

PowerPivot és a DAX

Margitfalvi Árpád

Copyright © Margitfalvi Árpád, 2019

tartalomjegyzék

tematika és koncepció	
mit	2
	2
alaptogalmak telenítés	З
kezelő felület	3
táblák	4
kapcsolat	4
importálás munkafüzetből	
munkafüzet kiválasztása	5
munkalapok kivalasztasa	6
táblák a bővítmény ablakában	0 8
lokális adatkapcsolatok	9
a táblák mentése	9
univerzális adatkapcsolat létrehozása	10
adattípusok	
a DAX adattípusai	13
az adattipus megallapítása	13
szelektív importálás	14
a villámnézet ablaka	16
az oszlopok szelektálása	16
a sorok szelektálása	17
szövegfájl importálása	
fogalmak	19
beolvasas	19
	20
fogalmak	21
az egyesítő tábla létrehozása	22
a táblázat előkészítése	23
rekordok hozzáadása az egyesítő táblához	24
további vágólap-műveletek	25
tábla adatbázís-táblázatból	26
rogalmak a tábla létrebozása	26
tábla a kimutatás forrásából	20
létrehozás	27
táblák	
az adatnézet felépítése	28
tábla-műveletek	29
rekord-navigáció	29
kijelölés	30
Mezok kezelese rokordok automatikus rondozóso	30
rekordok felhasználój rendezése	32
a mező adattípusa és számformátuma	32
műveletek visszavonása ismétlése	32
tábla frissítése	33
kapcsolatok beolvasása	
fogalmak	34
kapcsolatok importalasa	34
műveletek kancsolatnézetben	36
kancsolatok automatikus létrehozása	50
fogalmak	37
a modul működése	37
kapcsolat manuális létrehozása	
fogalmak	40
nem kapcsolódó rekordok	40
a kapcsolat letrenozasa a kapcsolat törlése és megszakadása	41 12
Data Analysis Expressions	40
a DAX képlet jellemzőj	44
nevek a DAX képletében	45
adattípusok konvertálása a képletben	45

számított objektumok	
számított mező fogalma	46
számított mező létrehozása	46
képletek újra számolása	48
automatikus osszesítés	49
czűrők a DAX képletben	50
mezőhívatkozások	54
PowerPivot-kimutatás	J_
a hővítmény kimutatás-modulia	56
a PowerPivot-kimutatás segédablaka	56
kimutatás és kimutatásdiagram létrehozása	58
tételek egyedi sorrendje a kimutatásban	59
egyéni nézetek	
fogalmak	60
egyéni nézetek kezelése	60
több a többhöz kapcsolat	
fogalmak	62
számított mező több a többhöz kapcsolatban	62
összesítések több a többhöz kapcsolatban	63
feltételes formázás a kimutatásban	
fogalmak	66
a felteteles formazas kezelese	66
celertekes elemzes letrenozasa	66
	69
	71
dátumok kozolóso	71
tételek csonortosítása a kimutatásban	71
karakteres tételek csonortosítása	72
numerikus tételek csoportosítása	73
dátum-tételek csoportosítása	74
dátumok automatikus csoportosítása	75
a naptár-tábla	76
a naptár-tábla létrehozása	77
a naptár-tábla kapcsolatai	79
naptár-tábla a kimutatásban	80
időszak-kezelő függvények	80
meghatározott napok adatainak feldolgozása	81
dátumot adó időszak-kezelő függvények	82
dátumokat adó időszak-kezelő függvények	83
az időszak-kezelő függvények rendszerezése	88
idő-kezelő függvények	90
információs függvények	91
logikai fuggvenyek	94
elagazasok kezelese	95
elojeles szamok kezelese	96
	97
teszt-értékek generálása	97
további matematikaj függvények	99
szám és szöveg konvertálása	99
hierarchia kezelése	104
hierarchia a kimutatásban	105
hierarchikus mezőcsoport	106
egyedi bejegyzések lekérdezése	108
szűrő-kezelés a képletben	110
szűrő-információk	113
szűrőfüggvények	116
kereső függvények	122
statisztikai függvények	125
szöveg-kezelő függvények	131
tábla-kezelő függvények	136
tabla-kezelő függvények rendszerezése	149
tüggvények előfordulása	151

