقدار موینگ در چارت AUDUSD اگر مثلا به جای اینها مینوشتید "

timeframe int

با دادن مقدار در این قسمت مشخص میکنید که کدوم تایم فریم مد نظرتون هست! اگر 0 باشه همون تایمی که اکسپرت ران هست رو خواهد خوند ولی اگر مثلا بگذارید 1440 میره و تایم فریم دیلی رو نگاه میکنه

PERIOD_M1	1	.minute 1
PERIOD_M5	5	.minutes 5
PERIOD_M15	15	.minutes 15
PERIOD_M30	30	.minutes 30
PERIOD_H1	60	.hour 1
PERIOD_H4	240	.hour 4
PERIOD_D1	1440	.Daily
PERIOD_W1	10080	.Weekly
PERIOD_MN1	43200	.Monthly
0 (zero)	0	.Timeframe used on the chart

هست که مربوط به اعداد صحیحه.int همونطور که میبینید نوع تایم فریم ار نوع

int period:

از اینجا به بعد مربوط میشه به تنظیمات موینگ در اینجا ما موینگ 50 و 100 رو استفاده خواهیم کرد.

int ma_shift:

باید 0 بزاریم چون نمیخواهیم که موینگ ما شیفت داشته باشه.

int ma_method :

نوع موینگ رو براش مشخص میکنیم که در اینجا ما Simple Moving لازم داریم که اگر اینپوت موینگ اوریج رو نگاه کنید میبینید که این در واقع اولین گزینه هست

یادتون باشه در برنامه نویسی معمولا اولین مقدار 0 هست نه 1 ! پس در اینجا هم 0 قرار خواهیم داد تا Simple Moving رو برامون اعمال کنه.

int applied_price

باز اگر تنظیمات اینپوت موینگ اوریج رو نگاه کنید به صورت دیفالت بر روی Close هست که عدد مربوطش به علت اینکه اولین هست باز هم 0 خواهد بود.

int shift:

آخرین پارامترمون هست که با گذاشتن عدد در اینجا به تابع میگیم که ما مقدار موینگ اوریج کدوم کندل رو ارزش میخواهیم که باز در اینجا ما مقدار موینگ کنلی رو که در حال بازی کردن هست رو نیاز داریم پس میشه کندل 0. با این اطلاعاتی که تا حالا یاد گرفتیم میتونیم متغیرهای موینگ 50 و 100 رو در تابع start تعریف کنیم. به این

صورت:

```
MDA7- Moving & MACD.mq4
MDA7
FX.MDA7@YAHOO.CO.UK

//-----
//| property copyright "MDA7"
//| property link      "FX.MDA7@YAHOO.CO.UK"
//-----
double SMA50 ;
double SMA100 ;
double MACD ;

//-----
//| expert initialization function
//-----
int init()
{
//-----
return(0);
}

//-----
//| expert deinitialization function
//-----
int deinit()
{
//-----
return(0);
}

//-----
//| expert start function
//-----
int start()
{
//-----
SMA50 = iMA(Symbol(), 0 , 50 , 0 , 0 , 0 , 0);
SMA100 = iMA(Symbol(), 0 , 100 , 0 , 0 , 0 , 0);
//-----
return(0);
}

PCM BROKERS
```

پست 8 - استراتژی میانگین متحرک و macd قسمت سوم

نوبت میرسه به گرفتن اطلاعات از اندیکاتور MACD و تعریف اون توی تابع start

```
double iMACD( str symbol, int timeframe, int fast_ema_period, int slow_ema_period, int signal_period, int  
applied_price, int mode,int shift)
```

پارامترهای Symbol و timeframe که مشخص هست

از fast_ema_period به بعد مربوط به تنظیمات MACD هست اگر اینپوت اندیکاتور رو نگاه کنید به ترتیب

12

26

9

Apply to Close

رو میبینید که پارامتر آخر (Apply to Close) چون اولین هست پس صفر خواهد بود.

حالا میرسیم به پارامتر mode:

اندیکاتور MACD دو مقدار رو به ما برمیگردونه اولین مقدار عدد اعشاری مربوط به میله های MACD. هست و

عدد دوم مربوط به خط موینگی که روی MACD (قرمز رنگ) میفته هست

که ما در اینجا به عدد هیتوگرامهای مکدی نیاز داریم و چون اولین مقدار هست باز صفر خواهد گرفت.

الان میتونیم مقدار MACD کندل صفر رو به این صورت تعریف کنیم:

```

//+-----+
//|                                     MDA7- Moving & MACD.mq4                                     |
//|                                     MDA7                                     |
//|                                     FX.MDA7@YAHOO.CO.UK                                     |
//+-----+
#property copyright "MDA7"
#property link      "FX.MDA7@YAHOO.CO.UK"

double SMA50 :
double SMA100 :
double MACD :

//+-----+
//| expert initialization function                                     |
//+-----+
int init()
{
//-----
//-----
return(0);
}
//+-----+
//| expert deinitialization function                                 |
//+-----+
int deinit()
{
//-----
//-----
return(0);
}
//+-----+
//| expert start function                                           |
//+-----+
int start()
{
//-----
SMA50 = iMA(Symbol(), 0, 50, 0, 0, 0, 0);
SMA100 = iMA(Symbol(), 0, 100, 0, 0, 0, 0);
MACD = iMACD(Symbol(), 0, 12, 26, 9, 0, 0, 0);
//-----
return(0);
}

```

PCM BROKERS

پست 9 - تعریف شرایط خرید و فروش

قوائد ورود به معامله خرید :

- (1) در موقعی که قیمت کمتر از هر دو میانگین متحرک 50 و 100 دوره ای است بازار را زیر نر بگیرید .
- (2) هرگاه قیمت یکی از میانگین متحرک های را به سمت بالا قطع کرده و از آن 10 پیپ فاصله گرفت به اندیکاتور MACD نگاه کنید . اگر این اندیکاتور طی میله اخیر از منفی به مثبت تغییر جهت داده است وارد معامله خرید شوید.

- (3) حد زیان را در نقطه حداقل 5 میله قبلی قیمت قرار دهید .

- (4)نیمی از معامله را با سود دو برابر مخاطره ای که پذیرفته اید ببندید و در نیم دیگر حد توقف را به نقطه ورود منتقل فرمایید.

- (5)نیمه دوم معامله را وقتی ببندید که قیمت SMA50 را به سمت پایین شکسته و 10 پیپ با آن فاصله گرفته باشد .

قواعد ورود به معامله فروش :

- (1) در موقعی که قیمت بیش از هر دو میانگین متحرک 50 و 100 دوره های بازار را زیر نظر بگیرید.

(2) هرگاه قیمت یکی از میانگین متحرک های راست به سمت بالا قطع کرد و از آن 10 پیپ فاصله گرفت به اندیکاتور MACD نگاه کنید اگر این اندیکاتور طی 5 میله اخیر از مثبت به منفی تغییر جهت داده اسن وارد معامله فروش شوید.

(3) حد زیان را در نقطه حداکثر 5 میله قبلی قیمت قرار دهید.

(4) نیمی از معامله را با سود دو برابر مخاطره ای پذیرفته اید ببندید و در نیم دیگر حد توقف را به نقطه ورود منتقل فرمایید.

