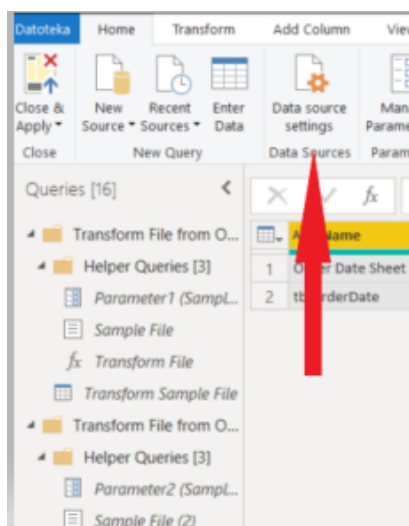
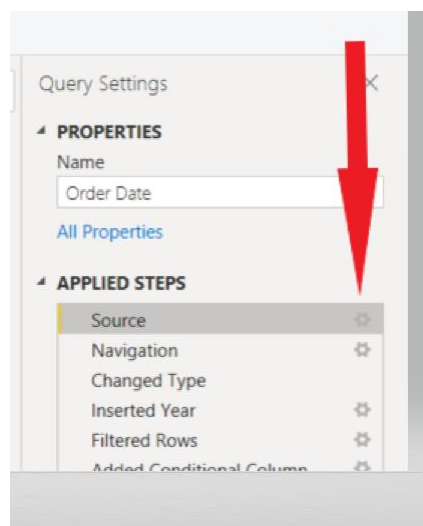


Kratka ponovitev 2. in 3. srečanja.

Na 2. srečanju smo si pogledali kako v Power Query (v nadaljevanju PQ) uvozimo še par drugih tipov datotek. Tisti tabeli iz spleta smo dodali še excelove datoteke, tekstovne datoteko in mapo. To smo storili na tak način, da smo si iz spleta prenesli **henkel01.zip** datoteko in njeno vsebino spravili v mapo **c:\powerbihenkel**. Pot je pomembna, oz. je bila na začetku, saj smo pokazali tudi možnosti kako nastaviti pot (glej slike).







V zavihku home



Kliknemo na zobnik v korakih

Ko smo pgnali to našo datoteko **primer01start.pbix** smo v podoknu polja videli, da so v tem modelu že naložene 3 datoteke. Za tisto iz spleta ni problem, ker se naslov ni spremenil, ti dve pa smo uvozili iz diska in zato je pomembno, da je pot nastavljena pravilo.

Naš model smo pričeli dopolnjevati najprej z excel datoteko, kjer smo si pogledali razliko med uvozom tabele ali celotnega lista (če je urejena vedno uvozimo tabelo).

<input type="checkbox"/> 	<input type="checkbox"/> 	Samo podatki v tabeli.
<input type="checkbox"/> 	<input type="checkbox"/> 	Vsi podatki na listu.

Uvoz tabela / list	Uvoz tekst / CSV
--------------------	------------------

File Origin	Delimiter	Data Type Detection
1250: Central European (Windows)	Tab	Based on first 200 rows

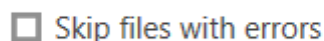
Pri tekst datoteki PQ prepozna tip po prvih 200 vrsticah ali če spremenimo, gre pogledat vse (načeloma ni potrebno) in je bil potreben še kak korak več, da smo sami spravili podatke v tip tabele (prvo vrstico smo povzdignili v glavo tabele).

Majhna razlika se pojavila pri uvozu mape. Če imamo tabele z isto vsebino, kot v našem primeru prodaja po državah, v isti mapi, jo lahko uvozimo kot mapo. Ne, da lahko, še bolje je, da uvozimo mapo, ker če bi se v mapo dodala ena datoteka npr. Sloveniasales, bi jo pri PowerBI pri posodobitvi avtomatsko dodal naslednjič, ko stisnemo refresh. Enako je mapa boljša opcija, če pride Corona in se npr. Avstralija odloči, da več naših uslug ne potrebuje in če mi uvažamo mapo, bo PowerBI gledal

samo kaj je v mapi in bo uvozil kar pač ima. Če pa bi ročno dodajali datoteko po datoteko, pa bi se model sesul, ker ne bi našel Avstralije.

