//+------------------------------------------------------------------+

//| OpenOneOrder.mq4 |

//| Scriptong |

//| |

//+------------------------------------------------------------------+

#property copyright "Scriptong"

#property link ""

//---- input parameters

extern double Lots = 0.1; // Объем открываемой сделки

extern string Direction = "Buy"; // Направление сделки - Buy или Sell

extern double StartPrice = 1.23450; // Цена открытия позиции

extern double StopLoss = 1.22; // Уровень стоп-приказа или отключения

extern double TakeProfit = 1.25; // Уровень фиксации прибыли

extern int TrailingStop1 = 150; // Единоразовый перенос уровня в безубыток

extern int TrailingStop2 = 200; // Классический трейлинг

extern string OpenOrderSound = "ok.wav"; // Звук для открытия позиции

extern string StopLossSound = "alert2.wav";// Звук для уровня стопа

extern string TakeProfitSound = "news.wav";// Звук для фиксации прибыли

extern int MagicNumber = 0; // Магик для пометки своей позиции

bool Activate, IsWorkEnd, FreeMarginAlert, IsComment, FatalError;

double Spread, StopLevel, NBid, NAsk;

int TDir;

//+--------------------------------------------------------------------------------------+

//| Функция инициализации эксперта |

//+--------------------------------------------------------------------------------------+

int init()

 {

// Блок 1. Сброс всех используемых флагов ------------------------------------------------

 Activate = False;

 FatalError = False;

 IsWorkEnd = False;

 IsComment = False;

// Блок 1. -------------------------------------------------------------------------------

// Блок 2. Проверка разрешения автоторговли ----------------------------------------------

 if(!IsTradeAllowed() && !IsTradeContextBusy())

 {

 Comment("Автоматическая торговля запрещена!",

 " Измените соответсвующие настройки МТ4 и проверьте",

 " возможность автоторговли у брокера!");

 return(0);

 }

// Блок 2. -------------------------------------------------------------------------------

// Блок 3. Сбор информации об условиях торговли ------------------------------------------

 Spread = ND(MarketInfo(Symbol(), MODE\_SPREAD)\*Point);

 StopLevel = ND(MarketInfo(Symbol(), MODE\_STOPLEVEL)\*Point);

 double MinLot = MarketInfo(Symbol(), MODE\_MINLOT);

 double MaxLot = MarketInfo(Symbol(), MODE\_MAXLOT);

 double LotStep = MarketInfo(Symbol(), MODE\_LOTSTEP);

// Блок 3. -------------------------------------------------------------------------------

// Блок 4. Проверка корректности объема сделки -------------------------------------------

 if(Lots < MinLot || Lots > MaxLot)

 {

 Comment("Параметром Lots был задан неправильный объем сделки! Советник отключен!");

 return(0);

 }

 Lots = MathRound(Lots/LotStep)\*LotStep;

// Блок 4. -------------------------------------------------------------------------------

// Блок 5. Проверка корректности типа сделки (Direction) ---------------------------------

 if(Direction == "buy" || Direction == "buY" || Direction == "bUy" ||

 Direction == "bUY" || Direction == "Buy" || Direction == "BuY" ||

 Direction == "BUy" || Direction == "BUY")

 TDir = 1;

 else

 if(Direction == "sell" || Direction == "selL" || Direction == "seLl" ||

 Direction == "seLL" || Direction == "sEll" || Direction == "sElL" ||

 Direction == "sELl" || Direction == "sELL" || Direction == "Sell" ||

 Direction == "SelL" || Direction == "SeLl" || Direction == "SeLL" ||

 Direction == "SEll" || Direction == "SElL" || Direction == "SELl" ||

 Direction == "SELL")

 TDir = 2;

 else

 {

 Comment("В параметре Direction указан неизвестный тип сделки.",

 " Советник отключен!");

 return(0);

 }

// Блок 5. -------------------------------------------------------------------------------