(5) نیمه دوم معامله را وقتی ببندیم که قیمت SMA50 را به سمت بالا شکسته و 10 پیپ با آ» فاصله گرفته باشد .

توجه : اگر قیمت بین دو میانگین متحرک 50 و 10 دوره های است وارد معامله نشوید.

این همون متنی هست که در چند پست قبل دیده بودید.

شرایط به رنگ آبی توپر و استاپ ها به رنگ قرمز و شرایط تیپی به رنگ سبز نشون داده شده

حالا ما میخواهیم شرایط خرید و فروش رو به برنامه بفهمونیم.

دو متغیر به نامهای BuyMode و SellMode رو تعریف میکنیم ولی ابتدا باید بدونیم که از چه نوع متغیری باید استفاده کنیم.

اگر درست حدس زده باشید در لحظه ای که اکسپرت وجود حالت خرید رو بررسی میکنه تنها 2 حالت ممکنه اتفاق بیفته!

یا شرایط خرید برقرار هست یا نه!!!

پس اگر شما هم به این شکل فکر کرده باشید آفرین بر شما..

تا اینجا فهمیدیم که نوعی که باید برای متغیرهای خرید و فروش عنوان کنیم از نوع bool هستند.

پس به قبل از تابع init رفته و دو متغیر رو بدون دادن مقدار تعریف میکنیم (دیگه فکر نمیکنم نیازی باشه که توضیح بدم که چرا بدون دادن مقدار!!)


```

//+-----+
//|                                     MDA7- Moving & MACD.mq4 |
//|                                     MDA7 |
//|                                     FX.MDA7@YAHOO.CO.UK |
//+-----+
#property copyright "MDA7"
#property link      "FX.MDA7@YAHOO.CO.UK"

double SMA50 ;
double SMA100 ;
double MACD ;

bool BuyMode ;
bool SellMode ;

//+-----+
//| expert initialization function |
//+-----+
int init()
{
//-----
//-----
return(0);
}

//+-----+
//| expert deinitialization function |
//+-----+
int deinit()
{
//-----
//-----
return(0);
}

//+-----+
//| expert start function |
PCM BROKERS

```

حالا میریم سراغ تابع start تا این دو رو مقدار دهی کنیم.

شرایط خرید :

هرگاه قیمت یکی از میانگین متحرک های را به سمت بالا قطع کرده و از آن 10 پیپ فاصله گرفت به اندیکاتور MACD نگاه کنید . اگر این اندیکاتور طی میله اخیر از منفی به مثبت تغییر جهت داده است وارد معامله خرید شوید.

برای "و" از علامت && و برای "یا" از || استفاده خواهیم کرد.

حالا اینجا گفته که هریک از موینگها یعنی فرقی نمیکند که 50 باشد "یا" 100

و اما برای قیمت چی بنویسیم؟ آیا به نظر شما قیمت Bid که روی چارت نشون داده میشه بهترین انتخاب نیست؟

پس تا اینجا رو بنویسیم تا بریم سراغ بعدی

(SMA100 + 10 * Point < Bid) || (SMA50 + 10 * Point < Bid)

10 * Point یعنی همون 10 پیپ خودمون

نوبت مکدی میشه و شرط مکدی این هست که مثبت باشه (یعنی بالای صفر رو نشون بده)
پس شرط قبلی رو مینویسیم و کلهش رو داخل پرانتز میگذاریم .

((SMA100 + 10 * Point < Bid) || (SMA50 + 10 * Point < Bid)) && (0 < MACD))

لطفاً به تعداد و نحوه گذاشتن پرانتز ها دقت کنید , با تمرین و تکرار خواهید توانست که درست بنویسیدشون با
یک علامت کم یا زیاد یا برنامتون کار نخواهد کرد و یا کاری به غیر از چیزی که میخواهید انجام خواهد گرفت
یک شرط هم در انتهای متن داشتیم که اگر قیمت مابین موینگها بود وارد پوزیشن نشه
در واقع میتونیم اینطوری فکر کنیم که قیمت بایستی از هر دوتا موینگ هم بالاتر باشه پس کدمون به این شکل
خواهد شد :

((SMA100 +10 * Point < Bid) && (SMA50 + 10 * Point) && (0 < MACD))

اگر دقت کنید من علامت || رو حذف کردم و به جاش && گذاشتم علتش هم اینه که وقتی قیمت مابین دوتا
موینگ باشه نباید وارد معامله بشیم پس وقتی قیمت به اندازه 10 پیپ از موینگ بالایی فاصله گرفته باشه حتماً
فاصلش از موینگ پایینی بیش از 10 پیپ خواهد بود
در مورد نوشتن شرایط خرید دقیقاً عکس این حالتها خواهد بود و برای اینکه پست طولانی نشه اون رو میذارم به
عده خودتون

```

//+-----+
//|                                     MDA7- Moving & MACD.mq4 |
//|                                     MDA7 |
//|                                     FX.MDA7@YAHOO.CO.UK |
//+-----+
#property copyright "MDA7"
#property link      "FX.MDA7@YAHOO.CO.UK"

double SMA50 ;
double SMA100 ;
double MACD ;

bool BuyMode ;
bool SellMode ;

//+-----+
//| expert initialization function |
//+-----+
int init()
{
//----
//----
return(0);
}
//+-----+
//| expert deinitialization function |
//+-----+
int deinit()
{
//----
//----
return(0);
}
//+-----+
//| expert start function |
//+-----+
int start()
{
//----
SMA50 = iMA(Symbol(), 0, 50, 0, 0, 0, 0);
SMA100 = iMA(Symbol(), 0, 100, 0, 0, 0, 0);

MACD = iMACD(Symbol(), 0, 12, 26, 9, 0, 0, 0);

// BUY AND SELL MODES-----+
BuyMode = ((MACD > 0) && (Bid > SMA50 + 10 * Point) && (Bid > SMA100 + 10 * Point));
SellMode = ((MACD < 0) && (Bid < SMA50 - 10 * Point) && (Bid < SMA100 - 10 * Point));
//----
return(0);
}
//+-----+

```

PCM BROKERS

پست 10 - مشخص کردن استاپ برای خرید و فروش

قواند ورود به معامله خرید :

حد زیان را در نقطه حداقل 5 میله قبلی قیمت قرار دهید .

حالا ما باید در بین 5 کندل قبلی به دنبال کندلی بگردیم که کمترین low رو داره برای این کار تابعی بصورت آماده

در متا هست به این ترتیب:

int iLowest(string symbol, int timeframe, int type, int count=WHOLE_ARRAY, int start=0)

واما پارامترهای تابع:

symbol و timeframe که هیچ

type یعنی اینکه دنبال کمترین مقدار توی چه چیزی میگردیم (کمترین های , لو , این) که در اینجا ما دنبال

کمترین Mode_Low میگردیم

count یعنی دنبال کترین مقدار بین چه بازه بگرد

start یعنی از کجا شروع کن و دنبال کمترین مقدار بین بازه بگرد.

