

```
//Hull Moving Average.mq4
```

```
#property indicator_chart_window  
#property indicator_buffers 2  
#property indicator_color1 Magenta  
#property indicator_color2 Aqua  
#property indicator_width1 4  
#property indicator_width2 4
```

← #property命令を記述

```
//インジケータバッファの宣言
```

```
double Up[];  
double Down[];  
double Ex_MA[];  
double Hull[];
```

← インジケータバッファを宣言

```
//変数の宣言
```

```
extern int HMA_Period = 48;  
extern int Applied_Price = 0;
```

← 変数を宣言

```
int init()
```

```
{  
    IndicatorBuffers(4);  
  
    //インジケータバッファのインデックス  
    SetIndexBuffer(0,Up);  
    SetIndexBuffer(1,Down);  
    SetIndexBuffer(2,Ex_MA);  
    SetIndexBuffer(3,Hull);  
  
    //インジケータのラベル  
    SetIndexLabel(0,"Up");  
    SetIndexLabel(1,"Down");  
  
    //インジケータのスタイル  
    SetIndexStyle(0,DRAW_LINE,STYLE_SOLID);  
    SetIndexStyle(1,DRAW_LINE,STYLE_SOLID);  
  
    //インジケータの描画開始時点  
    SetIndexDrawBegin(0,HMA_Period + 20);  
    SetIndexDrawBegin(1,HMA_Period + 20);  
  
    return(0);  
}
```

← 基本設定を記述

```
int start()
```

```
{  
    int limit = Bars - IndicatorCounted();  
  
    //Hull移動平均線の計算  
    for(int i = limit - 1; i >= 0; i--)  
    {  
        double Half_HMA_Period = HMA_Period / 2;  
        Half_HMA_Period = MathFloor(Half_HMA_Period);  
  
        Ex_MA[i] = 2 * iMA(NULL,0,Half_HMA_Period,0,MODE_LWMA,Applied_Price,i)  
                - iMA(NULL,0,HMA_Period,0,MODE_LWMA,Applied_Price,i);  
    }  
  
    for(i = limit - 1; i >= 0; i--)  
    {  
        double Square_Period = MathSqrt(HMA_Period);  
        Square_Period = MathFloor(Square_Period);  
  
        Hull[i] = iMAOnArray(Ex_MA,0,Square_Period,0,MODE_LWMA,i);  
        Hull[i] = NormalizeDouble(Hull[i],MarketInfo(Symbol(),MODE_DIGITS));  
    }  
  
    //カラーコーディング  
    for(i = limit - 1; i >= 0; i--)  
    {  
        if(Hull[i+1] <= Hull[i])  
        {  
            Up[i] = Hull[i];  
  
            if(Hull[i+2] >= Hull[i+1])  
            {  
                Up[i+1] = Hull[i+1];  
            }  
            Down[i] = EMPTY_VALUE;  
        }  
        else if(Hull[i+1] >= Hull[i])  
        {  
            Down[i] = Hull[i];  
  
            if(Hull[i+2] <= Hull[i+1])  
            {  
                Down[i+1] = Hull[i+1];  
            }  
            Up[i] = EMPTY_VALUE;  
        }  
    }  
  
    return(0);  
}
```

← 具体的な処理内容を記述