// Блок 6. Проверка корректности параметров TrailingStop1 и TrailingStop2 ----------------

 if(ND(TrailingStop1\*Point - StopLevel) < 0)

 {

 Comment("Слишком малое значение параметра TrailingStop1. Советник отключен!");

 return(0);

 }

 if(ND(TrailingStop2\*Point - StopLevel) < 0)

 {

 Comment("Слишком малое значение параметра TrailingStop2. Советник отключен!");

 return(0);

 }

// Блок 6. -------------------------------------------------------------------------------

// Блок 7. Проверка корректности переменных StratPrice, StopLoss и TakeProfit ------------

 // Работал ли советник на этом счете?

 if(IsOwnOrder()) // да, работал, т. к. есть открытая позиция

 {

 IsWorkEnd = True;

 Comment("Советник запущен заново и отслеживает работу текущей позиции #",

 OrderTicket());

 }

 else // не работал или уже отработал, т. к. открытых позиций нет

 {

 // Проверка правильности стартовой цены и параметров StopLoss и TakeProfit

 if(TDir == 1)

 {

 // StartPrice

 if(ND(StartPrice-Ask) < 0)

 {

 Comment("Указана неправильная цена StartPrice. Советник отключен!");

 return(0);

 }

 //StopLoss

 if(ND(StartPrice-StopLoss-StopLevel-Spread) < 0)

 {

 Comment("Уровень StopLoss выставлен очень близко от StartPrice.",

 " Советник отключен!");

 return(0);

 }

 // TakeProfit

 if(ND(TakeProfit-StartPrice-StopLevel+Spread) < 0)

 {

 Comment("Уровень TakeProfit выставлен очень близко от StartPrice.",

 " Советник отключен!");

 return(0);

 }

 }

 else

 {

 // StartPrice

 if(ND(Bid-StartPrice) < 0)

 {

 Comment("Указана неправильная цена StartPrice. Советник отключен!");

 return(0);

 }

 //StopLoss

 if(ND(StopLoss-StartPrice-StopLevel-Spread) < 0)

 {

 Comment("Уровень StopLoss выставлен очень близко от StartPrice.",

 " Советник отключен!");

 return(0);

 }

 // TakeProfit

 if(ND(StartPrice-TakeProfit-StopLevel+Spread) < 0)

 {

 Comment("Уровень TakeProfit выставлен очень близко от StartPrice.",

 " Советник отключен!");

 return(0);

 }

 }

 Comment("Советник включен. Ожидается достижение цены ",

 DoubleToStr(StartPrice, Digits), " для открытия позиции ", Direction);

 }

// Блок 7. -------------------------------------------------------------------------------

 Activate = True; // Разрешаем работу советника, все параметры корректны.

//----

 return(0);

 }

//+--------------------------------------------------------------------------------------+

//| Приведение значений к точности одного тика |

//+--------------------------------------------------------------------------------------+

double ND(double A)

{

 return(NormalizeDouble(A, Digits));

}

//+--------------------------------------------------------------------------------------+

//| Функция деинициализации советника |

//+--------------------------------------------------------------------------------------+

int deinit()

 {

//----

 Comment("");

//----

 return(0);

 }

//+--------------------------------------------------------------------------------------+

//| Расшифровка сообщения об ошибке |

//+--------------------------------------------------------------------------------------+

string ErrorToString(int Error)

{

 switch(Error)

 {

 case 2: return("зафиксирована общая ошибка, обратитесь в техподдержку.");

 case 5: return("у вас старая версия терминала, обновите ее.");

 case 6: return("нет связи с сервером, попробуйте перезагрузить терминал.");

 case 64: return("счет заблокирован, обратитесь в техподдержку.");

 case 132: return("рынок закрыт.");

 case 133: return("торговля запрещена.");

 }

}

//+--------------------------------------------------------------------------------------+