Opcija za uvoz mape pride prav, če npr. nimate dostopa do vseh podatkov, pa vam periodično nalagajo posodobitve na določeno podatki so po navadi v točno določeni obliki.

Pri uvozu mape še levo spodaj opazimo možnost »skip errors« (preskoči napake).

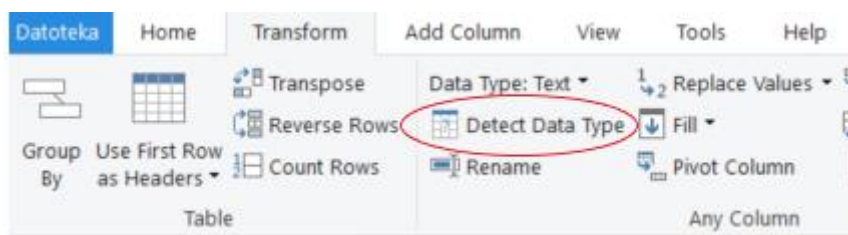


Pri uvažanju mape pa se lahko zgodijo tri stvari.

- Ali vse štima
- Ali preskoči napake (če označimo to možnost)
- Ali zablokira

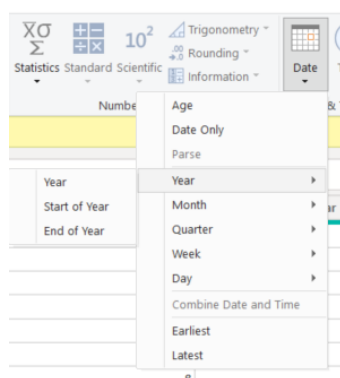
S tem dodajanjem smo prišli na iz naših 3 querijev (poizvedb) na 7. Teh 7 smo začeli pripravljat na nadaljnjo obdelavo – začeli smo jih čistiti. To pomeni, da smo stolpce preimenovali v večini primerov nastavili podatkovni tip, par pa smo jih pustili nalašč napačnih, da bomo danes videli, kako izgleda, ko pride do napak pri dodajanju in združevanju.

Pri nastavljanju pravilnega podatkovnega tipa smo si pomagali tudi s funkcijo »auto detect«, ki načeloma dela super, se pa zna zgodit, da hoče včasih malo preveč pomagat – kar smo opazili pri *Order Date* – stolpec Month Year.



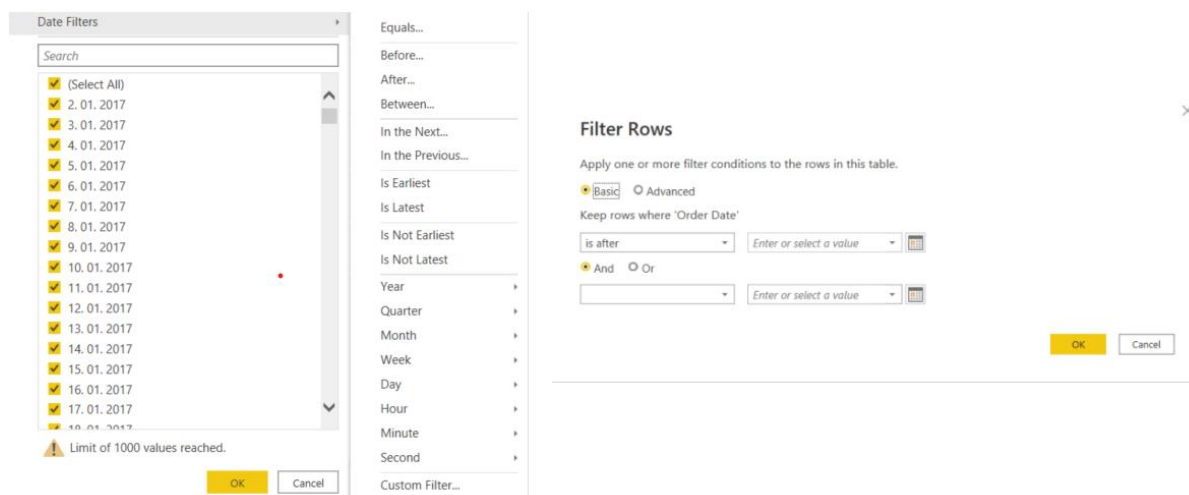
Napaka je bila vstavljena, da nas opozori, da moramo poznati svoje podatke in jih po uporabi avtomatskih funkcij še enkrat preveriti. Ker to niso vaši podatki, seveda tega niste mogli vedeti, razen, če ste si doma prej odprli to tabelo.