پس میشه این:

int iLowest(Symbol(), 0 , Mode_Low , 5 , 0)

خوب پس باید بریم اول قبل از تابع init تعریفشون کنیم بعد توی استارت مقدار دهی کنیم

حالا اگر به نوع تابع دقت کنید int هست (عدد صحیح) پس این تابع شماره کندلی رو که کمترین 0اص رو در 5

کندل گذشته برمیگردونه

حالا نیاز به مقدار لوی این کندل داریم و اون هم با تابع زیر بدست میاد

double iLow(string symbol, int timeframe, int shift)

Shift که همون شماره کندل هست رو چند خط پیش حساب کردیم و بجای shift همون مقدار رو قرار میدیم.

برای استاپ سل هم باید برعکس این دنبال "بیشترین" " های " در 5 کندل گذشته باشیم

```

//|
//|
//|-----+-----+-----+
//| MDA7 |
//| FX.MDA7@YAHOO.CO.UK |
//|-----+-----+-----+
#property copyright "MDA7"
#property link "FX.MDA7@YAHOO.CO.UK"

double SMA50 ;
double SMA100 ;
double MACD ;

bool BuyMode ;
bool SellMode ;

double slB ;
double slS ;

//+-----+-----+-----+
//| expert initialization function |
//+-----+-----+-----+
int init()
{
//----
//----
return(0);
}
//+-----+-----+-----+
//| expert deinitialization function |
//+-----+-----+-----+
int deinit()
{
//----
//----
return(0);
}
//+-----+-----+-----+
//| expert start function |
//+-----+-----+-----+
int start()
{
//----
SMA50 = iMA(Symbol(), 0 , 50 , 0 , 0 , 0 , 0);
SMA100 = iMA(Symbol(), 0 , 100 , 0 , 0 , 0 , 0);

MACD = iMACD(Symbol(),0, 12 , 26 , 9 , 0 , 0,0);

// BUY AND SELL MODES-----+
BuyMode = ((MACD > 0) && (Bid > SMA50 +10 * Point) && (Bid > SMA100 + 10 * Point));
SellMode = ((MACD < 0) && (Bid < SMA50 -10 * Point) && (Bid < SMA100 - 10 * Point));

// StopLoss & TakeProfit-----+
slB = iLow (Symbol(), 0 , iLowest (Symbol(),0,MODE_LOW , 5 , 0) );
PCM BROKERS ymbol(), 0 , iHighest(Symbol(),0,MODE_HIGH, 5 , 0) );
//----

```

پست 11 - دستورات خرید و فروش

تا اینجا متغیر ها رو تعریف کردیم و شرایط رو برای خرید و فروش مشخص کردیم بعدش رفتیم سراغ استاپلاس تقریباً الان میتونیم بریم سر نوشتن دستورات خرید و فروش و به برنامه میگیم اگر شرایط بای برقرار بود یک بای بگیر و اگر شرایط سل برقرار شد یک پوزیشن سل باز کن

if(BuyMode)

```

{
}

```

وقتی BuyMode == رو برابر چیزی قرار ندادیم یعنی اینکه بای مد ترو باشه و میشد بنویسیم BuyMode == True که علامت "==" به مفهوم "برابری" هست که مسلماً با "=" فرق میکنه.

تابع دستور برای اردرگیری به این شکل هست :

int OrderSend(string symbol, int cmd, double volume, double price, int slippage, double stoploss, double takeprofit, string comment=NULL, int magic=0,datetime expiration=0, colorarrow_color=CLR_NONE)

Symbol : قبلا توضیح داده شده

cmd : نوع پوزیشن (بای - سل - بای لیمیت - سل لیمیت - بای استاپ - سل استاپ)

Volume : لات سایز پوزیشن

price : قیمت باز شدن اردر

slippage : وقتی که دستور رو صادر کرد اگر قیمت به اندازه slippage از لحظه صدور دستور حرکت داشته باشه

باز هم اردر خنأهء گرفت اگر بیشتر شد دیگه اردر رو بیخیال میشه

stoploss : مقدار عدد استاپلاس رو وارد میکنیم

takeprofit : تی پی

comment : میتونیم بوسیله این قسمت یک توضیح خطی بگذاریم مثلا "MACD & moving" و جلوی هر اردری

که اکسپرت خواهد گرفت این عبارت نشون داده میشه و میتونید متوجه بشید که این اردر توسط خودتون یا

کدوم اکسپرت گرفته شده

magic : برای اردرها میتونید یک مجیک نامبر تعریف کنید و در صورت داشتن اردرهای متفاوت اونها رو با مجیک

نامبر به اکسپرت بفهمونید که منظورتون کدوم نوع هست

expiration : در اردر های لیمیت و استاپ اگر به میزان این مقدار زمان اردر باز نشه به صورت اتوماتیک حذف

میشه دقت کنید که نوعش datetime هست

arrow_color : تعریف میکنید که وقتی اردر گرفته شد فلش اون روی چارت به چه رنگی باشه.

int OrderSend(Symbol(),OP_BUY, LotSize, Ask , 2 , slB , 0 , "Moving & MACD" , 0123456789 , 0 , Blue)

به این ترتیب دستور معامله خرید در جفت ارزی که روش هستیم / با لایح LotSize / در قیمت Ask / با اسلیپیج

2 / و استاپی که قبلا براش تعریف کرده بودیم / تی پی رو 0 گذاشتیم تا چیزی اعمال نشه چون بعدا با وجود

شرایطی میخوئیم نصف اردر رو ببندیم /مجیک نامبر 0123456789 در نظر گرفتیم که این عدد بصورت دلخواه هر

عددی میتونه باشه / کامنتی که در نظر گرفتیم رو نوشتیم . اکسپایر رو 0 گذاشتیم چون پوزیشن ما اینستنت

هست (از نوع لیمیت یا استاپ نیست) که بخوئیم بعدا حذفش کنیم / و در انتها رنگ آبی رو برای پوزیشن باش

انتخاب کردیم.

حالا میخوئیم متغیر LotSize رو قبل از init تعریف کنیم و کاری کنیم به در ورودی و در قسمت اینپوت اسکپرت

داشته باشیو تا بعدا در صورت نیاز بتونیم تغییر بدیم .

برای این کار کافیست که قبل از نوع متغیرتون بنویسید extern . به همین راحتی

پس میشه

extern double LotSize = 0.2;

با اینکار ما متغیر LotSize رو با مقدار اولیه 0.2 تعریف کردیم و با گذاشتن extern اون رو به قسمت ورودی اکسپرت بردیم تا بعدا بتونیم عوضش کنیم.