tematika és koncepció

mit

A PowerPivot (ErősKimutatás) bővítmény az adatelemzés eszköze az Excel programban, amellyel különböző fájlokban és különböző formátumokban tárolt adathalmazokat egyetlen egységként vizsgálhatunk. A bővítmény az elemzendő adatokról másolatot készít, majd a másolatokat saját formátumára alakítja át és táblákban helyezi el. A táblák a PowerPivot saját windows-ablakában állnak. A táblákban tárolt adatok nem módosíthatók, de az aktualizálás lehetőségét a bővítmény biztosítja.

A táblák egyetlen egységként való kezelését a köztük lévő kapcsolatok teszik lehetővé. A kapcsolatokat általában a felhasználó hozza létre, de a bővítmény képes a létező kapcsolatokat a forrásadatokkal együtt beolvasni, illetve meghatározott feltételek teljesülése esetén automatikusan létrehozni. A kapcsolatokkal egyesített táblák alkotják a PowerPivot adatbázist.

Az adatbázis kimutatásos elemzése már az Excel ablakban történik, a PowerPivot saját kimutatás moduljával, amelyről a bővítmény a nevét kapta.

A PowerPivot képleteiben a "Data Analysis Expressions" (rövidítve "DAX") nevű, a bővítményhez tervezett, számítási rendszert kell használnunk. Ez azt jelenti, hogy a PowerPivot és az Excel operátor- és függvény-készlete, valamint szemantikai és szintaktikai szabály-rendszere eltérő!

A bővítmény elsősorban összekapcsolt táblák egy egységként történő elemzésére szolgál, de önálló, kapcsolatok nélküli táblák vizsgálatára is alkalmas.

Ez a leírás a 2019-es Excel verzió "Microsoft PowerPivot for Excel" nevű, COM bővítményt mutatja be, de csak azokat a funkcióit tárgyalja, amelyek önállóan, más szoftver igénybevétele nélkül is működnek! A leírás nem foglalkozik a DAX és más számítástechnikai nyelvek kapcsolatával és csak Excel munkafüzetben, formázást nem tartalmazó szövegfájlban, illetve Access adatbázisban álló adatok feldolgozását ismerteti!

hogyan

Ez a leírás olyan gyakorlott Excel felhasználóknak készült, akik még teljesen kezdők a PowerPivot használatában. A szemléltető példák a lehető legegyszerűbbek, de a fogalmak egyértelmű meghatározása számos ponton "nehéz" szöveget eredményezett.

Az Excel és a PowerPivot objektumainak megkülönböztetésére a következő fogalom-párosokat használom, program - bővítmény (Excel - PowerPivot) sorrendben: táblázat - tábla, sor - rekord, oszlop - mező, cella - bejegyzés.

A leírásban az utasítások elemei a következő sorrendben követik egymást: a menüszalag lapjának neve, a csoport neve, a parancslista vagy a parancsgomb neve. Az utasítások kiemelését dőlt betűformázással oldottam meg.

A szövegben a perjel (/) két fogalom vagylagos összekapcsolását jelenti. Tehát a "hónap/negyedév/év" szöveg jelentése: hónap vagy negyedév vagy év.

A bővítményben létrehozott adathalmaz egy kvázi adatbázis. Azért nem igazi adatbázis, mert a PowerPivot sokkal "megengedőbb" szabályrendszert alkalmaz, mint az adatbázis-kezelő szoftverek. Ennek ellenére, a jó olvashatóság érdekében, a szövegben a jelző nélküli fogalmat használom.

> Margitfalvi Árpád margitfalvi.arpad@gmail.com

alapfogalmak

telepítés

A Microsoft Excel 2019-es verziójának telepítése után a PowerPivot bővítményt is telepítenünk kell. Először jelenítsük meg a COM-bővítmények parancstábláját a Fájl, Beállítások, Bővítmények, Kezelés: COM-bővítmények, Ugrás... utasítással, majd a Létező bővítmények listából válasszuk a Microsoft PowerPivot for Excel tételt.

