```
//MA-Dev-R.mq4

#property indicator_separate_window

#property indicator_buffers 1

#property indicator_color1 SpringGreen

#property indicator_width1 1

#property indicator_level1 0

//インジケーターのバッファーの宣言
double MA_Dev_R[];

//変数の宣言
extern int MA_Period           = 25;
extern int MA_Method           = 0;
extern int MA_Applied_Price    = 0;
extern int Applied_Price       = 0;

int init()
{
 //インジケーターバッファーのインデックス
 SetIndexBuffer(0,MA_Dev_R);

 //インジケーターのラベル
 SetIndexLabel(0,"MA-Dev-R");
 IndicatorShortName("MA-Dev-R ("+(string)MA_Period+",
                    "+(string)MA_Method+",
                    "+(string)Applied_Price+")");

 //インジケーターのスタイル
 SetIndexStyle(0,DRAW_LINE,STYLE_SOLID);

 //インジケーターの描画開始時点
 SetIndexDrawBegin(0,MA_Period + 10);

 return(0);
}

int start()
{
 if(Bars < MA_Period)
   {
    return(0);
   }

 int limit = Bars - IndicatorCounted();

 for(int i = limit - 1; i >= 0; i--)
    {
     double MA = iMA(NULL,0,MA_Period,0,MA_Method,MA_Applied_Price,i);
     MA        = NormalizeDouble(MA,Digits);

     if(MA != 0)
       {
        double Value = 0;

        switch(Applied_Price)
              {
               case 0:
                 Value = Close[i];

                 MA_Dev_R[i] = (Value - MA) / MA * 100;
                 MA_Dev_R[i] = NormalizeDouble(MA_Dev_R[i],Digits);

                 break;

               case 1:
                 Value = Open[i];

                 MA_Dev_R[i] = (Value - MA) / MA * 100;
                 MA_Dev_R[i] = NormalizeDouble(MA_Dev_R[i],Digits);

                 break;

               case 2:
                 Value = High[i];

                 MA_Dev_R[i] = (Value - MA) / MA * 100;
                 MA_Dev_R[i] = NormalizeDouble(MA_Dev_R[i],Digits);

                 break;

               case 3:
                 Value = Low[i];

                 MA_Dev_R[i] = (Value - MA) / MA * 100;
                 MA_Dev_R[i] = NormalizeDouble(MA_Dev_R[i],Digits);

                 break;

               case 4:
                 Value = (High[i] + Low[i]) / 2;

                 MA_Dev_R[i] = (Value - MA) / MA * 100;
                 MA_Dev_R[i] = NormalizeDouble(MA_Dev_R[i],Digits);

                 break;

               case 5:
                 Value = (High[i] + Low[i] + Close[i]) / 3;

                 MA_Dev_R[i] = (Value - MA) / MA * 100;
                 MA_Dev_R[i] = NormalizeDouble(MA_Dev_R[i],Digits);

                 break;

               case 6:
                 Value = (High[i] + Low[i] + Close[i] + Close[i]) / 4;

                 MA_Dev_R[i] = (Value - MA) / MA * 100;
                 MA_Dev_R[i] = NormalizeDouble(MA_Dev_R[i],Digits);

                 break;
              }
       }
    }

 return(0);
}
```

#property命令を記述

インジケーターバッファーを宣言

変数を宣言

基本設定を記述

具体的な処理内容を記述