

Jak na Excel

Obsah cvičení

M-language

Teorie

Strukturované data

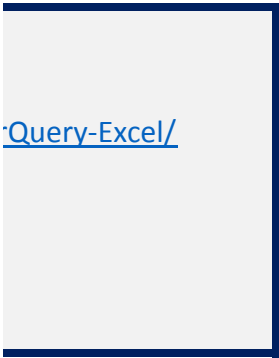
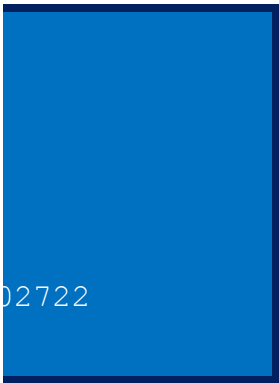
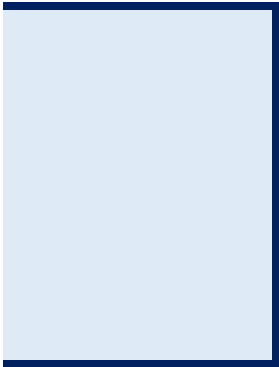
Pavel Lasák

Lektor, expert na Microsoft Excel, držitel prestižního ocenění
Microsoftu MVP v České republice

500

Další informace ke cvičení:

<https://office.lasakovi.com/excel/Query-editor/M-code-M-Language-zaklady-Power>





Co to je

Je to součást PowerQuery editoru
Jde o funkční jazyk pro vytváření

Jak jej mohu vidět?

V PowerQuery editoru musíte s
Případně si můžete zobrazit rozšíření
V základu je M kód o hodnotách

Základní úpravy

M kód v libovolném kroku (sekcii)

Případně přes Fx můžete vkládat

Pozor při práci jazyk respektuje
více o referenci u microsoftu
<https://docs.microsoft.com/cs->

Případně k dostupným knihovnam

hodnotách (Values) a dotazech

Výraz je něco, co se dá vyhodnotit

Hodnota je kus datové informace

Hodnoty mohou být

- čísla
- text
- logická hodnota
- binární
- datu
- čas
- ...

Hodnoty jsou ale i:

- seznamy
- záznam v tabulce (řádky)
- seznamy seznamů a
- seznamy tabulek
- tabulka

Poznámky a komentáře

```
// komentář  
*/ this is my  
comment */
```

Pojďme se na hodnoty podívat

Samostatné hodnoty (jednotlivé)

```
Samostatné hodnoty  
125.44  
"JakNaExcel"  
True  
null
```

Samostatné hodnoty vytvořené

```
Půjde hlavně o časové  
#time (hodiny, minuty, sekundy)  
#date (roky, měsíce, dny)  
#datetime (roky, měsíce, dny, hodiny, minuty, sekundy)  
#datetimezone (roky, měsíce, dny, hodiny, minuty, sekundy, časová zóna)  
#duration (dny, hodiny, minuty, sekundy)  
Například k vytvoření
```

Strukturovaná data

```
seznamy
```

= {1, 2}

{3, 4, 5}

{{1, 2}, {3, 4, 5}}

sekvenční seznamy

= {"a".."d"}

Seznam seznamů

{{1, 2}, {3, 4, 5}}

Neplést se zázname

Záznam Rekord

[FirstName = "John"

Tabulky

#table({"Letters", "N

Dotazy (výrazy) Expressior

Dotaz (Výrazy, vzore

Výraz 1 + 1 se vyho

Výraz 3 > 2 se vyhoc

Výraz „Hello“ a „Wo

Výraz Text.Upper („l

pojdme se podívat do v dotazer

Aritmetické operát

M kód umí klasické
#date (2018,12,25)

Porovnání - porovna

Hodnoty v M kódu r

x < y se vyhodnotí ja

x > y se vyhodnotí ja

x <= y se vyhodnotí j

x >= y se vyhodnotí j

x = y se vyhodnotí ja

x <> y bude vyhodnc

{1,2,3,4} = {1,2,3}

Zřetězení (sloučení)