Egy esetleges összeomlása után a program automatikusan letilthatja a bővítményt. Ebben az esetben a *Kezelés* lista *Letiltott bővítmények* elemével kezdeményezhetjük a PowerPivot ismételt betöltését.

A bővítmény 2019-os verziója nem kompatibilis a 2010-es változattal, ezért az ezekkel a verziókkal készült munkafüzetek PowerPivot tartalmával csak a fájl konvertálása után dolgozhatunk. A program felajánlja az átalakítást és jóvágyásunkat követően a konvertált változattal felülírja az eredeti munkafüzetet.

kezelő felület

A bővítmény vezérlőit a program menüszalagjának *PowerPivot* lapja és a bővítmény ablaka tartalmazza. Az elemzendő adatok másolatát a PowerPivot saját formátumára alakítja át és táblákba helyezi. A táblák megjelenítése és kezelése a bővítmény ablakában történik.

A táblák közötti kapcsolatok deklarálásával létrehozott adatbázis kimutatásos elemzése a program ablakában történik. A táblák és a kimutatások a munkafüzetben tárolódnak. A mentett fájl kiterjesztése: xlsx. Magyarul, nincs PowerPivot fájl és nincs PowerPivot fájlformátum. A munkafüzet PowerPivot tartalmára a fájlnév végé álló PP betűkkel utalhatunk. Például: éves jelentés PP.xlsx.

A programból a PowerPivot, Adatmodell, Kezelés utasítással léphetünk át a bővítmény ablakba. Az Excel-ablakba a bővítmény gyorselérési eszköztárának Váltás a munkafüzetre utasításával térhetünk vissza. A bővítmény ablakának bezárása nem okoz adatvesztést, mert a PowerPivot adatok a számítógép operatív tárjában tárolódnak. Mentésük a gazda-munkafüzet mentésével történik.



1. ábra a bővítmény- és a program ablaka közötti váltás vezérlői

A bővítmény nyelve, a szokásos beállítások mellett, azonos a program nyelvével. Ettől eltérő megjelenítést a program ablakában a PowerPivot, Beállítások, Nyelv utasítással kezdeményezhetünk.

táblák

A vizsgálandó adatok másolatát a PowerPivot táblákban helyezi el. A tábla oszlopai a mezők, nevük az oszlop első cellájában áll. A tábla sorai a rekordok. A rekord meghatározott mezőjében álló adat a mezőbejegyzés, vagy röviden bejegyzés. A hiányzó adatot "üres bejegyzésnek" nevezzük.

Az elemzendő, eredeti, adathalmaz a tábla forrása. A tábla-készítés első lépése a forrás-adatok beolvasása. A "beolvasás" kifejezés metafora, valójában az adatok másolása történik ebben a fázisban. A bővítmény a beolvasott adatok vizsgálata alapján állítja be a mezők adattípusát. Minden mezőben csak a beállított típusú adat kerülhet, ezért az ettől eltérőeket a bővítmény átalakítja a mező adattípusára, vagy ha ez nem lehetséges, akkor törli őket. Ez az egységesítő folyamat a homogenizálás.

Tábla készülhet belső vagy külső forrásból. Belső forrásnak nevezem, azt az adathalmazt, amely abban a munkafüzetben áll, amelyből a bővítményt elindítottuk. Külső adatforrás állhat másik munkafüzetben, text-fájlban, Access adatbázisban...



2. ábra külső forrás beolvasásának vezérlői a bővítmény ablakában

A külső forrás beolvasását importálásnak nevezzük, amelyet a felhasználó egy varázsló segítségével felügyel. A bővítmény a külső forrást tartalmazó fájl nevét, elérési útvonalát és egyéb tulajdonságait névvel azonosított objektumba menti el, amelynek neve adatkapcsolat. Az adatkapcsolat biztosítja a táblák frissíthetőségét.

kapcsolat

A kapcsolat két tábla összetartozásának deklarációja, amely a két tábla egy-egy mezőjét tartalmazza. Ezek a mezők a táblák kapcsoló mezői. Valójában nem a két tábla, hanem rekordjaik tartoznak össze. Gondoljunk a "megrendelők" és "megrendelések" táblák rekordjaira. A kapcsoló mezők azonos bejegyzései biztosítják az összetartozó rekordok "lekérdezhetőségét". Például a megrendelők tábla "megrendelő AZ" mezőjének "007" bejegyzése a megrendelések tábla három rekordjának "megrendelő AZ" mezőjében található meg. Ezek szerint a 007-es azonosítójú megrendelőnek három megrendelését tároljuk a megrendelések táblában.