//| Открытие позиции |

//| Возвращает: |

//| True - Позиция открыта успешно |

//| False - Ошибка открытия |

//+--------------------------------------------------------------------------------------+

bool OpenOrder(int Type, double Price, double SL, double TP)

{

// Блок 1. Проверки достаточности свободных средств --------------------------------------

 if(AccountFreeMarginCheck(Symbol(), OP\_BUY, Lots) <= 0 || GetLastError() == 134)

 {

 if(!FreeMarginAlert) // если недостаток свободных средств зафиксирован впервые

 {

 Print("Недостаточно средств для открытия позиции. Free Margin = ",

 AccountFreeMargin());

 FreeMarginAlert = True; // возводим флаг, чтобы не было повторных сообщений

 }

 return(False);

 }

 FreeMarginAlert = False; // Сброс флага нехватки средств, средства есть

// Блок 1. -------------------------------------------------------------------------------

// Блок 2. Перевод типа поиции в строковый вид -------------------------------------------

 if(Type == OP\_BUY)

 string S = "BUY";

 else

 S = "SELL";

// Блок 2. -------------------------------------------------------------------------------

// Блок 3. Открытие позиции --------------------------------------------------------------

 if(WaitForTradeContext()) // ожидание освобождения торгового потока

 {

 Comment("Отправлен запрос на открытие позиции...");

 int ticket=OrderSend(Symbol(), Type, Lots, Price, 0, SL, TP,

 NULL, MagicNumber, 0, CLR\_NONE); // открытие позиции

 // Попытка открытия позиции завершилась неудачей

 if(ticket<0)

 {

 int Error = GetLastError(); // запоминаем номер ошибки

 if(Error == 2 || Error == 5 || Error == 6 || Error == 64

 || Error == 132 || Error == 133) // список фатальных ошибок

 {

 Comment("Фатальная ошибка при открытии позиции т. к. "+ErrorToString(Error)+

 " Советник отключен!");

 FatalError = True; // советник не будет работать после фатальной ошибки

 }

 else

 Comment("Ошибка открытия позиции ", S, ": ", Error); // нефатальная ошибка

 return(False);

 }

 // ---------------------------------------------

 // Удачное открытие позиции

 Comment("Позиция ", S, " открыта успешно!");

 PlaySound(OpenOrderSound); // выдача звукового сигнала об успешном открытии позиции

 return(True); // успешное открытие позиции

 // ------------------------

 }

 else

 {

 Comment("Время ожидания освобождения торгового потока истекло!");

 return(False); // открыть позицию не удалось

 }

}

//+--------------------------------------------------------------------------------------+

//| Проверка существования своей позиции |

//+--------------------------------------------------------------------------------------+

bool IsOwnOrder()

{

 for(int i = 0; i < OrdersTotal(); i++) // Проверка всех позиций и ордеров счета

 if(OrderSelect(i, SELECT\_BY\_POS)) // Если позиция с номером i существует и выбрана

 if(OrderSymbol() == Symbol() && // Позиция должна быть на текущем инструменте

 OrderMagicNumber() == MagicNumber)// и с тем же магиком

 return(True);

 return(False);

}

//+--------------------------------------------------------------------------------------+

//| Двухуровневый трейлинг-стоп |

//| TrailingStop1 - выставляется один раз после достижения нужного кол-ва пунктов прибыли|

//| TrailingStop2 - движущийся. Включается после отработки трейлинг-стопа первого уровня |

//+--------------------------------------------------------------------------------------+

void TrailingStopDoubleLevel()

{

// Блок 1. Нормализация основных характеристик позиции -----------------------------------

 {

 double OpPrice = ND(OrderOpenPrice()); // Приведение цены открытия к нужной точности

 double SL = ND(OrderStopLoss()); // Приведение стоп-приказа к нужной точности

// Блок 1. -------------------------------------------------------------------------------