V tej tabeli smo še uporabili funkcijo izvažanja, da smo pokazali, kako lahko iz stolpcev dobimo podatke, odvisno od nastavljenega podatkovnega tipa (na sliki primer podatkovnega tipa »Date«).

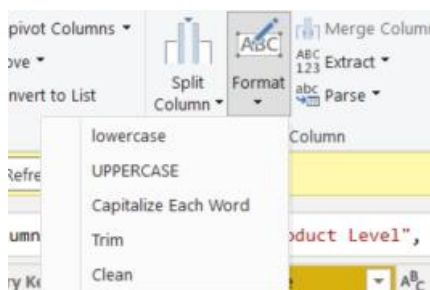


Verjetno ste že opazili, da se zgoraj na traku pri različnih zavihkih ukazi ponavljajo. Razlika je, da če ukaz izvedem pod zavihkom *Rransform*, se bo transformacija zgodila tukaj, v tem stolpcu, če pa to opravim pod zavihkom *Add Column*, pa mi bo PQ na koncu vrigel nov stolpec, tako kot smo to storili mi.

Od nastavljenega podatkovnega tipa bodo odvisni tudi filtri, pri našem datumskem so tako na voljo (na sliki tipi filtrov in primer okna, ko izberemo možnost *After*)



V tabeli *Product* smo delali še največ čiščenja, v prvem stolpcu smo se znebili duplikatov (desni klik na stolpec in *Remove Duplicates*). Pri imenih pa smo prikazali delovanje formatiranja, kjer smo uporabili *Trim*, *Clean* in pa *Capitalize Each Word*, ker pač včasih pridejo podatki iz sistemov, ki se med sabo v celoti ne podpirajo in se ti »skrivni znaki« odstranijo.

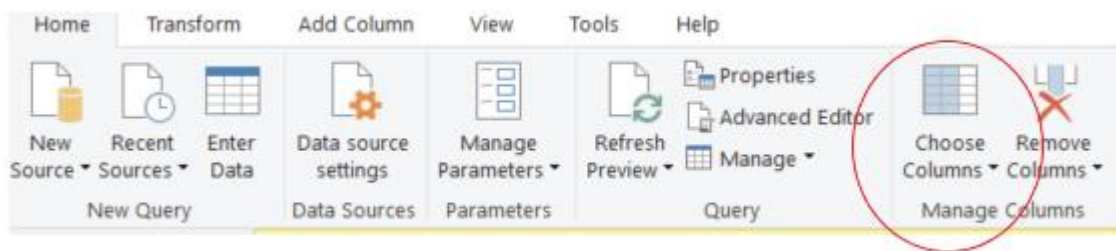


Da smo pokazali, koliko je teh znakov, smo dodali še dva stolpca in sicer dolžina pred čiščenjem in pa po, do tega smo prišli, da smo v zavihku *Add Column* extractali (izvozili) dolžino (lokacija na zgornji sliki desno od možnosti *Format*).

Prikazali smo tudi menjavo vrednosti pri opisu produktov, kjer smo menjali *null* z »brez opisa«. To smo storili z desnim klikom na stolpec in opcijo *Replace Values*.