Můžete zřetězit text

„Ahoj“ & „Svět“ se v

{1,2,3} & {3,4,5} se v

Logický

Můžete provádět op

Komentáře

```
//This is a sing:  
M code goes here  
on multiple line:
```

Len

Nechte prohlášení

Příkaz let umožňuje,

```
let  
  a = 1,  
  b = 2,  
  c = a + b  
in  
  c
```

podívám se podrobně

Variable Names

```
let  
    #"Is this a variable"  
in  
    #"Is this a variable"
```

Each Statements

Každá prohlášení
Každý výraz je zkratkou

```
let  
    Source = #tal  
    #"Added Custo  
in  
    #"Added Custo
```

```
let  
    Source =  
    #"Added C  
in  
    #"Added C
```

If Then Else Statement

Pokud jde o logické

if [logický výraz k testu]
 pak [udělej, když je pravda]
else [to když falešný]
Syntaxe je přímá a je

Funkce

Funkce je mapování

nechat

Produkt = (x, y) =:

Výsledek = Produ

v

Výsledek

Opět v samostném č

Jiné ukázky kódu

Načtení dat

= Excel.CurrentWork

Filtrace

= Table.SelectRows(

Změna typu

= Table.TransformCr

Přejmenování

= Table.RenameColu

Přidání sloupce

= Table.AddColumn

Odebrání sloupce

= Table.RemoveColu

Další ?

Napadá vás co do tohoto přehle

M-Language , M-kód

<https://office.lasakovi.com/>

ru. Někdy označován jako M-code, M-language, M-kód, M-jazyk.

ní dotazu na práci s tabulkami. V základu si jej píše Excel v PowerQuery sám na základě vašich čiností,

spustit a zobrazit řádek vzorců

šířený editor. Co který příkaz znamená se pokusím rozebrat dále

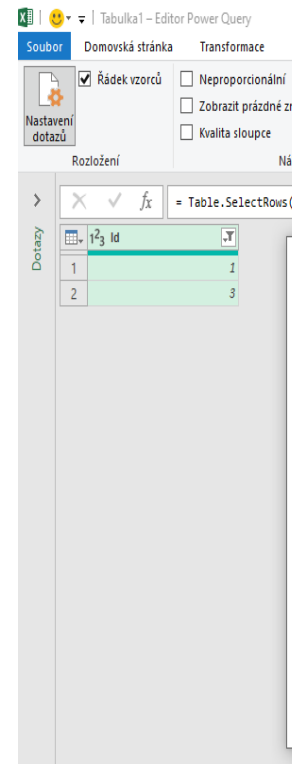
h (Values) a dotazech/výrazech (Expressions) o čemž se rozepíšu dále

Popis okna

Jméno tabulky

Vlastní kód

Případné chyby



) z dotazu, upravíte kliknutím do vzorce a přidáním, vylepšením. Potvrzení změny přes ikonu 'OK'.

at funkce (o tom v samostatném článku)

: malá a velká písmena "JakNaExcel" je něco jiného než "Jaknaexcel"

[cz/powerquery-m/index](https://office.lasakovi.com/cz/powerquery-m/index)

ám se dostanete přes slovo #shared



1 (Expressions)



otit. Nebo-li pokud vyhodnotím mohu vrátit zpět nějakou "hodnotu". Z dotazu $1 + 1$ mohu tento výra:
ce nebo-li

ádek)
td.

Jednořádkový
více řádkový

dívat podrobněji

ré)

y jsou základním stavebním kamenem pro všech ostatních hodnoty (např. záznam se skládá ze smamc
číslo
text
logická hodnota
skutečně prázdná hodnota, i designově graficky se liší v tabulkách

é funkcí

vé funkce
ity, sekundy)
, dny)
ésíce, dny, hodiny, minuty, sekundy)
y, měsíce, dny, hodiny, minuty, sekundy, offset-hodiny, offset-minuty)
iny, minuty, sekundy)
ní data 2018-12-31 je nutné jej zkonstruovat pomocí vnitřní funkce #date (2018, 12, 31).