// Блок 2. Классический трейлинг длинной позиции -----------------------------------------

 if(OrderType() == OP\_BUY)

 if(ND(SL-OpPrice) >= 0) // Если стоп выше или равен цене открытия позиции

 {

 if(ND(NBid-Point\*TrailingStop2-SL) > 0) // Если BID больше стопа на TrailingStop2

 if(WaitForTradeContext()) // Ждем освобождение торгового потока

 if(!OrderModify(OrderTicket(), 0,

 ND(NBid-TrailingStop2\*Point), OrderTakeProfit(), 0)) // Модификация

 Print("Не удалось изменить ордер BUY. Второй трал. Ошибка №",

 GetLastError(), ", Старый: ", SL, ", Новый: ",

 NBid-TrailingStop2\*Point); // Развернутое сообщение об ошибке

 }

// Блок 2. -------------------------------------------------------------------------------

// Блок 3. Перенос стопа в безубыток длинной позиции -------------------------------------

 else

 {

 if(ND(NBid-OpPrice-TrailingStop1\*Point) > 0 && // Если BID больше цены открытия

 ND(NBid-OpPrice-StopLevel) > 0) // на TrailingStop1 пунктов и дальше, чем

 if(WaitForTradeContext()) // уровень стопов

 if(!OrderModify(OrderTicket(), 0, OpPrice,

 OrderTakeProfit(), 0)) // модификация

 Print("Не удалось изменить ордер BUY. Первый трал. Ошибка №", // сообщение

 GetLastError(), ", Старый: ", SL, ", Новый: ", OpPrice);// в журнал

 }

// Блок 3. -------------------------------------------------------------------------------

// Блок 4. Классический трейлинг короткой позиции ----------------------------------------

 if(OrderType()==OP\_SELL) // Для короткой позиции

 if(ND(OpPrice-SL) >= 0 && SL != 0) // Стоп ниже цены открытия и не равен нулю

 {

 if(ND(OpPrice-NAsk-StopLevel) > 0 && // ASK ниже цены открытия на StopLevel

 ND(SL-Point\*TrailingStop2-NAsk) > 0) // и ниже стопа на TrailingStop2 пунктов

 if(WaitForTradeContext()) // Ожидаем освобождения торгового потока

 if(!OrderModify(OrderTicket(), 0, ND(NAsk+TrailingStop2\*Point),

 OrderTakeProfit(), 0)) // модификация

 Print("Не удалось изменить ордер SELL. Второй трал. Ошибка №",

 GetLastError(), ", Старый: ", SL, ", Новый: ", NAsk+TrailingStop2\*Point);

 }

// Блок 4. -------------------------------------------------------------------------------

// Блок 5. Перенос стопа в безубыток короткой позиции ------------------------------------

 else

 {

 if(ND(OpPrice-NAsk-Point\*TrailingStop1) > 0 &&//ASK ниже открытия на TrailingStop1

 ND(OpPrice-NAsk-StopLevel) > 0) // и на StopLevel пунктов

 if(WaitForTradeContext()) // Ожидаем торговый поток

 if(!OrderModify(OrderTicket(), 0, OpPrice, OrderTakeProfit(), 0))//модификация

 Print("Не удалось изменить ордер SELL. Первый трал. Ошибка №",

 GetLastError(), ", Старый: ", SL, ", Новый: ", OpPrice);

 }

// Блок 5. -------------------------------------------------------------------------------

 }

 return(0);

 }

//+--------------------------------------------------------------------------------------+

//| Ожидание торгового потока. Если поток свободен, то результат True, иначе - False |

//+--------------------------------------------------------------------------------------+

bool WaitForTradeContext()

{

 int P = 0;

 // цикл "пока"

 while(IsTradeContextBusy() && P < 5)

 {

 P++;

 Sleep(1000);

 }

 // -------------

 if(P == 5)

 return(False);

 return(True);

}

//+--------------------------------------------------------------------------------------+

