

# **MSE6 – Analýza dat za pomoci Power Pivot, Power Query a Power Map**

## POWER QUERY

### Datové typy

#### Primitivní typy

Typy v jazyce M tvoří nesouvislou hierarchii vycházející z typu any. Tento typ označuje všechny hodnoty. Jakákoli hodnota jazyka M odpovídá přesně jednomu primitivnímu podtypu any. Tady jsou uzavřené sady primitivních typů odvozených z typu any:

- type null označuje hodnotu null
- type logical označuje hodnoty true a false
- type number označuje číselné hodnoty
- type time označuje časové hodnoty
- type date označuje kalendářní hodnoty
- type datetime označuje hodnoty data a času
- type datetimezone označuje hodnoty datetimezone
- type duration označuje hodnoty doby trvání
- type text označuje textové hodnoty
- type binary označuje binární hodnoty
- type označuje hodnoty typu
- type list označuje hodnoty seznamu
- type record označuje hodnoty záznamu
- type table označuje hodnoty tabulky
- type function označuje hodnoty funkce
- type anynonnull označuje všechny hodnoty s výjimkou hodnoty null, vnitřní typ none neoznačuje žádné hodnoty

#### Další typy

- type list označuje skupinu hodnot libovolného typu
- type record označuje skupinu klíčů a jejich hodnot
- type table označuje skupinu hodnot rozdělených do řádků a sloupců, přičemž každý sloupec má unikátní jméno a je mu přiřazen datový typ pro všechny hodnoty

## Operátory

Výraz	Popis
<b>i</b>	Výraz identifikátoru
<b>@i</b>	Výraz identifikátoru
<b>(x)</b>	Výraz v závorkách
<b>x[i]</b>	Vyhledávání
<b>x{y}</b>	Přístup k položce
<b>x(...)</b>	Vyvolání funkce
<b>{x, y,</b>	Inicializace seznamu
<b>...}</b>	
<b>[ i = x,</b>	Inicializace záznamu
<b>... ]</b>	
<b>...</b>	Není implementováno
<b>+x</b>	Identita
<b>-x</b>	Opačné číslo
<b>not x</b>	Logická negace
<b>x meta</b>	Přidružení metadat
<b>y</b>	
<b>x * y</b>	Násobení
<b>x / y</b>	Dělení
<b>x + y</b>	Sčítání
<b>x - y</b>	Odčítání
<b>x &lt; y</b>	Menší než
<b>x &gt; y</b>	Větší než
<b>x &lt;= y</b>	Menší než nebo rovno
<b>x &gt;= y</b>	Větší než nebo rovno
<b>x = y</b>	Je rovno
<b>x &lt;&gt; y</b>	Není rovno
<b>x as y</b>	Je kompatibilní primitivní typ s možnou hodnotou null, nebo chyba
<b>x is y</b>	Test, pokud je kompatibilní primitivní typ s možnou hodnotou null
<b>x and y</b>	Zkrácená konjunkce
<b>x or y</b>	Zkrácená disjunkce

## Základní operace s tabulkami

### Filtrování

Ponechání či naopak odstranění jednoho, nebo více řádků, které splňují / nesplňují podmínu (logika je stejná jako v MS Excel). Filtr je dostupný přes tlačítko filtru vedle názvu sloupce.

Zeme	Rok	Pocet obyvatel	Kod meny	Clenksa zeme EU
1 Czechia	2017	10578		
2 Czechia	2018	10610		
3 Czechia	2019	10649		
4 Slovakia	2017	5435		
5 Slovakia	2018	5443		
6 Slovakia	2019	5450		
7 Poland	2017	37972		
8 Poland	2018	37976		
9 Poland	2019	37972		
10 Switzerland	2017	8149		
11 Switzerland	2018	8480		
12 Switzerland	2019	8544		