کدی که تا حالا نوشتیم به این شکل خواهد بود:

شکل 1- از ابتدا تا تابع start

```
//+-----+
//|                                     MDA7- Moving & MACD.mq4 |
//|                                     MDA7 |
//|                                     FX.MDA7@YAHOO.CO.UK |
//+-----+
#property copyright "MDA7"
#property link      "FX.MDA7@YAHOO.CO.UK"

double SMA50 ;
double SMA100 ;
double MACD ;

bool BuyMode ;
bool SellMode ;

double s1B ;
double s1S ;

extern double LotSize = 0.2 ;
int ticket;

//+-----+
//| expert initialization function |
//+-----+
int init()
{
//-----
return(0);
}
//+-----+
//| expert deinitialization function |
//+-----+
int deinit()
{
//-----
return(0);
}
//+-----+
//| expert start function |
//+-----+
int start()
{
PCM BROKERS
```

شکل 2- مابقی کد (تابع Start)

```

//+-----+
int start()
{
//-----
SMA50 = iMA(Symbol(), 0 , 50 , 0 , 0 , 0 , 0);
SMA100 = iMA(Symbol(), 0 , 100 , 0 , 0 , 0 , 0);

MACD = iMACD(Symbol(),0, 12 , 26 , 9 , 0 , 0 ,0);

// BUY AND SELL MODES-----+

BuyMode = ((MACD > 0) && (Bid > SMA50 +10 * Point) && (Bid > SMA100 + 10 * Point));
SellMode = ((MACD < 0) && (Bid < SMA50 -10 * Point) && (Bid < SMA100 - 10 * Point));

// StopLoss & TakeProfit-----+

slB = iLow (Symbol(), 0 , iLowest (Symbol(),0,MODE_LOW , 5 , 0 ) );
slS = iHigh(Symbol(), 0 , iHighest(Symbol(),0,MODE_HIGH, 5 , 0 ) );

// Buy AND Sell Orders -----+

if(BuyMode)
{
ticket = OrderSend(Symbol(),OP_BUY,LotSize,Ask, 2 ,slB, 0 ,"Moving & MACD", 123456789 , 0 , Blue);
}

if(SellMode)
{
ticket = OrderSend(Symbol(),OP_SELL,LotSize,Bid, 2 ,slS, 0 ,"Moving & MACD", 123456789 , 0 , Red);
}

//-----
return(0);
}
//+-----+

```

PCM BROKERS

خوب حالا کاری که میخوام انجام بدید این هست که از ابتدا تا این پست رو (یعنی 11 تا پستی رو که در مورد آموزش بوده) یکبار مطالعه کنید و بدون اینکه تاپیک و کدها رو نگاه کنید از اول و چشم بسته کد رو از اول بنوسید.

بعد ران کنید ببینید درست کار میکنه یا اگر و اگر مشکلی داره سعی کنید بفهمید که چرا؟؟!!!! و به من هم بگید تا باهم حلش کنیم.

لطفا دوستانی که وارد هستند کمی توی جواب دادن صبر کنند تا اونایی که مثل من تازه کار هستند بتونند خودشون فکر کنند و مساله رو درک کنند.

برای اون دسته از دوستانی که کاملا تازه کار هستند :

بعد از نوشتن کد از منوی بالای صفحه روی Compile کلیک کنید تا برنامتون به زبان ماشین تبدیل شده و در قسمت اکسپرت متاتریدرتون نشون داده بشه

اگر بعد از کامپایل اروری دریافت کردید بدونید که کدتون یه جاییش ایراد داره و باید بگردید و ایرادتونرو پیدا کنید

حتماً حتماً حتماً توایستراتژی تستر یا روی حساب دمو کدهاتون رو امتحان کنید چون حالا حالاها خیلی کار داریم تا بتونیم کدی بنویسیم که دقیقاً چیزی که ما می‌خواهیم رو انجام بده 🤖

بینم چه میکنیییییییدددد

همه چیز اولش یکم فکر میبره

همونطور که قبلا گفتیم وقتی یک اندیکاتور یا اکسپرت رو ران میکنیم کد ما از ابتدا شروع شده و اول متغیر ها رو میشناسه بعد یک بار تابع init اجرا میشه بعد با هر تیک جدید تابع start یک بار از ابتدا تا انتها اجرا میشه و این پروسه اجرا شدن start تا زمانی که اندیکاتور یا اکسپرت رو حذف کنیم ادامه پیدا میکنه

ما شرایط خرید و فروش رو تعریف کردیم بعد هم دستور خریو و فروش رو با اون شرایط نوشتیم و اتفاقی که افتاد این بود که وقتی یک بار تابع start اجرا میشد و شرط بای یا سل برقرار بود یک اردر میگرفت .

در تیک جدیدی که به بازار میومد دوباره شرط برقرار بود و باز در تیک جدید پوزیشن گرفته میشد.

حال برای مشکل چه کنیم؟!!

همونطور که دوست خوبمون آقای hamkar81 در پستشون گفتند در زبان mql4 تابعی داریم که تعداد پوزیشن‌ها رو به ما برگردونه :

این تابع تعداد یوزیشنهای چارت رو برمیگردونه پس میتونیم به شرط بگذاریم به این نحو :

```

//| expert start function |
//+-----+
int start()
{
//-----
SMA50 = iMA(Symbol(), 0 , 50 , 0 , 0 , 0 , 0);
SMA100 = iMA(Symbol(), 0 , 100 , 0 , 0 , 0 , 0);

MACD = iMACD(Symbol(),0, 12 , 26 , 9 , 0 , 0 ,0);

// BUY AND SELL MODES-----+

BuyMode = ((MACD > 0) && (Bid > SMA50 +10 * Point) && (Bid > SMA100 + 10 * Point));
SellMode = ((MACD < 0) && (Bid < SMA50 -10 * Point) && (Bid < SMA100 - 10 * Point));

// StopLoss & TakeProfit-----+

sLB = iLow (Symbol(), 0 , iLowest (Symbol(),0,MODE_LOW , 5 , 0 ) );
sLS = iHigh(Symbol(), 0 , iHighest(Symbol(),0,MODE_HIGH, 5 , 0 ) );

// Buy AND Sell Orders -----+
if(OrdersTotal() < 1)
{
    if(BuyMode)
    {
        ticket = OrderSend(Symbol(),OP_BUY,LotSize,Ask, 2 ,sLB, 0 ,"Moving & MACD", 123456789 , 0 , Blue);
    }

    if(SellMode)
    {
        ticket = OrderSend(Symbol(),OP_SELL,LotSize,Bid, 2 ,sLS, 0 ,"Moving & MACD", 123456789 , 0 , Red);
    }
}
//-----
return(0);
}
PCM BROKERS-----+

```

پست 13 - آیا مشکل واقعا حل شد؟؟؟

ازتون میخوام که کد رو بنویسید و ببینید که درست کار میکنه یا نه!

اگر کد رو توی تستر امتحان کنید دقیقا درست عمل خواهد کرد.

ازتون میخوام که یه حساب دمو باز کنید و فقط یه پندینگ در یک جای پرت و یک جفت ارزی که این کد رو روی

اون ران نخواهید کرد باز کنید تا فعال نشه بعد رو اکانت دمو و روی یک جفت ارز مجزا کد رو ران کنید.

و بگید که چه اتفاقی افتاد و چرا؟

پست 14 - حل کامل مشکل تعداد اردرها..

هونطور که دیدیم Orserstotal همه اردهای باز رو میشمره و براش فرقی نمیکنه که این اردر روی چه جفت ارزی

باشه یا فرقی نمیکنه که لیمیت باشه یا پوزیشن باز

فرض کنید شما در حال ترید کردن بر روی طلا هستید و این اکسپرت رو هم روی یورو/دلار ران میکنید ، اگر روی

طلا پوزیشن بازی داشتید در واقع Orderstotal بیشتر از 0 خواهد بود

پس اکسپرت نخواهد تونست که کاری انجام بده

برای حل این مساله ما باید بین Orderstotal به دنبال پوزیشنهای بگردیم که هم ارز با ارزی باشه که اکسپرت رو ران کردیم ، در ضمن به پوزیشنهای پندینگ هم کاری نخواهیم داشت. یعنی بین تمام اردر ها از 0 تا کلشون باید بگردیم و پوزیشنهایی رو که لازم داریم رو سوا کنیم بعد بشماریمشون...