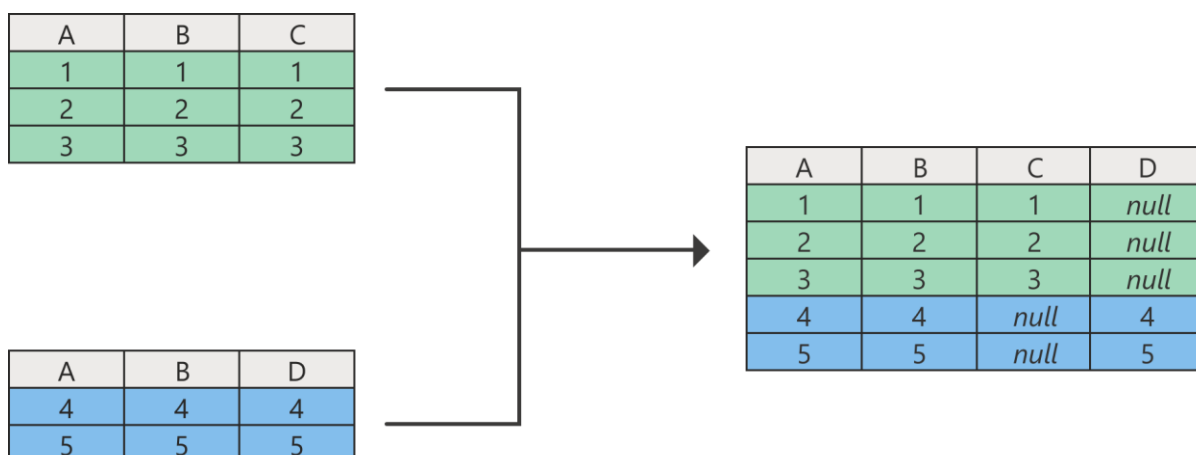
V datoteki s teritoriji smo stolpec tudi razbili na več delov (*Split Column* – glej zgoraj). Delimeter (ločilo), po katerem delimo PQ prepozna avtomatsko, vedno pa še obstaja možnost, da si ga pod *Custom* nastavimo sami.

Pri čiščenju podatkov pa je potrebno odstraniti tudi, za analizo nekoristne, stolpce. Lahko jih odstranimo z desnim klikom in izberemo *Remove Columns*, boljša opcija pa je, da gremo na zavihek home in pod skupino ukazov *Manage Columns*, izberemo možnost *Choose Columns*.



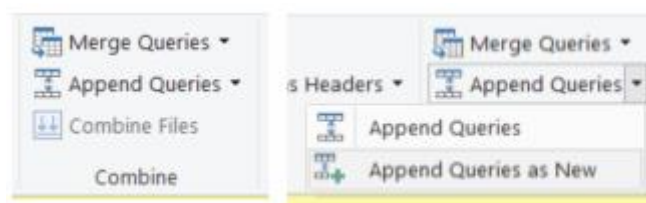
To je boljši in varnejši način brisanja, sploh, če gre za tabelo z ogromno stolpci. Tu imamo opcije da izbiramo stolpce ali gremo na stolpce. Označim katere želim obdržati, kliknem *Ok* in pogledam k našim korakom. Vidimo, da se ime koraka spremeni iz *Removed Column* na *Removed Other Columns*. In spet podobno, kot je bila situacija prej pri mapah, če uporabimo to možnost, ko nekdo doda kaj v tabelo bo model še vedno deloval, ker odvzame vse ostale. V nasprotnem primeru pa ne.

Potem smo si še pogledali kako deluje kombiniranje querijev. Obstajata dva načina *Append* (dodajanje) in *Merge* (združevanje). Da bo lažje razumljivo, si bomo prvo pogledali na slikah kako delujeta v principih.



Append oz. dodajanje (<https://docs.microsoft.com/en-us/power-query/append-queries>)

Če imam dve tabeli in želim eno dodati drugi (eno pod drugo), stisnem *Append*. To možnost (kot tudi *Merge*) najdem na traku v skupini *Combine Columns*.



Pri obeh možnostih združevanja se ob kliku na puščico dol, pojavi možnost, da ustvari se poizvedbi združita ali pa se ustvari nova.

Če imata tabeli enako število stolpcev, ki so enakega podatkovnega tipa in imena, potem seveda ni nobenega problema, ker se zgodi, da če je leva tabela 1 in desna 2, bo desno samo prilepil spodaj k levi. To so idealne razmere, ki pa, kot si lahko predstavljate, so bolj kot ne redkost. V večini primerov

se nam zgodi, da stolpci ne štimajo ali po tipu ali pa po imenu. In v takem primeru se zgodi to, da bo PBI dodal stolpec na tak način in dobili bomo en kup teh sivih vrednost, ki bojo v tabeli imeli vrednost *null*. Tega nočemo in če se le da to sami prej očistimo.