Sekvenční seznamy můžete vytvářet ve formátu {x..y}. {2..5} vytvoří seznam {2, 3, 4, 5}. Toto funguje i

Jsou také možné seznamy seznamů, takže {{1, 2}, {3, 4, 5}} je seznam dvou seznamů. První seznam ob PowerQuery zobrazí List a prokliknutím se teprve zobrazí hodnoty daném seznamu...

m (Records)

```
[, Age = 38]
```

Záznam je uspořádaná sekvence polí. Každé pole se skládá z názvu pole, které jedinečně identifikuje pole. Záznam je uspořádaná sekvence polí. Každé pole se skládá z názvu pole, které jedinečně identifikuje pole. Záznam můžete definovat pomocí hranatých závorek. [FirstName = "John", Age = 38] je záznam se dvěma poli. Záznamy záznamů jsou také možné, [Osoba = [Jméno = "Jan", Věk = 38]] je záznam s jedním polem s záznamem. Jsou také možné prázdné záznamy, [] je prázdný záznam.

Můžete odkazovat na hodnotu pole v záznamu podle názvu pole. [Jméno = "Jan", Věk = 38] [Jméno] s

```
umbers"}, {"A", 1}, {"B", 2}, {"C", 3})
```

Tabulka je uspořádaná posloupnost řádků, kde každý řádek je seznam.

Tabulky lze sestavit pouze pomocí vlastní funkce. Tabulku můžete sestavit pomocí funkce #table () ze

```
#table ({"Dopisy", "Čísla"}, {"A", 1}, {"B", 2}, {"C", 3})
```

 vytvoří tabulku se 2 sloupci, 3 řádky a záhlaví sl

Je možné vytvořit prázdnou tabulku pomocí prázdných seznamů ve vnitřní funkci #table (). #table ({}),

Na libovolnou hodnotu v tabulce můžete odkazovat s indexem řádků na základě nuly a názvem záhlaví

1s podrobněji

ec) je cokoli, co lze vyhodnotit a vrátit hodnotu. Ono i hodnota je vlastně dotaz ;) Hodnota 1 vyhodnotí .
dnocuje jako 2.

lnotí jako pravdivý.

lrd" se hodnotí jako „Hello World“.

Hello World“) se vyhodnotí jako „HELLO WORLD“.

ch výrazech funkcích lze používat

ory

aritmickými operátory +, -, * a / (což je logické ale neškodí zmínit). Umožní přidávat, odečítat, násobit a dělit. #duration (7, 0, 0, 0) se vyhodnotí na 2019-01-01.

řivací operátory

můžete porovnat pomocí operátorů porovnání <,>, <=,>=, =, <=.

ko true, pokud x je menší než y.

ko true, pokud x je větší než y.

jako true, pokud x je menší nebo rovno y.

jako true, pokud x je větší nebo rovno y.

ko true, pokud x se rovná y.

oceno jako true, pokud x není rovno y.

|

rové a sloučené seznamy, záznamy a tabulky pomocí ampersand & operator.

vyhodnotí jako „Ahoj svět“.

vyhodnotí na {1,2,3,3,4,5}.

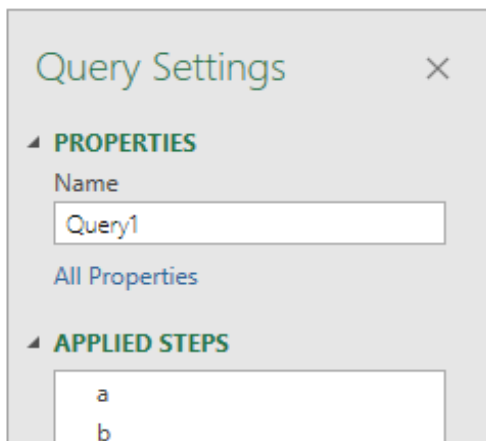
erace s booleovskými hodnotami (nebo výrazy, které se vyhodnocují v booleovských hodnotách) s op

```
le line comment
```

```
/*This is a comment
```

```
s*/ M code goes here
```