برای اینکار از حلقه for استفاده میکنیم

for(i=0 ;i<OrdersTotal(); i++)

همونطور که ملاحظه میکنید for سه قسمت داره که با سمیکالن " ; " از هم جدا شدند. از سمت چپ اولین مقدار گویای این هست که از چه عددی باید شروع کنیم دومین مقدار نشان دهنده این هست که تا چه مقداری باید پیش ببریم مقدار سوم هم بازگو کننده این هست که دفعه دوم شمارنده به کجا منتقل بشه یا به زبان ساده چند تا چند تا باید جلو ببریم.

$i++$ یعنی اینکه هر بار i رو یک واحد افزایش میدیم و میشد این رو به این صورت هم بنویسیم $i = i + 1$ در ابتدا $i = 0$ هست ، بار دوم که for اجرا میشه $i = 0 + 1$ بار سوم $i = 1 + 1$ و تا اینکه مقدارمون برسه به 1 دونه کمتر از orderstotal.

پس ما نیاز داریم که قبل از اینکه دستور اردر بدیم ، تعداد پوزیشنهای باز این جفت ارز رو داشته باشیم پس قبل از دستورات ordersend این کار رو انجام میدیم فعلا شکل رو داشته باشد تا بقیه پارامتر ها رو توضیح بدم .

```
//Orders Counting -----+
    int orderscnt = 0;
    for(int i=0; i < OrdersTotal(); i++)
    {
        OrderSelect(i,SELECT_BY_POS,MODE_TRADES);
        if((OrderSymbol() == Symbol()) && ((OrderType() == OP_SELL) || (OrderType() == OP_BUY)))
        {
            orderscnt++;
        }
    }

// Buy AND Sell Orders -----+
    if((BuyMode) && (orderscnt < 1))
    {
        ticket = OrderSend(Symbol(),OP_BUY,LotSize,ask, 2 ,slB, 0 ,"Moving & MACD", 123456789 , 0 , Blue);
    }

    if((SellMode) && (orderscnt < 1))
    {
        ticket = OrderSend(Symbol(),OP_SELL,LotSize,Bid, 2 ,slS, 0 ,"Moving & MACD", 123456789 , 0 , Red);
    }

//-----
    return(0);
}
//+-----+
```

PCM BROKERS

همون طور که میبینید قبل از ordersend ها باید پوزیشن‌ها رو می‌شمریم که این کار انجام شد و در قسمت اردر گیری شرط قبلی رو برداشتم و بجاش شرط جدید اضافه کردم.

بعد از اینکه حلقه for رو نوشتیم باید که بوسیله OrderSelect پوزیشن رو بردارید (سلکت کنید) تا بتونیم سیمبل و نوع بای یا سل بودنش رو تشخیص بدیم اگر این کار رو انجام ندیم برنامه با مشکل مواجه میشه و نمیتونه تشخیص بده که منظور از نوع بای یا سل بودن کدوم پوزیشن رو باید چک کنه!

یه متغیر از نوع عدد صحیح بنام orderscnt هم تعریف کردیم که در واقع تعداد پوزیشن‌ها رو در اون ذخیره خواهیم کرد

قبل از شمارش به اون عدد 0 دادیم تا در هر تیک قبل از شمردن تعداد رو صفر بکنه و از نو بشمره تا عدد قبلی در تیک قبل با تعداد جدید جمع نشن.

++ هم رو بالا توضیح دادم خدمتون

هر گاه شرط برقرار بشه یک واحد به orderscnt اضافه خواهد شد قبل از for. مقدارش 0 هست و اگر پوزیشنی پیدا کنه که ما لازم داریم یک واحد بهش اضافه میشه

واین حلقه تا زمانی که همه پوزینهای باز رو چک کنه ادامه پیدا میکنه و وقتی کار حلقه تموم شد ارزش خارج میشه و میره سراغ خطوط بعدی که همون ordersend هست.

حالا ازتون میخوام که حلقه رو برعکس بنویسید یعنی i از orderstotal شروع بشه . به 0 ختم بشه

پست 15 - بستن نصف

نیمی از معامله را با سود دو برابر مخاطره ای که پذیرفته اید ببندید و در نیم دیگر حد توقف را به نقطه ورود منتقل فرمایید.

این همون متنی هست که قبلا قرار داده شده

با توجه به متن وقتی که پوزیشن ما به میزان دو برابر فاصله استاپ لاس توی سود بود ، نصف سائزش باید بسته بشه

ما قبلا نقطه استاپ رو به دست آورده بودیم کافی هست که تفاضل نقطه ورود و استاپ رو محاسبه کنیم و بعد شرطی قرار بدیم که وقتی قیمت با نقطه ورود به میزان 2 برابر اون فاصله گرفت نصف پوزیشن بسته بشه

برای این کار لازم هست که ابتدا اردرها رو مثل قبل بگردیم و سلکت کنیم و تک تک بررسیشون کنیم

و اگر شرط برقرار شد دستور close بهش بدیم

اگر یادتون باشه قبلا برای شمردن پوزیشنها این کار رو کرده بودیم

و برای اینکه دوباره کاری نشه میتونیم داخل همون کد ، کدهای جدیدمون رو بگنجونیم..

```
//Orders Counting & Close Orders-----+
int orderscnt = 0;
for(int i=0; i < OrdersTotal(); i++)
{
    OrderSelect(i,SELECT_BY_POS,MODE_TRADES);
    if((OrderSymbol() == Symbol()) && ((OrderType() == OP_SELL) || (OrderType() == OP_BUY)))
    {
        orderscnt++;
    }

    if((OrderSymbol() == Symbol()) && (OrderType() == OP_SELL))
    {
        if((OrderOpenPrice() - Ask) >= (s1S - OrderOpenPrice()) * 2)
        {
            OrderClose(OrderTicket(), OrderLots() / 2, Ask, slippage, Black);
        }
    }

    if((OrderSymbol() == Symbol()) && (OrderType() == OP_BUY))
    {
        if(Bid - OrderOpenPrice() >= (OrderOpenPrice() - s1B) * 2)
        {
            OrderClose(OrderTicket(), OrderLots() / 2, Bid, slippage, Black);
        }
    }
}

// Buy AND Sell Orders -----+
if((BuyMode) && (orderscnt < 1))
{
    ticket = OrderSend(Symbol(),OP_BUY,LotSize,Ask, 2 ,s1B, 0 ,"Moving & MACD", 123456789 , 0 , Blue);
}

PCM BROKERS
if((SellMode) && (orderscnt < 1))
```

شمارش پوزیشنها

بستن نیمی از پوزیشن فروش در دو برابر سود

بستن نیمی از پوزیشن خرید در دو برابر سود

پست 16 - انتقال استاپ به نقطه ورود

نیمی از معامله را با سود دو برابر مخاطره ای که پذیرفته اید ببندید و در نیم دیگر حد توقف را به نقطه ورود منتقل فرمایید.