A kapcsolatok létrehozása a felhasználó feladata, de meghatározott feltételek teljesülése esetén, a felhasználó utasítására, a PowerPivot is el tudja készíteni a deklarációt. Access-táblák importálásakor a bővítmény a táblák közötti kapcsolatokat is be tudja olvasni a forrás-fájlból.

A bővítmény a kapcsolatokkal egyesített tábla-csoportot nem adatbázisnak, hanem adatmodellnek nevezi, aminek semmi értelme! A hétköznapi nyelvhasználat modell szavának számtalan értelmezése közül, egynek sincs köze, még tágabb értelmezésben sem, a PowerPivot objektumokhoz. Ráadásul az adatmodell fogalom az adatbázis-kezelés terminológiájában teljesen mást jelent.

importálás munkafüzetből

munkafüzet kiválasztása

Az elemzendő adatok beolvasását a Tábla importálása varázsló elnevezésű segédprogram végzi, amelyet a PowerPivot ablak Kezdőlap, Külső adatok beolvasása, Más forrásokból parancsával indíthatunk el.

A segédprogram első, Kapcsolódás adatforráshoz panelján listát találunk a lehetséges adatforrásokról. Válasszuk ki a Szövegfájlok csoportban az Excel-fájl elemet, majd kattintsunk a Tovább gombra.

lábla importálása varázsló	?	\times
Kapcsolódás adatforráshoz Létrehozhat kapcsolatot adatforrással, vagy használhat már létező kapcsolatot.		
Relációs adatbázisok		^
Microsoft Access Kapcsolatot létesíthet egy Microsoft Access-adatbázissal. Importálhat az adatbázisból táblákat, nézeteket vagy egy lekérdezés által visszaadott adatok	at.	
Szövegfájlok		
Adatokat importálhat egy Excel-fájlból.		
Szövegfájl Adatokat importálhat egy szövegfájlból.		~
< <u>V</u> issza <u>I</u> ovább > <u>B</u> efejezés	Mégs	e

3. ábra a segédprogram első panelja

A varázsló második, *Kapcsolódás Microsoft Excel-fájlhoz* feliratú paneljának *Tallózás…* nyomógombjával választhatjuk ki a munkafüzetet. Megnyitott fájlt ne válasszunk, mert abból nem tudja a bővítmény kiolvasni az adatokat. Egy munkamenetben csak egy Excel fájlból tudunk importálni. Ebből következően, ahány munkafüzetünk van, annyiszor kell a műveletsort megismételni.

Tábla importálása varázsló		? X
Kapcsolódás Microsoft Excel-fájlh Adja meg a Microsoft Excel-fájlh	oz noz való kapcsolódáshoz szükséges adatoka	t.
Kap <u>c</u> solat felhasználóbarát neve:	Excel 01 munkalapok	
E <u>x</u> cel-fájl elérési útja:	E:\temp\01 munkalapok.xlsx	Talló <u>z</u> ás
	Az első sor adatai alkossák az <u>o</u> szlopf	ejléceket.
	<u>S</u> peciális <u>K</u>	apcsolat tesztelése
$\sim \sim$	\sim	

4. ábra a segédprogram második panelja

A kiválasztott fájl nevét és elérési útvonalát a bővítmény automatikusan adatkapcsolatba menti. Az objektum neve: Excel<szóköz><munkafüzet neve>. Egyedi nevet a Kapcsolat felhasználóbarát neve mezőben adhatunk meg.