//| Выяснение причины закрытия последней позиции: |

//| 0 - по TakeProfit |

//| 1 - по StopLoss |

//| 2 - ручное закрытие (между TP и SL) |

//| 3 - неизвестная ошибка или закрытая позиция не найдена) |

//+--------------------------------------------------------------------------------------+

int CloseReason()

{

// Блок 1. Поиск своей позиции -----------------------------------------------------------

 for(int i = OrdersHistoryTotal()-1; i >= 0; i--) // Поиск среди закрытых позиций

 if(OrderSelect(i, SELECT\_BY\_POS, MODE\_HISTORY)) // Позиция выбрана

 if(OrderSymbol() == Symbol() && // По текущему инструменту

 OrderMagicNumber() == MagicNumber) // и с нужным магиком

 break; // Нашли - прерывается цикл

// Блок 1. -------------------------------------------------------------------------------

 if(i >= 0) // Цикл не дошел до конца, значит, поиск увенчался успехом

 {

// Блок 2. Выяснение причины для длинной позиции -----------------------------------------

 if(OrderType() == OP\_BUY)

 {

 if(ND(OrderClosePrice()-OrderTakeProfit()) >= 0 // Если цена закрытия больше или

 && OrderTakeProfit() != 0) // равна TakeProfit и он был

 return(0); // установлен, то закрыто по TakeProfit

 if(ND(OrderStopLoss()-OrderClosePrice()) >= 0 && // Если цена закрытия меньше или

 OrderStopLoss() != 0) // равна StopLoss и он был

 return(1); // установлен, то закрыто по StopLoss

 return(2); // иначе - закрыто вручную

 }

// Блок 2. -------------------------------------------------------------------------------

// Блок 3. Выяснение причины для короткой позиции ----------------------------------------

 if(OrderType() == OP\_SELL)

 {

 if(ND(OrderTakeProfit()-OrderClosePrice()) >= 0 // Если цена закрытия меньше или

 && OrderTakeProfit() != 0) // равна TakeProfit и он был

 return(0); // установлен, то закрыто по TakeProfit

 if(ND(OrderClosePrice()-OrderStopLoss()) >= 0 // Если цена закрытия меньше или

 && OrderStopLoss() != 0) // равна StopLoss и он был

 return(1); // установлен, то закрыто по StopLoss

 return(2); // иначе - закрыто вручную

 }

 }

// Блок 3. -------------------------------------------------------------------------------

 return(3); // неизвестная ошибка

}

//+--------------------------------------------------------------------------------------+

//| Функция start эксперта |

//+--------------------------------------------------------------------------------------+

int start()

 {

// Блок 1. Разрешено ли советнику работать? ----------------------------------------------

 if(!Activate || IsComment || FatalError) // Отключается работа советника, если функция

 return(0); // init завершилась с ошибкой или выдано со-

 // общение об окончании работы эксперта, или

 // имела место фатальная ошибка

// Блок 1. -------------------------------------------------------------------------------

// Блок 2. сбор информации о свойствах инструмента ---------------------------------------

 Spread = ND(MarketInfo(Symbol(), MODE\_SPREAD)\*Point); // текущий спрэд по инструменту

 StopLevel = ND(MarketInfo(Symbol(), MODE\_STOPLEVEL)\*Point); //текущий уровень стопов

 NBid = ND(Bid); // нормализованная текущая цена BID

 NAsk = ND(Ask); // нормализованная текущая цена ASK

// Блок 2. -------------------------------------------------------------------------------

// Блок 3 - Действия при наличии открытой позиции ----------------------------------------

 if(IsOwnOrder()) // если есть открытая советником позиция

 {

 TrailingStopDoubleLevel(); // пытаемся подтянуть стоп

 return(0); // выход до следующего тика

 }

// Блок 3. -------------------------------------------------------------------------------

