

```

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);
}

```