### Seskupování hodnot

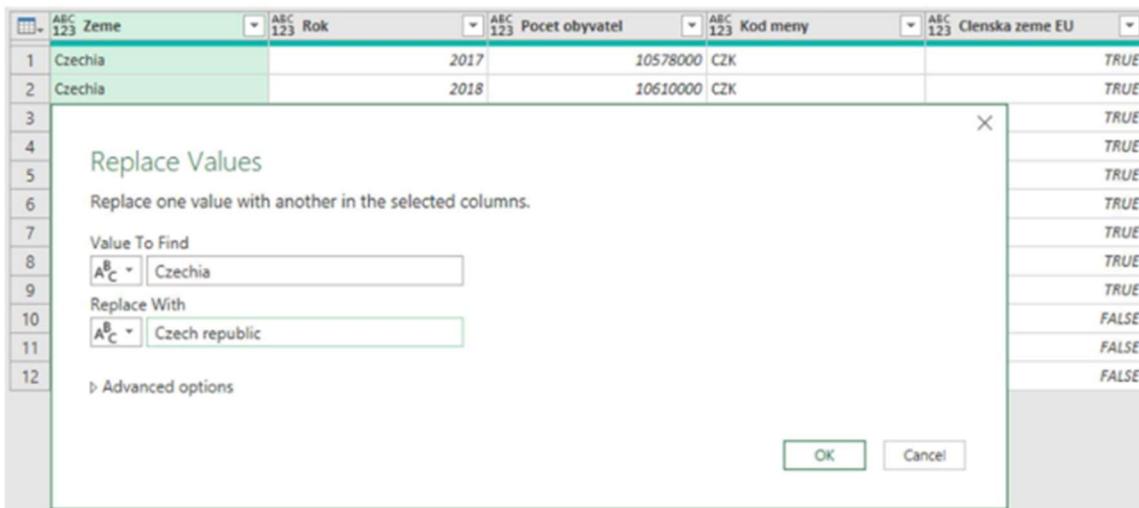
Agregace zvoleného sloupce. Funkce je dostupná přes tlačítko Seskupit podle v kartě Domů hlavní navigace.

Zeme	Rok	Pocet obyvatel	Kod meny	Clenksa zeme EU
1 Czechia	2017	10578000	CZK	TRUE
2 Czechia	2018	10610000	CZK	TRUE
3 Czechia	2019	10649000	CZK	TRUE

Group By  
Specify the column to group by and the desired output.  
Basic Advanced  
Zeme  
New column name: Prumerny pocet obyvatel  
Operation: Average  
Column: Pocet obyvatel

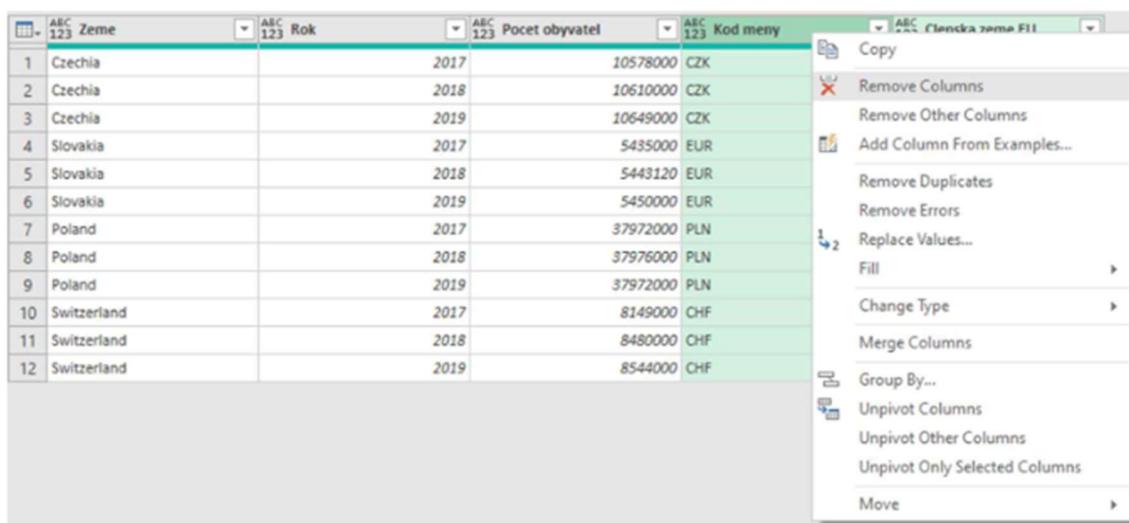
## Nahrazení hodnot

Nahrazení staré hodnoty za novou, funkce je dostupná přes tlačítko Nahradit hodnoty v kartě Domů hlavní navigace



## Manipulace se sloupcí

Ponechání či odstranění vybraných sloupců. Funkce je dostupná po kliknutí pravým tlačítkem na název sloupce.