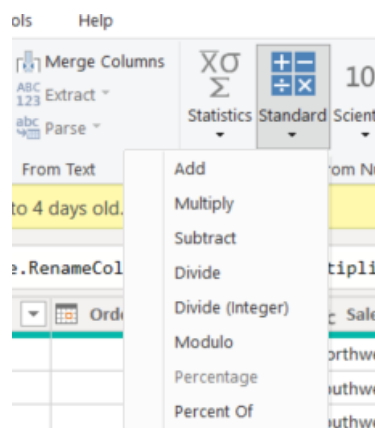
Končni uporabnik, ni nujno tisti, ki bo podatke pripravljaj, bo imel na voljo to okno. In v tem oknu, če se bo analizirala prodaja, bi bilo verjetno bolje, da bi imeli celotno prodajo, ne pa Ameriko ločeno od sveta, ker če želimo ločit, lahko to preprosto naredimo s filtri.

Prisotni na srečanju so imeli nalogo, da uredijo tabelo z vsemi državami, na način, da jo bomo lahko združili s prodajo po ZDA. Torej, da smo z opcijo *Choose Columns*, odstranili stolpce in obe tabeli preoblikovali, da sta imeli enako število stolpcev.

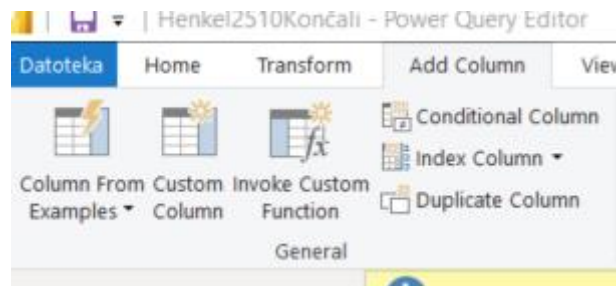
Pri dodajanju smo ustvari nov query, ki dobi privzeto ime *Append1*. Preimenovali smo ga v *Sales* in uredili napake, ki so nastale zaradi napačnega poimenovanja in napačnega tipa (smo jih pred tem pustili nalašč). Na kratko, če združimo dve tabeli s po pet stolpci in ime združen query več stolpcev, to pomeni, da nismo bili najbolj točni z našimi poimenovanji stolpcev. Če pa se nam ustvari podatkovni tip Any, pa niso bili pravilno nastavljeni podatkovni tipi. Napak ne popravljamo v združenem queriju, temveč gremo v izvirne (lahko bi jih tu, a ne vemo, če bomo tiste še kdaj potrebovali, ali jih bo kdo drug... tako nam narekuje dobra praksa). Ko so napake v izvornih querijih popravimo, se avtomatsko popravijo tudi v združenem.

Po brisanju stolpcev je sledilo še dodajanje le teh. Pogledali smo si načine dodajanja stolpcev v PQ. Pogledali smo si že opcijo *extract*, ko smo gledali dolžino. Ne pozabit enake komande so pod *Transform*, kot *Add Column*, tako da če nismo dobili novega, smo verjetno pozabili prestavit.