, aby byla sada hodnot vyhodnocena a přiřazena názvům proměnných a poté použita v následujícím v





článek je v samostatném článku jen to mám

Variable Names

Is this a variable name? Wow!!! = 1 + 1

Is this a variable name? Wow!!!

```
let
  #"Is this a variable name? Wow!!!" = 1 + 1
in
  #"Is this a variable name? Wow!!!"
```

podtržítka) pro deklarování funkcí, které berou jeden parametr s názvem _ (podtržítka).

```
Table.FromSource(
  Source, Table.AddColumn(Source, "Double", each 2*[Numbers])
) as Custom"
```

```
#table({"Numbers"}, {{1}, {2}, {3}, {4}, {5}}),
Custom" = Table.AddColumn(Source, "Double", each 2*[Numbers])
Custom"
```

výrazy, kód M je ve srovnání s jinými jazyky dost řídký. Nejsou k dispozici žádné příkazy case nebo loop

stování]

to pravda]

]

je jako většina ostatních programovacích

```
if [logical expression to test]
then [do this when true]
else [do this when false]
```

ový

ze sady hodnot parametrů na hodnotu. Spolu s funkcemi ve standardní knihovně vám M kód umožňu

```
> x * y,  
|kt (2,3)
```

šlánki

```
let  
  Product = (x, optional y) => if y is null then x else x * y,  
  Result = Product(2)  
in  
  Result
```

```
book(){[Name="Tabulka1"]}[Content]
```

```
Zdroj, each ([Id] <> 2))
```

```
columnTypes("#Filtrované řádky",{{"Id", Int64.Type}})
```

```
μms("#Změněný typ",{{"Id", "Pokus"}})
```

```
(#"Přejmenované sloupce", "Součet", each [Pokus] + 1, type number)
```

```
μms("#Vložené: Součet",{{"Součet"}})
```

edového článku doplnit prosím o informaci



ale proč se na tento jazyk nepodívat hlouběji?

```
let  
Zdroj = Excel.CurrentWorkbook(){[Name="Tabulka1"]}[Content],  
#"Změněný typ" = Table.TransformColumnTypes(Zdroj,{{"Id", Int64.Type}}),  
#"Filtrované řádky" = Table.SelectRows("#Změněný typ", each ([Id] <> 2))  
in  
#"Filtrované řádky"
```

```
"try <expr  
= try "foo"-  
result =  
try error
```

neny můžete zanořit klávesnicí na X nebo stisknutím ESC.

Function Name	Category
Tabulka1	Table
Tabulka1 (2)	Record
Value.ResourceExpression	Function
Resource.Access	Function
Kusto.Contents	Function
Kusto.Databases	Function
DataLake.Contents	Function
DataLake.Files	Function
List.NonNullCount	Function
List.MatchesAll	Function
List.MatchesAny	Function
List.Range	Function

	List.RemoveItems	Function
	List.ReplaceValue	Function

z vyhodnotí a získat hodnotu 2.

ostatných hodnot viz dále) .

také pro textové znaky. {"A" .. "d"} vytvoří seznam {"a", "b", "c", "d"}.

obsahuje čísla 1 a 2 a druhý seznam obsahuje čísla 3, 4 a 5.

pole a hodnoty pole, která může být libovolným typem hodnoty.

pole a hodnoty pole, která může být libovolným typem hodnoty.

dvěma poli. První pole v záznamu má název pole Jméno a hodnotu „Jan“. Druhé pole v záznamu má název pole Osoba a hodnotou pole, která je záznamem.

se vyhodnotí jako „Jan“.

seznamu záhlaví sloupců a seznamu řádků.

sloupců Písmena a čísla.