## Přidání sloupců

### Sloupec z příkladu

Na základě uživatelem definovaného vstupu, Power Query se snaží najít vzorec, který by mohl vést k vygenerování uživatelem zadané hodnoty. Funkce podobná automatickému doplňování v MS Excel, uživateli je však následně k náhledu postup zvolený Power Query v jazyku M. Funkce je dostupná přes tlačítko Podmíněný sloupec karty Přidání sloupce hlavní navigace.

Zeme	Rok	Pocet obyvatel	Kod meny	Clesnska zeme EU	Datum
Czechia	2017	10578000 CZK			1.1.2017
Czechia	2018	10610000 CZK			1.1.2018
Czechia	2019	106490000 CZK			1.1.2019
Slovakia	2017	54350000 EUR			1.1.2017
Slovakia	2018	54443120 EUR			1.1.2018
Slovakia	2019	54500000 EUR			1.1.2019
Poland	2017	37972000 PLN			1.1.2017
Poland	2018	37976000 PLN			1.1.2018
Poland	2019	37972000 PLN			1.1.2019
Switzerland	2017	81490000 CHF			1.1.2017
Switzerland	2018	84800000 CHF			1.1.2018
Switzerland	2019	85440000 CHF			1.1.2019

### Vlastní sloupec

Hodnota ve sloupci je definována funkcí napsanou v jazyku M. Funkce je dostupná přes tlačítko Vlastní sloupec karty Přidání sloupce v hlavní navigaci.

## Podmíněný sloupec

Zjednodušená varianta pro přidání sloupce, jehož hodnota je návratovou hodnotou podmíněného výrazu If Else. Funkce je dostupná přes tlačítko Podmíněný sloupec karty Přidání sloupce v hlavní navigaci.

	ABC 123 Zeme	ABC 123 Rok	ABC 123 Poct obyvatel	ABC 123 Kod meny	ABC 123 Clnska zeme EU		
1	Czechia	2017	10578000 CZK		TRUE		
2	Czechia	2018	10610000 CZK		TRUE		
3							
4	Add Conditional Column					X	
5	Add a conditional column that is computed from the other columns or values.						
6							
7	New column name						
8	Clenska zeme eurozony						
9	Column Name	Operator	Value ⓘ	Output ⓘ			
10	If	Kod meny	equals	ABC 123 EUR	Then ABC 123 true	...	
11	Add Clause						
12	Else ⓘ	ABC 123 false					
						OK	Cancel

## Spojování tabulek

### Levá tabulka

	Zeme	Rok	Pocet obyvatel	Kod meny	Cleńska zeme EU
1	Czechia	2017	10578000	CZK	TRUE
2	Czechia	2018	10610000	CZK	TRUE
3	Czechia	2019	10649000	CZK	TRUE
4	Slovakia	2017	5435000	EUR	TRUE
5	Slovakia	2018	5443120	EUR	TRUE
6	Slovakia	2019	5450000	EUR	TRUE
7	Poland	2017	37972000	PLN	TRUE
8	Poland	2018	37976000	PLN	TRUE
9	Poland	2019	37972000	PLN	TRUE
10	Switzerland	2017	8149000	CHF	FALSE
11	Switzerland	2018	8480000	CHF	FALSE
12	Switzerland	2019	8544000	CHF	FALSE

### Pravá tabulka

	Zeme	Rok	Prumerny vek
1	Czechia	2017	42,1
2	Czechia	2018	42,1
3	Czechia	2019	42,1
4	Slovakia	2017	39,2
5	Slovakia	2018	39,2
6	Slovakia	2019	39,2
7	Poland	2017	41,1
8	Poland	2018	41,1
9	Poland	2019	41,1
10	Germany	2017	45,7
11	Germany	2018	45,7
12	Germany	2019	45,7

### Left join – spojení zleva

	Zeme	Rok	Pocet obyvatel	Kod meny	Cleńska zeme EU	Prumerny vek
1	Czechia	2017	10578000	CZK	TRUE	42,1
2	Czechia	2018	10610000	CZK	TRUE	42,1
3	Czechia	2019	10649000	CZK	TRUE	42,1
4	Slovakia	2017	5435000	EUR	TRUE	39,2
5	Slovakia	2018	5443120	EUR	TRUE	39,2
6	Slovakia	2019	5450000	EUR	TRUE	39,2
7	Poland	2017	37972000	PLN	TRUE	41,1
8	Poland	2018	37976000	PLN	TRUE	41,1
9	Poland	2019	37972000	PLN	TRUE	41,1
10	Switzerland	2017	8149000	CHF	FALSE	null
11	Switzerland	2018	8480000	CHF	FALSE	null
12	Switzerland	2019	8544000	CHF	FALSE	null