بعد از اینکه نیمی از پوزیشن بسته شد نیاز داریم تا با دستوری استاپلاس رو به نقطه 0 منتقل کنیم (اردر را باید مدیفای کنیم)

برای این منظور از دستور OrderModify استفاده میکنیم

و تنها کاری که باید انجام دهیم این هست که دقیقا بعد از اینکه دستور OrderClose رو نوشتیم در سطر پائین

دستور Modify رو بنویسیم

`bool OrderModify(int ticket, double price, double stoploss, double takeprofit, datetime expiration, color arrow_color=CLR_NONE)`
به این شکل :

```
//Orders Counting & Close Orders-----+
int orderscnt = 0;
for(int i=0; i < OrdersTotal(); i++)
{
    OrderSelect(i,SELECT_BY_POS,MODE_TRADES);
    if((OrderSymbol() == Symbol()) && ((OrderType() == OP_SELL) || (OrderType() == OP_BUY)))
    {
        orderscnt++;

        if((OrderSymbol() == Symbol()) && (OrderType() == OP_SELL))
        {
            if(OrderOpenPrice() - Ask >= (s1S - OrderOpenPrice()) * 2)
            {
                OrderClose (OrderTicket(), OrderLots() / 2, Ask, 2, Black);
                OrderModify(OrderTicket(), Bid, OrderOpenPrice(), 0, 0, Violet);
            }
        }

        if((OrderSymbol() == Symbol()) && (OrderType() == OP_BUY))
        {
            if(Bid - OrderOpenPrice() >= (OrderOpenPrice() - s1B) * 2)
            {
                OrderClose(OrderTicket(), OrderLots() / 2, Bid, 2, Black);
                OrderModify(OrderTicket(), Ask, OrderOpenPrice(), 0, 0, Lime);
            }
        }
    }
}

// Buy AND Sell Orders -----+
if((BuyMode) && (orderscnt < 1))
{
    ticket = OrderSend(Symbol(),OP_BUY,LotSize,Ask, 2,s1B, 0,"Moving & MACD", 123456789 , 0 , Blue);
}
```

پست 17 - ادامه بستن نصف پوزیشن و انتقال استاپلاس

همونطور که قبلا گفتیم باید بتونیم که شبیه به ماشین فکر کنیم و با دونستن این قضیه که با هر تیک 1 بار تابع

start اجرا خواهد شد ، وقتی شرطی رو مینویسیم بایستی حساب کتاب تیکهای بعدی رو هم بکنید

بستن نصف پوزیشن و انتقال استاپ در تیک اول درست عمل میکنه ولی بعدش چه اتفاقی میفته؟!

slB - OrderOpenPrice() در تیکهای بعدی همیشه 0 خواهد شد چون از این به بعد استاپ مساوی نقطه ورود

هست و $2*0=0$ هست

پس شرط ما که فاصله قیمت با نقطه ورود بیش از 2 برابر فاصله استاپ تا نقطه ورود همیشه برقرار خواهد بود و

در تیکهای بعدی دائما حجم پوزیشن رو نصف خواهد کرد و این عمل رو تا جایی که مینیمم لاتج بروکر باشه

ادامه خواهد داد....

پس الان ما نیاز داریم که فکر کنیم و یک مشخصه برای پوزیشنی که قرار هست نصفش بسته بشه رو پیداکنیم

که این مشخصه در تیکهای بعدی مانع از نصف کردن اردر بشه

مثلا میتونیم بگیریم که اگر اردری حجمش برابر LotSize باشه این عمل رو انجام بده یا اینکه مثلا اگر نقطه ورود

مخالف استاپلاس باشه و یا ...

برنامه نویسی مثل اثر انگشت میمونه و هر کسی با سلیقه خودش میتونه به برنامه و ایده ای که داره شکل

بده فقط کافی هست که یکم دستتون راه بیفته بقیش در عرض نهایتا چند ساعت فکر کردن و پیدا کردن تکه

های پازل حل خواهد شد.

پس با توجه به اینکه لات سایز ما در قسمت ورودی قابل تنظیم هست و همیشه **ثابت هست** من از این گزینه

استفاده میکنم تا برنامه وقتی یک بار این کار رو انجام داد و لات سایز رو نصف کرد در تیکهای بعدی وقتی لات

سایز پوزیشن رو چک میکنه دیگه شرط برقرار نباشه و اردرها رو تکه تکه نکنه.

شما میتونید از مثال دوم یا هر کاری که مانع از این کار بشه استفاده کنید.

```
//Orders Counting & Close Orders-----+
int orderscnt = 0;
for(int i=0; i < OrdersTotal(); i++)
{
    OrderSelect(i,SELECT_BY_POS,MODE_TRADES);
    if((OrderSymbol() == Symbol()) && ((OrderType() == OP_SELL) || (OrderType() == OP_BUY)))
    {
        orderscnt++;
    }

    if((OrderSymbol() == Symbol()) && (OrderType() == OP_SELL) && (OrderLots() == LotSize))
    {
        if(OrderOpenPrice() - Ask >= (sls - OrderOpenPrice()) * 2)
        {
            OrderClose (OrderTicket(), OrderLots() / 2, Ask, 2, Black);
            OrderModify(OrderTicket(), Bid, OrderOpenPrice(), 0, 0, Violet);
        }
    }

    if((OrderSymbol() == Symbol()) && (OrderType() == OP_BUY) && (OrderLots() == LotSize))
    {
        if(Bid - OrderOpenPrice() >= (OrderOpenPrice() - slb) * 2)
        {
            OrderClose(OrderTicket(), OrderLots() / 2, Bid, 2, Black);
            OrderModify(OrderTicket(), Ask, OrderOpenPrice(), 0, 0, Lime);
        }
    }
}

// Buy AND Sell Orders -----+
PCM BROKERS) && (orderscnt < 1))
{
```

شرط چک کردن حجم پوزیشن
اضافه شد

جهت یادآوری :

علامت برابری "="

علامت مساوی "="

علامت نابرابری (مخالف بودن) "!="

علامت بزرگتر ">"

علامت بزرگتر مساوی ">="

علامت کوچکتر "<"

علامت کوچکتر مساوی "<="

پست 18 - ادامه بستن نصف پوزیشن و انتقال استاپلاس (قسمت دوم)

هر بار که کدی رو مینویسید بهتر هست که ابتدا در استراتژی تستر اون رو ران کنید و در همون پنجره استراتژی

تستر به قسمت ژورنال برید و اگر اروری دیدید باید متوجه بشید که یک جای کارتون ایراد داره

اگر یادتون باشه در پستهای قبلی یک فایل پی دی اف براتون آپلود کرده بودم در صفحات 100 و 101 میتونید

لیست ارورها روبینید

حال اگر کد رو تا همینجا بنویسید و روی تستر ران کنید متوجه خواهید شد که پوزیشن رو نصف میکنه ولی

استاپ جابجا نمیشه ، با رفتن به قسمت ژورنال ببینیم که ارور 4108 بهمون میده و اگه در پی دی اف نگاه کنید

این ارور مربوط به شماره تیکت هست (یعنی تیکتی با این عدد نمیتونه پیدا کنه) حالا فکر کنیم ببینیم که چه اتفاقی داره میفته؟

وقتی که اردر i ام رو سلکت میکنیم در واقع تمام پارامترهایی که داخل توابع orderclose و ordermodify مربوط به همون پوزیشن i ام باید باشه.