{}) vytvoří prázdnou tabulku.

ví sloupce. #table ({"Dopisy", "Čísla"}, {"A", 1}, {"B", 2}, {"C", 3}) {2} [Dopisy] se vyhodnotí jako „C“, Pi

a vrátí 1.

obit a dělit hodnoty.

operátory not, a nebo.

výrazu, který následuje za příkazem in.

p. K dispozici je pouze výraz if... then... else....

t pro snadnější čtení zobrazen na samostatných řádcích.

je vytvářet vlastní funkce.



essions to try> otherwise <default if error>

+1 otherwise "There was an error"

```
try "foo"+1, if result[HasError] then "There was an error" else result[Value]
```

```
· "negative unit count" otherwise 42
```


ev pole Věk a hodnotu 38.

rotože se jedná o 2. řádek (na základě nulového indexu) sloupce Dopisy.

Strukturované data

Klasický seznam

= {1, 2}

Vícerozměrné seznam

= {{1, 2}, {3, 4, 5}}

Sekvenční seznamy

= {"a".."f"}

Record

= [Jméno = "Pavel", Funkce = "MVP", Město="Brno"]

= [Osoba = [Jméno = "Jan", Věk = 38]]

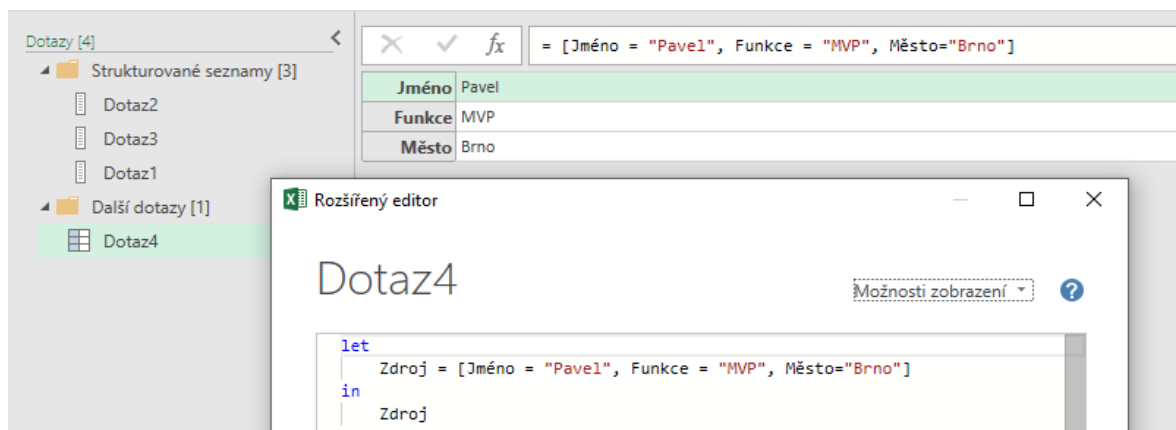
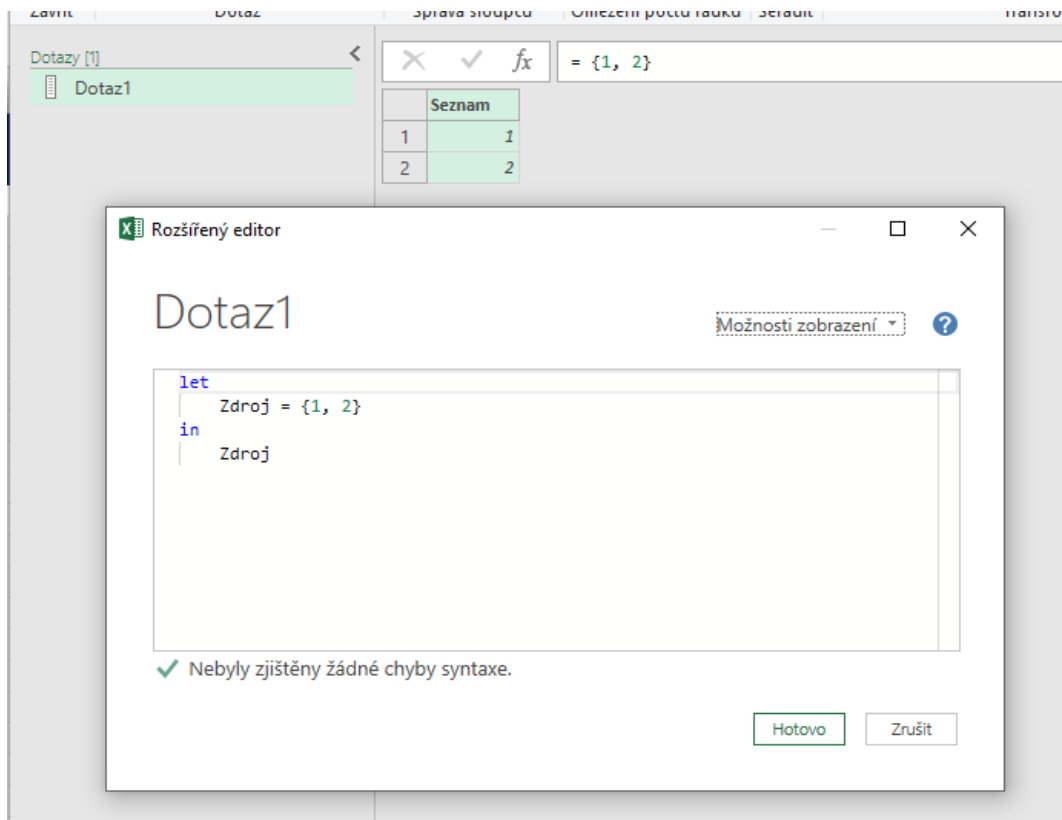
Tabulky