## Right join – spojení zprava

	Zeme	Rok	Pocet obyvatel	Kod meny	Clenksa zeme EU	Prumerny vek
1	Czechia	2017	10578000	CZK	TRUE	42,1
2	Czechia	2018	10610000	CZK	TRUE	42,1
3	Czechia	2019	10649000	CZK	TRUE	42,1
4	Slovakia	2017	5435000	EUR	TRUE	39,2
5	Slovakia	2018	5443120	EUR	TRUE	39,2
6	Slovakia	2019	5450000	EUR	TRUE	39,2
7	Poland	2017	37972000	PLN	TRUE	41,1
8	Poland	2018	37976000	PLN	TRUE	41,1
9	Poland	2019	37972000	PLN	TRUE	41,1
10	null	null	null	null	null	45,7
11	null	null	null	null	null	45,7
12	null	null	null	null	null	45,7

## Outer join – vše z obou

	Zeme	Rok	Pocet obyvatel	Kod meny	Clenksa zeme EU	Prumerny vek
1	Czechia	2017	10578000	CZK	TRUE	42,1
2	Czechia	2018	10610000	CZK	TRUE	42,1
3	Czechia	2019	10649000	CZK	TRUE	42,1
4	Slovakia	2017	5435000	EUR	TRUE	39,2
5	Slovakia	2018	5443120	EUR	TRUE	39,2
6	Slovakia	2019	5450000	EUR	TRUE	39,2
7	Poland	2017	37972000	PLN	TRUE	41,1
8	Poland	2018	37976000	PLN	TRUE	41,1
9	Poland	2019	37972000	PLN	TRUE	41,1
10	null	null	null	null	null	45,7
11	null	null	null	null	null	45,7
12	null	null	null	null	null	45,7
13	Switzerland	2017	8149000	CHF	FALSE	null
14	Switzerland	2018	8480000	CHF	FALSE	null
15	Switzerland	2019	8544000	CHF	FALSE	null

## Inner join – pouze shodné z obou tabulek

	Zeme	Rok	Pocet obyvatel	Kod meny	Clenksa zeme EU	Prumerny vek
1	Czechia	2017	10578000	CZK	TRUE	42,1
2	Czechia	2018	10610000	CZK	TRUE	42,1
3	Czechia	2019	10649000	CZK	TRUE	42,1
4	Slovakia	2017	5435000	EUR	TRUE	39,2
5	Slovakia	2018	5443120	EUR	TRUE	39,2
6	Slovakia	2019	5450000	EUR	TRUE	39,2
7	Poland	2017	37972000	PLN	TRUE	41,1
8	Poland	2018	37976000	PLN	TRUE	41,1
9	Poland	2019	37972000	PLN	TRUE	41,1

## Připojování tabulek

### Horní tabulka

	Zeme	Rok	Pocet obyvatel	Kod meny	Clenksa zeme EU
1	Czechia	2017	10578000	CZK	TRUE
2	Czechia	2018	10610000	CZK	TRUE
3	Czechia	2019	10649000	CZK	TRUE
4	Slovakia	2017	5435000	EUR	TRUE
5	Slovakia	2018	5443120	EUR	TRUE
6	Slovakia	2019	5450000	EUR	TRUE
7	Poland	2017	37972000	PLN	TRUE
8	Poland	2018	37976000	PLN	TRUE
9	Poland	2019	37972000	PLN	TRUE
10	Switzerland	2017	8149000	CHF	FALSE
11	Switzerland	2018	8480000	CHF	FALSE
12	Switzerland	2019	8544000	CHF	FALSE

### Spodní tabulka

	Zeme	Rok	Pocet obyvatel	Kod meny	Clenksa zeme EU
1	Czechia	2015	10538000	CZK	TRUE
2	Czechia	2016	10553000	CZK	TRUE
3	Slovakia	2015	5421000	EUR	TRUE
4	Slovakia	2016	5426000	EUR	TRUE
5	Poland	2015	38005000	PLN	TRUE
6	Poland	2016	37967000	PLN	TRUE