وقتی شما مثلا نصف اردر خرید رو میندید در واقع یک بای جدید انجام میدید با نصف لاتیتج و حتما باید شماره تیکت عوض شده باشه چون نمیشه که شما 2 تا پوزیشن با یک شماره تیکت داشته باشید.

برای حل این مشکل میتونیم ordermodify رو از داخل if خارج کنیم و براش یک if جداگانه بنویسیم به این شکل که : اگر "ardersiiml == سیمبل و اردرتایپ == بای و اردرلاتس == LotSize/2" بعدش ordermodify رو داخل این if قرار بدیم.

ولی یک کار دیگه بهتر از این هم میتونستیم انجام بدیم و اون اینکه با برقرار شدن شرط دو برابر بودن سود به استاپ ، ابتدا ordermodify میکردیم بعد دستور بستن نصفش رو قرار میدادیم

اینطوری تعداد سطرهای کدی که داریم مینویسیم خیلی کمتر خواهد بود. و هر چقدر از for و if کمتری استفاده کنید برنامه خیلی راحت تر کار خواهد کرد. پس کافیه که فقط جای دوتا سطرمون رو عرض کنیم .
به همین راحتی...

```
//Orders Counting & Close Orders-----+
int orderscnt = 0;
for(int i=0; i < OrdersTotal(); i++)
{
    OrderSelect(i,SELECT_BY_POS,MODE_TRADES);
    if((OrderSymbol() == Symbol()) && ((OrderType() == OP_SELL) || (OrderType() == OP_BUY)))
    {
        orderscnt++;
    }

    if((OrderSymbol() == Symbol()) && (OrderType() == OP_SELL) && (OrderLots() == LotSize))
    {
        if(OrderOpenPrice() - Ask >= (s1S - OrderOpenPrice()) * 2)
        {
            OrderModify(OrderTicket(), Bid, OrderOpenPrice(), 0, 0, Violet);
            OrderClose (OrderTicket(), OrderLots() / 2, Ask, 2, Black);
        }
    }

    if((OrderSymbol() == Symbol()) && (OrderType() == OP_BUY) && (OrderLots() == LotSize))
    {
        if(Bid - OrderOpenPrice() >= (OrderOpenPrice() - s1B) * 2)
        {
            OrderModify(OrderTicket(), Ask, OrderOpenPrice(), 0, 0, Lime);
            OrderClose (OrderTicket(), OrderLots() / 2, Bid, 2, Black);
        }
    }
}

// Buy AND Sell Orders -----+
if((RunMode) && (orderscnt < 1))
PCMBROKERS
ticket = OrderSend(Symbol(),OP_BUY,LotSize,Ask, 2 ,s1B, 0 ,"Moving & MACD". 123456789 , 0 , Blue);
```

پس همیشه به دو موضوع باید دقت کنیم :

1- تقدم و تاخر كدهایی كه مینویسیم (یعنی اگر يك سطر بالا و پایین بنویسیم با دونستن این مساله كه كد از

ابتدای start خط به خط و به ترتیب اجرا شده و به آخر start خواهد رسید ، ممكن هست كه دقیقاً مثل این

مشكل آخرمون برنامه درست كار نكنه در حالی كه همه چیز رو درست نوشتیم!)

2- با كمی فكر در مورد شرطها و حلقه ها میتونیم به جای نوشتن چندین for و if و حجیم شدن كد و كند شدن

سرعت تست گیری اونها رو یكجا بنویسیم ، مثلاً ما میتونستیم در همین قسمت از كد 3 تا for بنویسیم با 5 تا

if كه تعداد for رو به 1 و if رو به 3 رسوندیم .

بست 19 - بین نیمه دوم پوزیشن

در معامله خرید نیمه دوم معامله را وقتی ببندید كه قیمت SMA50 را به سمت پایین شكسته و 10 پیپ با آن

فاصله گرفته باشد . (معامله فروش برعكس)

برای این كار نیز باید ابتدا يك مشخصه برای پوزیشن دوم پیدا كنیم مثل لاتیج یا استاپلاس اردر.

و بعد شرط مربوط به اون رو در قسمت اردر سلكت ها بنویسیم.

```
for(int i=0; i < OrdersTotal(); i++)
{
    //Shomareshe Position haye Baz
    OrderSelect(i,SELECT_BY_POS,MODE_TRADES);
    if((OrderSymbol() == Symbol()) && ((OrderType() == OP_SELL) || (OrderType() == OP_BUY)))
    {
        orderscnt++;
    }

    //Bastane Nimeye 2vome Positione Buy
    if((OrderSymbol() == Symbol()) && (OrderType() == OP_BUY) && (OrderLots() == LotSize / 2))
    {
        if(Bid < SMA50 -10 * Point)
        {
            OrderClose (OrderTicket(), OrderLots(), Bid, 2, Black);
        }
    }

    //Bastane Nimeye 2vome Positione Sell
    if((OrderSymbol() == Symbol()) && (OrderType() == OP_SELL) && (OrderLots() == LotSize /2))
    {
        if(Bid > SMA50 +10 * Point)
        {
            OrderClose (OrderTicket(), OrderLots(), Ask, 2, Black);
        }
    }

    //Ja be ja kardane S/L va Bastane Nesfe Sell
    if((OrderSymbol() == Symbol()) && (OrderType() == OP_SELL) && (OrderLots() == LotSize))
    {
        if(OrderOpenPrice() - Ask >= (s1S - OrderOpenPrice()) * 2)
        {
            OrderModify(OrderTicket(), Bid, OrderOpenPrice(), 0, 0, Violet);
            OrderClose (OrderTicket(), OrderLots() / 2, Ask, 2, Black);
        }
    }

    //Ja be ja kardane S/L va Bastane Nesfe Buy
    if((OrderSymbol() == Symbol()) && (OrderType() == OP_BUY) && (OrderLots() == LotSize))
    {
        if(Bid - OrderOpenPrice() >= (OrderOpenPrice() - s1B) * 2)
        {
            OrderModify(OrderTicket(), Ask, OrderOpenPrice(), 0, 0, Lime);
            OrderClose (OrderTicket(), OrderLots() / 2, Bid, 2, Black);
        }
    }
}
```

شمردن تعداد پوزیشن

بستن نیمه دوم بای(پوزیشنی كه حتمش نصف شده بود)

بستن نیمه دوم سل (پوزیشنی كه حتمش نصف شده بود)

1- مدیفای كردن استاپ
2- بستن نصف پوزیشن اول
برای سل

1- مدیفای كردن استاپ
2- بستن نصف پوزیشن اول
برای بای

پست 20 - حل مشکل رعایت کردن فاصله 10 پیپ در بروکرهای 4 رقمی و 5 رقمی

همونطور که خاطرتون هست در کد برای فاصله 10 پیپ از point استفاده کردیم

کار point و Digits تقریباً شبیه به هم هست

point در بروکر 4 رقمی عبارت است از 0.0001 و در بروکر 5 رقمی 0.00001 رو نشون میده یعنی در واقع آخرین