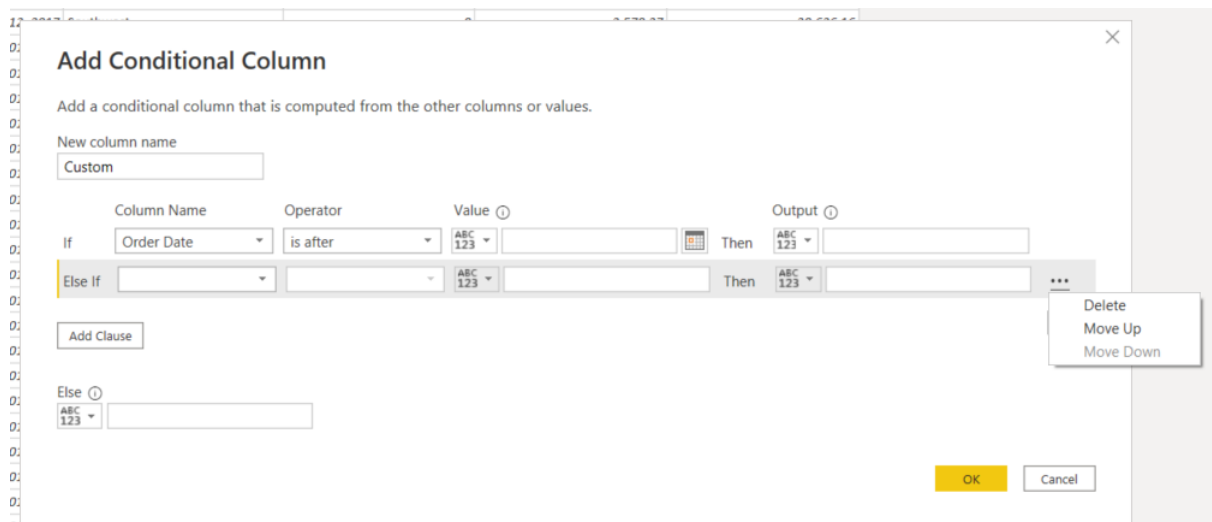
Za izvajanje osnovnih računskih operacij med stolpci je najenostavneje, da označimo željene stolpce (ctrl + levi klik), ter kliknemo na željeno operacijo.



V PQ obstaja tudi možnost, dodajanja stolpca iz primera, torej *Column From Example* (kot v Excelu). To pomeni, da vpišem vzorec iz enega stolpca. Kliknem zgoraj na opcijo in mu lahko naročim, da naj gleda samo označen stolpec ali pa vse. Ustvari se nov stolpec in začnem vpisovat, kar želim, ko sem zadovoljen stisnem Ctrl + Enter in PBI bo celoten stolpec zapolnil z istim vzorcem.



Pri ustvarjanju novih stolpcev, pa je potrebno še omeniti *Conditional Columns*, torej stolpce, pri katerih pišemo pogoje – IF stavke.

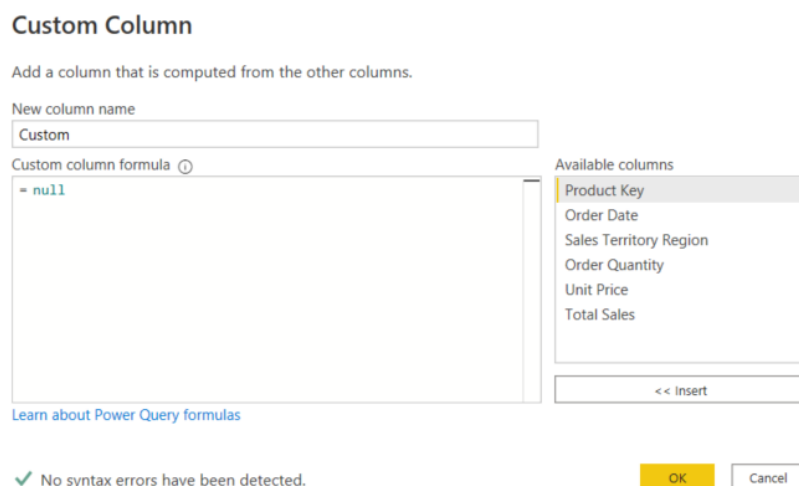


Pogoje poljubno dodajam z opcijo *Add Clause*. Če želim kaj izbrisati ali spremeniti vrstni red, kliknem na tropičje ob strani in to uredim.

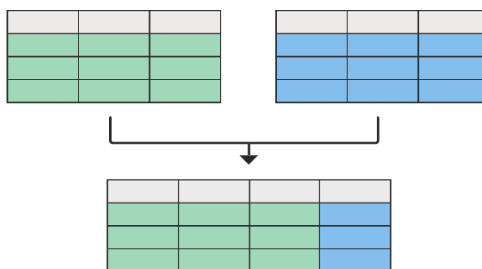
Če pozabimo vpisati vrednost – *Value*, nas PQ preprosto ne bo spustil naprej, če pozabimo *Output*, pa se bodo celice zapolnile z vrednostjo *null*.

Pogoje lahko pišemo tudi v *Custom Column*, ali pa izven PQ, v PowerBi, s čimer pa se bomo se srečali.