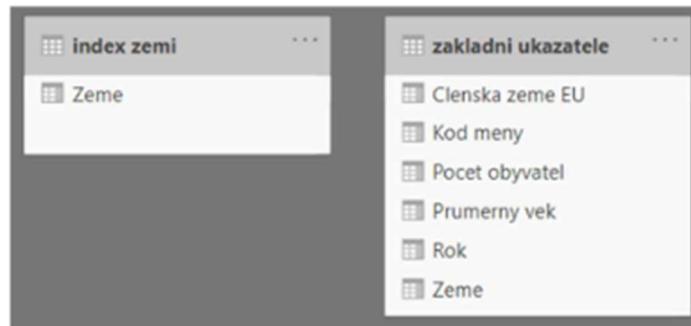
### Připojení

	Zeme	Rok	Pocet obyvatel	Kod meny	Clenksa zeme EU
1	Czechia	2017	10578000	CZK	TRUE
2	Czechia	2018	10610000	CZK	TRUE
3	Czechia	2019	10649000	CZK	TRUE
4	Slovakia	2017	5435000	EUR	TRUE
5	Slovakia	2018	5443120	EUR	TRUE
6	Slovakia	2019	5450000	EUR	TRUE
7	Poland	2017	37972000	PLN	TRUE
8	Poland	2018	37976000	PLN	TRUE
9	Poland	2019	37972000	PLN	TRUE
10	Switzerland	2017	8149000	CHF	FALSE
11	Switzerland	2018	8480000	CHF	FALSE
12	Switzerland	2019	8544000	CHF	FALSE
13	Czechia	2015	10538000	CZK	TRUE
14	Czechia	2016	10553000	CZK	TRUE
15	Slovakia	2015	5421000	EUR	TRUE
16	Slovakia	2016	5426000	EUR	TRUE
17	Poland	2015	38005000	PLN	TRUE
18	Poland	2016	37967000	PLN	TRUE

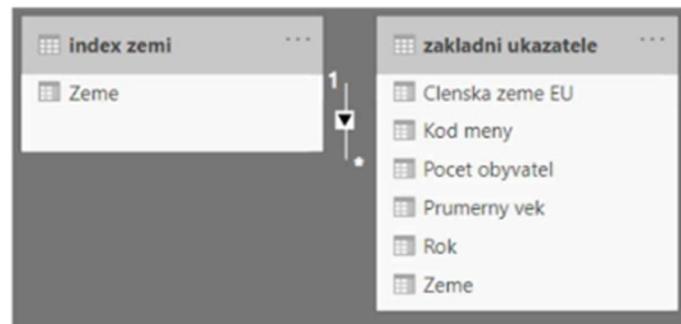
## POWER PIVOT

### Relace

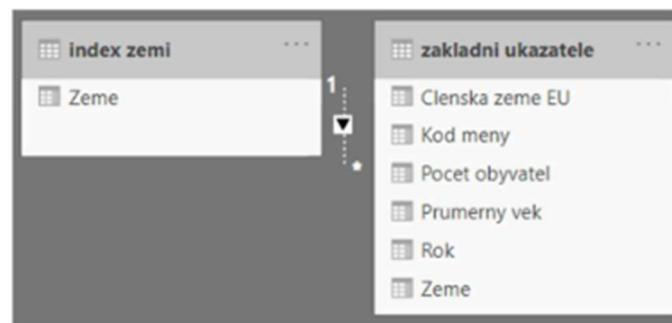
Nepropojené tabulky, tj.  
vzájemně se neovlivňující  
tabulky.



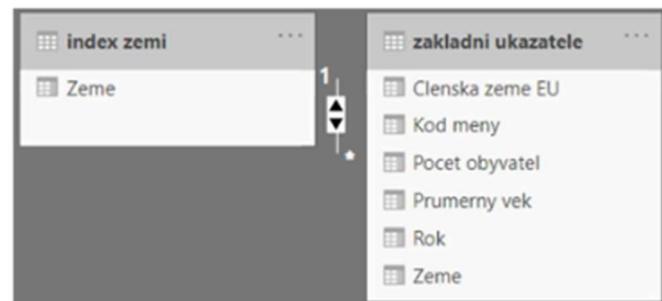
Tabulky propojené  
jednosměrnou relací, kdy  
tabulka 'index zemi' ovlivňuje  
tabulku 'zakladni ukazatele'.