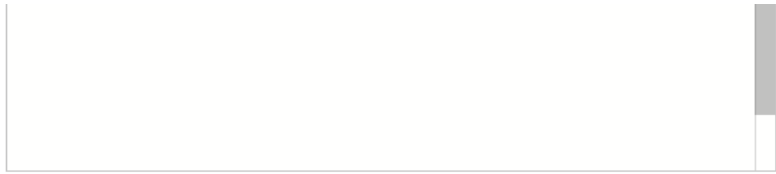
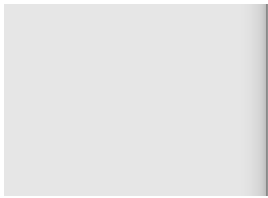
```
= #table({"Písmena", "Čísla"}, {"A", 1}, {"B", 2}, {"C
```

```
// ze seznamu  
= Table.FromList({"a", "b", "c", "d"}, null,  
{"Písmena"})  
// Ze záznamu (Record)  
= Table.FromRecords({[ID = 1, Jmeno = "Pavel", Tit =  
"MVP"], [ID = 2, Jmeno = "Eva", Tit = "Ing"], [ID = 3,  
Jmeno = "Iva", Tit = null]})
```

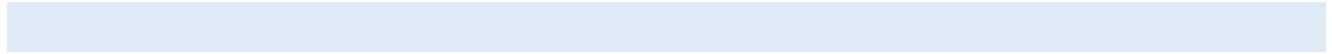
Language, M-kód

<https://office.lasakovi.com/>





✓ Nebyly zjištěny žádné chyby syntaxe.



:", 3}})



Tabulky

```
= #table({"Písmena", "Čísla"}, {"A", 1}, {"B", 2}, {"C
```

```
// ze seznamu  
= Table.FromList({"a", "b", "c", "d"}, null,  
{"Písmena"})  
// Ze záznamu (Record)  
= Table.FromRecords({[ID = 1, Jmeno = "Pavel", Tit =  
"MVP"], [ID = 2, Jmeno = "Eva", Tit = "Ing"], [ID = 3,  
Jmeno = "Iva", Tit = null]})
```


Language , M-kód

<https://office.lasakovi.com/>

:", 3}})

