```
//+------------------------------------------------------------------+
//|                                  Script_HighlightMissingDays.mq4 |
//|                                                        M Wilson  |
//|                                             https://www.mql5.com |
//+------------------------------------------------------------------+
#include <C_Chart_Drawing.mqh>

#property copyright "M Wilson"
#property link      "https://www.mql5.com"
#property version   "1.00"
#property strict
#property show_inputs

//+------------------------------------------------------------------+
//| Inputs                                                           |
//+------------------------------------------------------------------+
extern int I_TradingStartHour=6;
//Start Hour (BROKER Time), 5 would be 5:00
extern int I_TradingEndHour=21;
//End Hour (BROKER Time), 20 would be 20:00
extern int I_WeekendCloseHour=19;                //Weekend Close Hour (BROKER Time)
extern int I_NoTradingLineWidth=1;
//Line Width of lines where we cannot trade.   1, gives style_dot.
extern color I_NoTradingLineColor=clrWhite;      //Color of lines where we cannot trade.
extern int I_WeekendLineWidth=5;
//Line Width of lines where we have a weekend.
extern color I_WeekendLineColor=clrLightBlue;    //Color of lines where we cannot trade.
extern bool I_DeleteVerticalLinesBeforeRun=True; //Delete Vertical Bars.
//+------------------------------------------------------------------+
//| Script program start function                                    |
//+------------------------------------------------------------------+
void OnStart()
  {

   C_Chart_Drawing *objCD = new C_Chart_Drawing();
   int intBars=Bars;

   //Delete any vertial lines
   if(I_DeleteVerticalLinesBeforeRun)  objCD.DeleteAllVerticleLines();

   //Scan through the bars and find any large changes in dates.
   for(int i=1;i<intBars;i++)
   {
      datetime dtCandleT=iTime(Symbol(),0,i-1);
      datetime dtCandleTm1=iTime(Symbol(),0,i);

      //If there is more than 1 day between 2 candles, then highlight them.
      if(dtCandleT-dtCandleTm1>60*60*24)
      {
         string strNameT="MisDay_"+IntegerToString(dtCandleT);
         string strNameTm1="MisDay_"+IntegerToString(dtCandleTm1);
         objCD.CreateVerticalLine(0,strNameT,0,dtCandleT,I_WeekendLineColor,STYLE_SOLID,
I_WeekendLineWidth,True);
         objCD.CreateVerticalLine(0,strNameTm1,0,dtCandleTm1,I_WeekendLineColor,
STYLE_SOLID,I_WeekendLineWidth,True);
      }
      else if(!CanWeCreateNewTrades(dtCandleT))
      {
         string strNameT="NonTrade_"+IntegerToString(dtCandleT);
         ENUM_LINE_STYLE eStyle=STYLE_DOT;
         if(I_NoTradingLineWidth>1) eStyle=STYLE_SOLID;
         objCD.CreateVerticalLine(0,strNameT,0,dtCandleT,I_NoTradingLineColor,eStyle,
I_NoTradingLineWidth,True);
      }

   }

   objCD.RefreshChart();

   delete objCD;
```

```
  }
//+------------------------------------------------------------------+
bool CanWeCreateNewTrades(const datetime dtCandleT)
{
   bool boolRet = True;



//Evaluate if we are within the trading day.   If the start hour is greater than the end hou
   //we are day trading.

//PLEASE NOTE THAT WHEN YOU RUN THE STRATEGY SIMULATOR, IT MAKES THE LOCAL TIME AND CURRENT
   //TO BUILD A STRATEGY BASED AROUND CURRENTTIME RATHER THAN LOCALTIME.
   int intCurrHour = TimeHour(dtCandleT);
   if(I_TradingStartHour>I_TradingEndHour)
   {

//If we are NOT (greater than or equal to the trading start hour OR less than the End tradir
      if(!(intCurrHour>=I_TradingStartHour || intCurrHour<I_TradingEndHour))  boolRet=
False;
   }
   else
   {

//If we are NOT (greater than or equal to the trading start hour and less than the End Tradi
      if(!(intCurrHour>=I_TradingStartHour && intCurrHour<I_TradingEndHour))  boolRet=
False;
   }


//Ensure we cannot trade at any point in time greater than or equal to the weekend close hou
   if(IsPreWeekendDay(dtCandleT))
   {
      if(intCurrHour>=I_WeekendCloseHour) boolRet=False;
   }

   return boolRet;
}
bool IsPreWeekendDay(const datetime dtCandleT)
{
   bool boolRet=False;

   //0 is sunday, 5 is Friday
   if(TimeDayOfWeek(dtCandleT)==5) boolRet=True;

   return boolRet;
}
```