// Блок 4 - Действия при отсутствии открытой позиции -------------------------------------

 else

// Блок 4.1 - Позиция еще не была открыта ------------------------------------------------

 if(!IsWorkEnd) // если не поднят флаг окончания работы, то

 {

// Блок 4.1.1 - Направление "вверх" (длинная позиция) ------------------------------------

 if(TDir == 1) // если направление "вверх"

 {

 if(ND(NBid-StopLoss) <= 0) // BID меньше уровня StopLoss, заканчиваем работу

 {

 IsComment = True; // Поднимается флаг выдачи сообщения о конце работы

 Comment("Работа советника завершена достижением уровня ",

 "стопа до открытия позиции!"); // Выдача сообщения на экран

 PlaySound(StopLossSound); // Звуковое оповещение о достижении уровня стопа

 }

 else // BID больше уровня StopLoss

 // Проверка допустимости уровней стоп-приказа и профита

 if(ND(NAsk-StartPrice) >= 0 && // ASK больше или равен StartPrice

 ND(NBid-StopLoss-StopLevel) >= 0 && // а BID больше StopLoss на StopLevel

 ND(TakeProfit-NBid-StopLevel) >= 0) // TakeProfit дальше, чем StopLevel

 if(OpenOrder(OP\_BUY, NAsk, StopLoss, TakeProfit))//открытие длинной позиции

 IsWorkEnd = True; // Поднимаем флаг открытия позиции

 }

// Блок 4.1.1 ----------------------------------------------------------------------------

// Блок 4.1.2 - Направление "вниз" (короткая позиция) ------------------------------------

 else

 {

 if(ND(StopLoss-NAsk) <= 0) // ASK больше StopLoss, заканчиваем работу

 {

 IsComment = True; // Поднимается флаг выдачи сообщения о завершении работы

 Comment("Работа советника завершена достижением уровня ",

 "стопа до открытия позиции!"); // Выдача сообщения на экран

 PlaySound(StopLossSound); // Звуковое оповещение о достижении уровня стопа

 }

 else // ASK меньше уровня StopLoss

 // Проверка допустимости уровней стоп-приказа и профита

 if(ND(NBid-StartPrice) <= 0 && // BID меньше или равен StartPrice

 ND(NAsk-TakeProfit-StopLevel) >= 0 &&// ASK больше TP на StopLevel пунктов

 ND(StopLoss-NAsk-StopLevel) >= 0) // SL больше ASK более чем на StopLevel

 if(OpenOrder(OP\_SELL, NBid, StopLoss, TakeProfit)) // открытие позиции

 IsWorkEnd = True; // Поднимаем флаг открытия позиции

 }

// Блок 4.1.2 ----------------------------------------------------------------------------

 }

// Блок 4.1 ------------------------------------------------------------------------------

// Блок 4.2 - Позиция была закрыта на прошлом тике ---------------------------------------

 else // Позиции нет, но в текущем сеансе она открывалась

 {

 IsComment = True; // Поднимается флаг выдачи сообщения о завершении работы

 switch(CloseReason()) // Определение причины закрытия позиции

 {

 case 0: // позиция была закрыта по профиту

 Comment("Работа советника завершена закрытием позиции по TakeProfit!");

 PlaySound(TakeProfitSound); // Звуковое оповещение закрытия по прибыли

 break;

 case 1: // позиция была закрыта по стопу

 Comment("Работа советника завершена закрытием позиции по StopLoss...");

 PlaySound(StopLossSound); // Звуковое оповещение закрытия по стопу

 break;

 case 2: // позиция была закрыта вручную

 Comment("Работа советника завершена ручным закрытием позиции!");

 break;

 case 3: // неизвестная ошибка

 Comment("Работа советника завершена из-зи неизвестной ошибки.");

 break;

 }

 }

// Блок 4.2 ------------------------------------------------------------------------------

// Блок 4. -------------------------------------------------------------------------------

 return(0);

 }

//+------------------------------------------------------------------+