Custom column opcija nam pride prav tudi, če želimo dodati prazen stolpec v PQ, s tem, da pač njegovo vrednost nastavimo na *null*.



Torej za *Append* oz. dodajanje druge tabele pod prvo, je dobro, če štimajo stolpci. *Merge* oz. združevanje, pa ima malo več opcij, ker tukaj združujemo po vrsticah. Koncept združevanja na slikah:



Merge (<https://docs.microsoft.com/en-us/power-query/merge-queries-overview>)

Imamo dve tabeli, kjer se prvi dve vrstici ujemata, zadnja pa ne. Najpogostejši tip združevanja je *Left Join*, ki pomeni, da drugo tabelo, ki jo dodajamo pripojim k levi. Pripojile pa se bodo samo vrstice druge, ki se ujemajo z vrstico v prvi, ostale se izgubijo.

Right Join, deluje ravno obratno, tu se nam izgubijo vrstice iz prve, ki jo združimo z drugo.

Potem imamo tu še *Inner Join*, ki bo združil samo tiste vrstice, ki se ujemajo, izgubila se bosta tako tiger, kot slon.

In pa še *Full Join*, ki bo združil vse, ne glede na ujemanje. Kjer bi bil slon v desni tabeli in pa tiger v levi, bo vrednost *null*.

To niso vse opcije, sta še dve anti-opciji, leva in desna. *Left anti join* pripelje samo vrstice iz leve tabele v skupno, *Right anti join* deluje enako v nasprotni smeri.

Join kind	Icon	Description
Left outer		All rows from the left table, matching rows from the right table
Right outer		All rows from the right table, matching rows from the left table
Full outer		All rows from both tables
Inner		Only matching rows from both tables
Left anti		Only rows from the left table
Right anti		Only rows from the right table

<https://docs.microsoft.com/en-us/power-query/merge-queries-overview>

Ker že imam označen *Product*, moram izbrati samo še drugo tabelo. Najdem naš *Product Level*.

Select a table and matching columns to create a merged table.

Product Key	Subcategory Key	Product Name	Product Description	Standard Cost	Color	Text Before
AR-5381	<i>null</i>	Adjustable Race	Brez opisa	<i>null</i>	NA	AR
BA-8327	<i>null</i>	Bearing Ball	Brez opisa	<i>null</i>	NA	BA
BE-2349	<i>null</i>	Bb Ball Bearing	Brez opisa	<i>null</i>	NA	BE
BE-2908	<i>null</i>	Headset Ball Bearings	Brez opisa	<i>null</i>	NA	BE

ProductSubcategoryKey	Product Subcategory	Product Category
1	Mountain Bikes	Bikes
2	Road Bikes	Bikes
3	Touring Bikes	Bikes
4	Handlebars	Components
5	Bottom Brackets	Components

Left Outer (all from first, matching from second)

☐ Use fuzzy matching to perform the merge

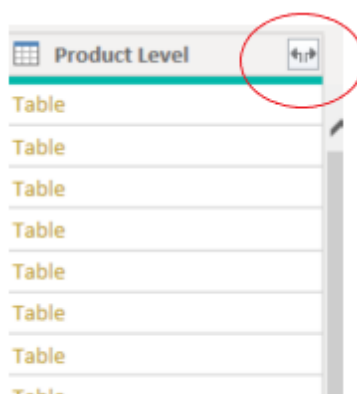
- > Fuzzy matching options

✓ The selection matches 295 of 504 rows from the first table.

OK

Cancel

Po združevanju se nam pokaže v queriju, ki je bil izbran kot prvi, dodaten stolpec. To v resnici ni stolpec, ampak gre za tabelo, kar vidimo po vrednostih v celicah in pa »razširjevalniku« ob strani (obkrožen na sliki).



Ko kliknemo nanj, se nam prikažejo vsi stolpci, katere smo prinesli iz druge tabele. Ko obkljukamo stolpce, ki jih hočemo, se nam dodajo vsi. Na tak način smo združili naši prvi dve tabeli. Za razumevanje smo še preverili, če znamo združiti tabeli s prodajnimi teritoriji in svetovnim prebivalstvom, na način, da se država pripoji samo vrstica s številom prebivalcev.