Tabulky propojené  
jednosměrnou neaktivní relací,  
takové tabulky se neovlivňují,  
relaci lze aktivovat funkcí  
USERELATIONSHIP.



Tabulky propojené obousměrnou  
relací, tj. tabulky se ovlivňují  
navzájem.



Jakýkoliv filtr v tabulce je automaticky aplikován na všechny tabulky propojené aktivní relací. Tuto funkcionality lze přirovnat k situaci, kdy je ze všech propojených tabulek vytvořena „rozšířená tabulka“, jejíž filtry se chovají stejně jako v MS Excel.

Velký počet relací, zejména těch obousměrných má negativní dopad na rychlosť výpočtu, proto je vhodné do Power Pivot nahrávat datový model řádně upravený v Power Query.

Zároveň je dobré zvážit definování, úpravu či aktivaci relace skrze funkce jazyka DAX.

### Příklad pomyslné rozšířené tabulky

Zeme	Zeme	Rok	Pocet obyvatel	Kod meny	Clenksa zeme EU	Prumerny vek
Czechia	Czechia	2017	10578000	CZK	TRUE	42,1
Slovakia	Czechia	2018	10610000	CZK	TRUE	42,1
Poland	Czechia	2019	10649000	CZK	TRUE	42,1
Switzerland	Slovakia	2017	5435000	EUR	TRUE	39,2
	Slovakia	2018	5443120	EUR	TRUE	39,2
	Slovakia	2019	5450000	EUR	TRUE	39,2
	Poland	2017	37972000	PLN	TRUE	41,1
	Poland	2018	37976000	PLN	TRUE	41,1
	Poland	2019	37972000	PLN	TRUE	41,1
	Switzerland	2017	8149000	CHF	FALSE	
	Switzerland	2018	8480000	CHF	FALSE	
	Switzerland	2019	8544000	CHF	FALSE	
	Czechia	2015	10538000	CZK	TRUE	
	Czechia	2016	10553000	CZK	TRUE	

## Počítané míry a sloupce

### Počítané sloupce

Počítají hodnotu pro každý řádek tabulky - výsledek je uložen v paměti - k přepočtu dat dochází po aktualizaci reportu - pracují s kontextem řádku.

Jsou vhodné v případě, kdy:

- výslednou hodnotu potřebujeme pro každý řádek tabulky
- plánujeme segmentaci na základě výsledné hodnoty kalkulovaného sloupce

### Počítané míry

- Nepočítají hodnotu pro každý řádek tabulky
- Pracují s kontextem filtru
- K přepočtu dochází vždy po změně kontextu filtru

Jsou vhodné v případě, kdy:

- výslednou hodnotu nepotřebujeme pro každý řádek tabulky
- neplánujeme segmentaci na základě výsledné hodnoty
- potřebujeme reagovat na změny v kontextu filtru

## Kontext výpočtu Power Pivot

Vzorce v Power Pivot mohou být ovlivněny filtry použitými v kontingenční tabulce, relacemi mezi tabulkami a filtry použitými ve vzorcích. Kontext je to, co umožňuje dynamickou analýzu. Princip vytváření a řešení problémů se vzorcí je důležitý.

Existují různé typy kontextu: kontext řádku, kontext dotazu a kontext filtru.

Kontext řádku se dá představit jako "aktuální řádek". Pokud jste vytvořili počítaný sloupec, obsahuje kontext řádku hodnoty v jednotlivých řádcích a hodnotách ve sloupcích, které souvisejí s aktuálním řádkem. K dispozici jsou také některé funkce (starší a nejstarší), které získávají hodnotu z aktuálního řádku, a pak tuto hodnotu použijte při provádění operací na celé tabulce.

Kontext dotazu odkazuje na podmnožinu dat, která se implicitně vytvoří pro každou buňku v kontingenční tabulce v závislosti na záhlaví řádků a sloupců.

Kontext filtru je sada hodnot povolených v jednotlivých sloupcích na základě omezení filtru, která byla použita na řádek nebo jsou definovány výrazy filtru ve vzorci.

## DAX

DAX znamená Data Analysis Expressions a jde o jazyk vzorců používaný v rámci Power Pivot.