رقم اعشار رو به ما نشون میده

Digits در بروکر 4 رقمی عبارت است از 4 و در بروکر 5 رقمی 5 رو نشون میده یعنی در واقع تعداد اعشار رو به

ما نشون میده

پس در واقع کدی که نوشته شده در بروکرهای 4 رقمی درست کار خواهد کرد ولی در بروکرهای 5 رقمی به

جای 10 پیپ ، 1 پیپ محاسبه خواهد شد

برای حل این مشکل راحت ترین راه این هست که هرجایی که از "10" point* استفاده کردیم ، عدد 10 رو به

ورودی ببریم تا در بروکرهای 4 رقمی 10 وارد کنیم و در 5 رقمی ها 100 وارد کنیم تا فاصله 10 پیپ درست

محاسبه بشه

من یک متغیر به نام Price_Distance در ابتدای کد تعریف میکنم و ابتدای متغیر extern مینویسم تا از ورودی

قابل تنظیم باشه بعد به جای همه 10 ها این متغیر رو قرار میدم

```
double SMA50 ;
double SMA100 ;
double MACD ;

bool BuyMode ;
bool SellMode ;

double s1B ;
double s1S ;

extern double Price_Distance = 10; //dar 4raghamiha 10 va dar 5 raghamiha 100 vared konid
extern double LotSize = 0.2 ;
int ticket;

//+-----+
//| expert initialization function |
//+-----+
int init()
{
//-----
//-----
return(0);
}
//+-----+
//| expert deinitialization function |
//+-----+
int deinit()
{
PCMBROKERS
```

```

// BUY AND SELL MODES-----+
BuyMode = ((MACD > 0) && (Bid > SMA50 + Price_Distance * Point) && (Bid > SMA100 + Price_Distance * Point));
SellMode = ((MACD < 0) && (Bid < SMA50 - Price_Distance * Point) && (Bid < SMA100 - Price_Distance * Point));

// StopLoss & TakeProfit-----+
PCM BROKERS
s1B = iLow (Symbol(), 0 , iLowest (Symbol(),0,MODE_LOW , 5 , 0 ));

//Orders Counting & Close Orders-----+

int orderscnt = 0;
for(int i=0; i < OrdersTotal(); i++)
{
    //-Shomareshe Position haye Baz
    OrderSelect(i,SELECT_BY_POS,MODE_TRADES);
    if((OrderSymbol() == Symbol()) && ((OrderType() == OP_SELL) || (OrderType() == OP_BUY)))
    {
        orderscnt++;
    }
    //-Bastane Nimeye Zvome Positione Buy
    if((OrderSymbol() == Symbol()) && (OrderType() == OP_BUY) && (OrderLots() == LotSize / 2))
    {
        if(Bid < SMA50 - Price_Distance * Point)
        {
            OrderClose (OrderTicket(), OrderLots(), Bid, 2, Black);
        }
    }
    //-Bastane Nimeye Zvome Positione Sell
    if((OrderSymbol() == Symbol()) && (OrderType() == OP_SELL) && (OrderLots() == LotSize / 2))
    {
        if(Bid > SMA50 + Price_Distance * Point)
        {
            OrderClose (OrderTicket(), OrderLots(), Ask, 2, Black);
        }
    }
    //-Ja be ja kardane S/L va Bastane Nesfe Sell
    if((OrderSymbol() == Symbol()) && (OrderType() == OP_SELL) && (OrderLots() == LotSize))
    {
        if(OrderOpenPrice() - Ask >= (s1S - OrderOpenPrice()) * 2)
        {
            PCM BROKERS
            OrderModify(OrderTicket(), Bid, OrderOpenPrice(), 0, 0, Violet);
        }
    }
}

```

پست 21 - نوشتن کد آخر و اتمام استراتژی

خیلی منتظر جواب شدم و خیلی هم امیدوار بودم که پیداش کنید.

علی الخصوص بعد از راهنمایی.ای که کردم ولی دیدم یکم زیاد طول کشید ...

تا حالا چندین بار گفتم یک بار دیگه هم میگم :

برای اینکه کد نویس خوبی باشید باید دقیقا مثل ماشین فکر کنید .

باید خودتون رو بگذارید جای ماشین و فکر تمام جاها رو بکنید...

فرض میکنیم شرایط بای برقرار شده و ما یک پوزیشن بای داریم و یک استاپ براس ست شده .

در مواقع نادری ممکن هست که قبل از اینکه به استاپ برسه شرایط عوض بشه یعنی شرط سل برقرار بشه ولی چون ما یک پوزیشن باز داریم شرط گرفتن سل برقرار نخواهد شد تا اینکه پوزیشنمون استاپ بخوره و در این حالت شاید مقدار پیپ زیادی رو در گرفتن سل از دست بدیم

یا باید برای بستن پوزیشن بای شرط $LotSize/2$ رو برداریم تا اگر از موینگ 50 به میزان 10 پیپ فاصله گرفت ببندتش

یا باید یک شرط جدید برای بستن تعریف کنیم به این منوال که اگر شرایط بای برقرار بود هر چی سل داشت رو ببند و برای سل برعکس.

```
//Orders Counting & Close Orders-----+
int orderscnt = 0;
for(int i=0; i < OrdersTotal(); i++)
{
    //-Shomareshe Position haye Baz
    OrderSelect(i,SELECT_BY_POS,MODE_TRADES);
    if((OrderSymbol() == Symbol()) && ((OrderType() == OP_SELL) || (OrderType() == OP_BUY)))
    {
        orderscnt++;
    }

    //-Bastane Positione Buy Agar Sharte Sell Bargarar Beshe
    if((OrderSymbol() == Symbol()) && (OrderType() == OP_BUY) && (SellMode))
    {
        OrderClose (OrderTicket(), OrderLots(), Bid, 2, Black);
    }

    //-Bastane Positione Sell Agar Sharte Buy Bargarar Beshe
    if((OrderSymbol() == Symbol()) && (OrderType() == OP_SELL) && (BuyMode))
    {
        OrderClose (OrderTicket(), OrderLots(), Ask, 2, Black);
    }

    //-Bastane Nimeye Zvome Positione Buy
    if((OrderSymbol() == Symbol()) && (OrderType() == OP_BUY) && (OrderLots() == LotSize / 2))
    {
        if(Bid < SMA50 -Price_Distance * Point)
        {
            OrderClose (OrderTicket(), OrderLots(), Bid, 2, Black);
        }
    }

    //-Bastane Nimeye Zvome Positione Sell
    if((OrderSymbol() == Symbol()) && (OrderType() == OP_SELL) && (OrderLots() == LotSize / 2))
    {
        if(Bid > SMA50 +Price_Distance * Point)
        {
            OrderClose (OrderTicket(), OrderLots(), Bid, 2, Black);
        }
    }
}
```

شرط های جدید برای بستن پوزیشنها هنگام برقرار شدن شرط مخالف

پست 22 - سورس فایل

سورس اکسپرت رو هم ضمیمه کردم.

با تشکر

محسن خاشعی (pumper)