```
//+------------------------------------------------------------------+
//|                                                         MACD.mq4 |
//|                                Copyright © 2005, David W. Thomas |
//|                                       mailto:davidwt@usa.net     |
//+------------------------------------------------------------------+
// This is the correct computation and display of MACD.
#property copyright "Copyright © 2005, David W. Thomas"
#property link      "mailto:davidwt@usa.net"
#property indicator_separate_window
#property indicator_buffers 2
#property indicator_color1 Blue
#property indicator_color2 Red
//#property indicator_color3 Green
//---- input parameters
extern int       FastMAPeriod=12;
extern int       SlowMAPeriod=26;
extern int       SignalMAPeriod=9;
//---- buffers
double MACDLineBuffer[];
double SignalLineBuffer[];
double HistogramBuffer[];
//---- variables
double alpha = 0;
double alpha_1 = 0;
//+------------------------------------------------------------------+
//| Custom indicator initialization function                         |
//+------------------------------------------------------------------+
int init()
  {
IndicatorDigits(MarketInfo(Symbol(),MODE_DIGITS)+1);
//---- indicators
SetIndexStyle(0,DRAW_LINE);
SetIndexBuffer(0,MACDLineBuffer);
SetIndexDrawBegin(0,SlowMAPeriod);
SetIndexStyle(1,DRAW_LINE,STYLE_DOT);
SetIndexBuffer(1,SignalLineBuffer);
SetIndexDrawBegin(1,SlowMAPeriod+SignalMAPeriod);
//   SetIndexStyle(2,DRAW_HISTOGRAM);
//   SetIndexBuffer(2,HistogramBuffer);
//   SetIndexDrawBegin(2,SlowMAPeriod+SignalMAPeriod);
//---- name for DataWindow and indicator subwindow label
IndicatorShortName("MACD("+FastMAPeriod+","+SlowMAPeriod+","+SignalMAPeriod+")");
SetIndexLabel(0,"MACD");
SetIndexLabel(1,"Signal");
//----
   alpha = 2.0 / (SignalMAPeriod + 1.0);
   alpha_1 = 1.0 - alpha;
//----
   return(0);
  }
//+------------------------------------------------------------------+
//| Custor indicator deinitialization function                       |
//+------------------------------------------------------------------+
int deinit()
  {
//----
//----
   return(0);
  }
//+------------------------------------------------------------------+
//| Custom indicator iteration function                              |
//+------------------------------------------------------------------+
int start()
  {
   int limit;
   int counted_bars = IndicatorCounted();
//---- check for possible errors
   if (counted_bars<0) return(-1);
//---- last counted bar will be recounted
   if (counted_bars>0) counted_bars--;
   limit = Bars - counted_bars;
   for(int i=limit; i>=0; i--)
     {
      MACDLineBuffer[i] = iMA(NULL,0,FastMAPeriod,0,MODE_EMA,PRICE_CLOSE,i) - iMA(NULL,0,SlowMAPeriod,0,MODE_EMA,PRICE_CLOSE,i);
      SignalLineBuffer[i] = alpha*MACDLineBuffer[i] + alpha_1*SignalLineBuffer[i+1];
      HistogramBuffer[i] = MACDLineBuffer[i] - SignalLineBuffer[i];
     }
//----
   return(0);
  }
//+------------------------------------------------------------------+
```