

# MQL4 COURSE

By Coders' guru [www.forex-tsd.com](http://www.forex-tsd.com)

-2

## SYNTAX

**Doufám**, že se vám líbila uvítací lekce, ve které jsem se snažil zodpovědět nejzákladnější otázky: co je MQL 4, proč MQL a kde psát v MQL 4?

Vždy největší a nejdůležitější otázkou(ami) je „jak“ – v této lekci se to dozvíte.

Nyní bych rád, kdybyste si uvolnili mysl a pečlivě si přečetli několik dalších konceptů.

Ted' se budeme bavit o pravidlech funkce SYNTAX v MQL4.

A tak, jak jsem už řekl dříve, pokud programujete v programu C (nebo ve vylepšeném C++), pak již o programu MQL 4 víte mnohé i bez mých lekcí, protože syntaxe MQL4 jsou velmi podobné syntaxím C.

Slovníkový význam pojmu SYNTAX v programovacích jazycích:

“Sestava povolených vyhrazených slov, jejich parametrů a správného uspořádání výrazů se nazývá syntaxe jazyka”. “Wikipedia”

Takže, pokud studujeme syntaxi jazyka, studujeme především jeho gramatiku a zapisujeme pravidla, z nichž se skládá:

- . • Formát
- . • Komentáře
- . • Identifikátory
- . • Rezervovaná slova

Nyní rozřízneme dort.

### 1-Formát:

Pokud zapíšete váš kód, můžete volně používat jakoukoliv sadu vymezení, svící a prázdných řádků a oddělovat váš kód a řádek kódu tak, aby se vše bylo příjemné pro čtení.

Např. všechny tyto řádky jsou v MQL 4 platné:

```
double MacdCurrent, MacdPrevious, SignalCurrent;
```

```
double  
MacdCurrent,  
MacdPrevious,  
SignalCurrent;
```

```
double           MacdCurrent,           MacdPrevious,           SignalCurrent;
```

Je však vidět, že první řádek je čitelnější a lépe srozumitelný. Jako všude jinde, i toto pravidlo má své výjimky: 1- Nemůžete používat řádky v “Kontrolní kompilaci”

Více se o „Kontrolní kompilaci“ dozvíte v další lekci, zatím si jen pamatujte, že se jedná o výjimku. Např. další řádek je v kompilátoru MQL 4 neplatný:

```
#property  
copyright "Copyright © 2004, MetaQuotes Software Corp."
```

Toto je platná “Kontrolní kompilace”:

```
#property copyright "Copyright © 2004, MetaQuotes Software Corp."
```

2- Nemůžete použít nový řádek nebo mezeru uprostřed konstantních hodnot, identifikátorů nebo klíčových slov.

Např. tento řádek je platný:

```
extern int MA_Period=13;
```

“extern” a “int” jsou zde klíčovými slovy, “MA\_Period” je identifikátorem a “13” je konstantní hodnotou.

V dalších lekcích se dozvíte více.

Příklad neplatných řádků:

```
extern int MA_Period=1
3;
```

```
extern int MA_Period=1    3;
```

Všimněte si svíce mezi řádkem 1 a 3.

```
ex
tern int MA_Period=13;
```

## 2- Komentáře:

K usnadnění programování má každý program svůj systém zapisování komentářů. Můžete použít komentáře pro zápis vašich kódů, které bude kompilátor ignorovat a vy je učiníte více srozumitelnými. Předpokládejme, že program zapíšete např. v létě a v zimě si jej budete chtít přečíst. Bez komentářů pro vás bude těžké porozumět všemu, i když jste třeba samotným tvůrcem programu.

MQL 4 (& C /C ++ ) používá dva druhy komentářů:

### 1- jednořádkové komentáře

Jednořádkový komentář začíná znakem “//” a končí novým řádkem:

Například:

```
//This is a comment
extern int MA_Period=13;
```

```
extern int MA_Period=13; //This is another comment
```

## 2-Víceřádkové komentáře

Víceřádkový komentář začíná znakem “/\* ”a končí znakem “\* /”.

A můžete komentovat jiný řádek nebo více vložením znaku “/\*” na začátek prvního řádku a “\* /” na konec posledního.

Například:

```
/* this  
is  
multi  
line  
comment*/
```

Můžete rovněž vnořit komentáře uvnitř víceřádkových komentářů:

```
/* this  
is  
multi //another comment nested here.  
line  
comment*/
```

Toto je rovněž platný komentář:

```
extern int /*HELLO! I'm a comment*/ MA_Period=13;
```

Toto je ale neplatný komentář:

```
extern int //test MA_Period=13;
```

### 3-Identifikátory:

Identifikátor je pojmenování, které zvolíte pro své proměnné konstanty a funkce. Např. MA\_Period je zde identifikátorem:

```
extern int MA_Period=13;
```

Pro zvolení těchto jmen platí několik pravidel a omezení:

- 1-Délka identifikátoru nesmí přesáhnout 31 znaků.
- 2-Identifikátor musí začínat písmenem (velkým či malým) nebo symbolem podtržení `_`. Prostřednictvím číslice nebo jiného symbolu identifikátor aktivován být nemůže.
- 3-Nemůžete používat žádná rezervovaná slova jako identifikátoru.  
Seznam rezervovaných slov vám rovněž brzy ukáží.
- 4- Jména identifikátorů jsou citlivá na velká a malá písmena, takže MA\_PERIOD není totéž co MA\_Period

Nyní uvádím několik příkladů:

<b>Name1</b>	Valid
<b>_Name1</b>	Valid
<b>1Name</b>	Invalid (don't start with number)
<b>~Name1</b>	Invalid (you can only use underline symbol)
<b>N~ame1</b>	Invalid (you can only use underline symbol)
<b>i_love_my_country_and_my_country_loves_all_the_world</b>	Invalid (you can't exceed the 31 characters length)
<b>Color</b>	Valid
<b>color</b>	Invalid (you can't use reversed word, and <b>color</b> is one of them)

### 4-Vyhrazená slova:

Jsou slova, která jazyk používá pro specifické úkony. Jsou tedy vyhrazeny pro potřeby jazyka a nemůžete je používat k pojmenování identifikátorů ani pro jiné účely.

Zde je uveden seznam vyhrazených slov (z průvodce MQ L 4):

Datové typy	Paměťové třídy	Operátory	Ostatní
bool	extern	break	false
color	static	case	true
datetime		continue	
double		default	

int		else	
string		for	
void		if	
		return	
		switch	
		while	

Zde je příklad neplatných kódů:

```
extern int datetime =13;  
int extern =20;  
double continue = 0;
```

Doufám, že se vám lekce líbila.  
Další lekce bude zaměřena na datové typy.  
Buďte tedy připraveni, opravdová tvrdá práce přichází!

Veškeré dotazy a návrhy srdečně vítám.

S pozdravem  
Coders'Guru  
20 -10 -20 05