Mt4 Expert Advisor Programming Tutorial

Mastering MT4 Expert Advisor Programming: A Step-by-Step Tutorial

For traders who want to automate their strategies, MetaTrader 4 (MT4) is the most widely used platform. Its built-in scripting language, MQL4, allows users to create custom Expert Advisors (EAs) that can trade on their behalf 24/7. This **mt4 expert advisor programming tutorial** will walk you through the entire process—from setting up your environment to deploying a fully functional automated system.

1. What Is MT4 and Why Use It?

MetaTrader 4 is a trading platform that provides advanced charting, technical analysis tools, and a powerful API for automated trading. Traders can run scripts, indicators, and EAs written in MQL4, a C-like language specifically designed for financial markets.

2. Understanding Expert Advisors

Expert Advisors are automated programs that analyze market conditions and execute trades without human intervention. They can:

- Monitor price movements in real time
- Place orders based on predefined rules
- Manage risk by setting stop-loss and take-profit levels
- Adapt to changing market conditions through built-in logic

3. Why Program an EA Instead of Manual Trading?

Automated trading offers several advantages:

1. **Speed** – Orders are executed instantly.
2. **Emotion-free** – No fear or greed influences decisions.
3. **Backtesting** – Strategies can be tested against historical data.
4. **Consistency** – Rules are applied uniformly every time.

4. Overview of MQL4 Syntax

MQL4 shares many features with C and C++, making it approachable for programmers familiar with those languages. Key concepts include:

- **Data Types** – int, double, string, bool.
- **Variables** – Declare with `int count = 0;`.
- **Functions** – `int OnInit()`, `void OnTick()`, `void OnDeinit()`.
- **Built-in Variables** – `Bid`, `Ask`, `Time[0]`.
- **Arrays** – For storing series of data.

Downloaded from www.suitsbysuits.com by Brady & Guerra • Knowledge is power. • Page 1 • 2024-11-06

5. Setting Up Your Development Environment

Follow these steps to prepare for coding:

1. **Install MetaTrader 4** – Download from your broker's website.
2. **Open the MetaEditor** – Press F4 or click "MetaEditor" from the toolbar.
3. **Create a New EA** – File > New > Expert Advisor (template).
4. **Save Your Project** – Give it a descriptive name like `MyFirstEA.mq4`.

6. The Basic Structure of an EA

Every EA contains three essential functions:

- **OnInit()** – Runs once when the EA is attached.
- **OnTick()** – Executes on every new market tick.
- **OnDeinit()** – Runs when the EA is removed.

Here's a skeleton to get you started:

```
#property strict

int OnInit() {
   // Initialization code
   return(INIT_SUCCEEDED);
}

void OnTick() {
   // Main trading logic
}

void OnDeinit(const int reason) {
   // Cleanup code
}
```

7. Adding Trading Logic

Let's build a simple moving-average crossover strategy. The EA will buy when the short-term MA crosses above the long-term MA and sell when the opposite occurs.

7.1. Define Parameters

Use `input` to allow users to tweak settings without recompiling.

```
input int FastPeriod = 10;      // Short MA period
input int SlowPeriod = 30;      // Long MA period
input double LotSize = 0.1;     // Trade volume
```

7.2. Calculate MAs

Use the built-in `iMA()` function.

```
double fastMA = iMA(NULL, 0, FastPeriod, 0, MODE_SMA, PRICE_CLOSE, 1);
double slowMA = iMA(NULL, 0, SlowPeriod, 0, MODE_SMA, PRICE_CLOSE, 1);
```

7.3. Detect Crossovers

Compare the current and previous values.

```
bool bullish = (iMA(NULL,0,FastPeriod,0,MODE_SMA,PRICE_CLOSE,1) > iMA(NULL,0,SlowPeriod,0,MODE_SMA,PRICE_CLOSE,1)) && (iMA(NULL,0,FastPeriod,0,MODE_SMA,PRICE_CLOSE,2) <=
iMA(NULL,0,SlowPeriod,0,MODE_SMA,PRICE_CLOSE,2)); bool bearish = (iMA(NULL,0,FastPeriod,0,MODE_SMA,PRICE_CLOSE,1) < iMA(NULL,0,SlowPeriod,0,MODE_SMA,PRICE_CLOSE,1)) &&
(iMA(NULL,0,FastPeriod,0,MODE_SMA,PRICE_CLOSE,2)
```

Downloaded from www.suitsbysuits.com by Brady & Guerra • Knowledge is power. • Page 3 • 2024-11-06