### Základní funkce jazyka DAX

#### Agregační funkce

SUM - součet

AVERAGE - průměr

MIN - minimum

MAX - maximum

SUMX – podmíněný součet, ekvivalent SUMIF v MS Excel

#### Počítané funkce

COUNT - počet

COUNTA – počet neprázdných hodnot

COUNTBLANKS – počet prázdných hodnot

COUNTROWS – počet řádků tabulky

DISTINCTCOUNT – počet unikátních hodnot

#### Logické funkce

AND – vrací TRUE v případě, že jsou splněny všechny logické podmínky, v opačném případu

vrací FALSE

OR – vrací TRUE v případě, že je splněna alespoň jedna logická podmínka, v opačném případu

vrací FALSE

NOT – mění hodnotu FALSE na TRUE a TRUE na FALSE

IF – kontroluje, jestli je splněna podmínka zadaná jako první argument, vrací hodnotu TRUE při splnění podmínky a FALSE při nesplnění podmínky

IFERROR – vyhodnotí výraz a vrátí zadanou hodnotu, pokud je hodnotou chyba, vrátí alternativní hodnotu

#### Datové funkce

DATE – vrátí zadané datum ve formátu datetime, očekává tyto argumenty: rok, měsíc, den

HOUR – extrahuje hodinu ze sloupce datového formátu datetime

NOW – vrací aktuální časovou značku ve formátu datetime

EOMONTH – vrací poslední den v měsíci, očekává argumenty: datum, počet měsíců

**Funkce časového měřítka**

DATEADD – vrací tabulku se sloupcem kalendářních dat, která jsou posunuta v čase o zadaný počet intervalů, očekává argumenty: datum, počet intervalu, interval)

DATESBETWEEN – vrací tabulku obsahující sloupec kalendářních dat začínající hodnotou počátečního data a pokračující ke konečnému datu, očekává argumenty: datum, počáteční datum, konečné datum)

FIRSTDATE – vrací první datum v aktuálním kontextu

LASTDATE – vrací poslední datum v aktuálním kontextu

DATESMTD – vrací tabulku obsahující sloupec kalendářních dat od začátku měsíce v rámci aktuálního kontextu

DATESQTD – vrací tabulku obsahující sloupec kalendářních dat od začátku čtvrtletí v rámci aktuálního kontextu

DATESYTD – vrací tabulku obsahující sloupec kalendářních dat od začátku roku v rámci aktuálního kontextu

**Filtrovací funkce**

ALL – vrací všechny řádky tabulky či sloupce, ignoruje všechny použité filtry

FILTER – vrací tabulku, která představuje výsledek filtrů aplikovaných na tabulku

## Operátory

Aritmetický operátor	Význam	Příklad
+ (znaménko plus)	Sčítání	3+3
- (znaménko minus)	Odcítání nebo znaménko minus	3-1-1
* (hvězdička)	Násobení	3*3
/ (lomítko)	Dělení	3/3
^ (stříška)	Umocnění	16^4

Operátor porovnání	Význam	Příklad
=	Je rovno	[Region] = "USA"
==	Je přesně rovno	[Region] = "USA"
>	Je větší než	[Sales Date] > "Jan 2009"
<	Je menší než	[Sales Date] < "Jan 1 2009"
>=	Větší než nebo rovno	[Amount] >= 20000
<=	Menší než nebo rovno	[Amount] <= 100
<>	Není rovno	[Region] <> "USA"

Textový operátor	Význam	Příklady
&& (dvojitý ampersand)	Vytvoří podmínu AND mezi dvěma výrazy, z nichž každý má jako výsledek logickou hodnotu. Pokud oba výrazy vrátí hodnotu TRUE, kombinace výrazů také vrátí hodnotu TRUE, v opačném případě kombinace vrátí hodnotu FALSE.	([Region] = "France") && ([BikeBuyer] = "yes"))
(dvojitý symbol svislé čáry)	Vytvoří podmínu OR mezi dvěma logickými výrazy. Pokud některý výraz vrátí hodnotu TRUE, výsledek je TRUE. Výsledek je FALSE jenom v případě, že oba výrazy jsou FALSE.	(([Region] = "France")    ([BikeBuyer] = "yes"))
IN	Vytvoří logickou podmínu OR mezi každým řádkem porovnávaným s tabulkou. Poznámka: Syntaxe konstruktoru tabulky používá složené závorky.	'Product'[Color] IN { "Red", "Blue", "Black" }