A létrehozandó tábla mezőnevei azonosak lesznek a forrás táblázat oszlopneveivel, ha kipipáljuk *Az első sor adatai alkossák az oszlopfejléceket* jelölőnégyzetet. Ha a vezérlő jelöletlen marad, akkor a mezők az F1, F2, F3... neveket kapják és az esetleges oszlop-nevek is elemzendő adatnak minősülnek.

munkalapok kiválasztása

A varázsló harmadik, Táblák és nézetek kijelölése feliratú panelján a munkafüzet lapjai és elnevezett tartományai közül választhatunk. A megjelenített listában az objektumok, nevük szerint, ABC sorrendbe szedve követik egymást.

Tábla importálása varázsló ?									
Táblák és nézetek kijelölése Válassza ki azokat a táblákat, illetve nézeteket, amelyekből adatokat kíván importálni.									
Fájlnév: E:\tananyagok\táblázatkezelés\Excel Mester\PowerPivot\2019\feladatok\01 munkalapok.xlsx									
	s liezelek.								
	Forrástábla	Felhasználóbarát név	Szűrő részletei						
	Bicske\$	Bicske							
	Csurgó\$	Csurgó							
	Csurgó\$_xInm#_FilterDatabase								
	Fonyód\$	Fonyód							
	Hatvans	Hatyan							

5. ábra a segédprogram harmadik panelja

A bővítmény a munkafüzet tartalmi vizsgálata nélkül állítja össze a lapok jegyzékét, ezért abban üres munkalapok is szerepelhetnek. A beolvasott munkalap-neveket a program dollárjellel egészíti ki: <lap-név>\$ illetve, ha a név szóközt is tartalmaz, akkor a karakterlánc aposztrófok között áll: '<lap ne-ve>\$'. A tartomány-nevek után a bővítmény nem tesz dollárjelet.

Az importálandó elemeket jelölőnégyzetekkel választhatjuk ki. A lista bal felső sarkában álló vezérlővel az összes lapot ki tudjuk jelölni. Ha a munkafüzet csak egyetlen lapot tartalmaz, akkor azt a varázsló automatikusan kiválasztja és tovább lép.

A listában hasonló nevű lap-párok is előfordulhatnak: <lapnév>\$ és <lapnév>\$_xlnm#_FilterDatabase. Az utóbbi a szűrés végrehajtásához szükséges segéd-objektum az Excelben, csak itt a segédprogramban a hanyag programozók nem rejtették el. A két munkalap, mellesleg, teljesen azonos.

A Villámnézet és szűrés nyomógomb parancstáblát jelenít meg a kiválasztott munkalap táblázatának első ötven sorával és az adatok szelektálására szolgáló eszközökkel. Az oszlopok szűrőlistáiban álló rendező-utasításokkal virtuálisan rendezhetjük a táblázatot és a művelet után az új sorrend szerinti "első ötven" sort látjuk.

A munkalapok listájának negyedik, *Felhasználó barát név* címkéjű oszlopa a létrehozandó tábla nevének megadására szolgál. Amikor a jelölőnégyzettel kiválasztjuk a munkalapot, akkor a bővítmény ezt a mezőt automatikusan kitölti. A név a tiltott karakterektől megtisztított munkalap-név lesz. A felhasználó ezt a karakterláncot felülírhatja. A tiltott karaktereket a következő táblázat tartalmazza.

.,:;	pont, vessző, kettőspont, pontosvessző
1 `))	aposztrófok, idézőjel
/ \	perjel, függőleges vonal, fordított perjel
!?	felkiáltójel, kérdőjel
& % \$	és jel, százalékjel, dollárjel
+ *	összeadás jel, csillag
= < >	relációs jelek
()[]{}	zárójelek

6. ábra tiltott karakterek a tábla nevében

A Kapcsolódó táblák kijelölése nyomógomb funkcióját később ismertetem. A Befejezés parancsgombra kattintva a segédprogram megkezdi az adatok beolvasását. A művelet fázisait a varázsló Importálás paneljén követhetjük. Ha a művelet végrehajtása nem lehetséges, akkor a varázsló a panel alján hibaüzenetet jelenít meg, ha az importálás megkezdésének nincs akadálya, akkor a varázsló tovább lép és az Importálás parancstáblán követhetjük nyomon a művelet fázisait.

Imp	ortálás leállítá	isa gombra.			
<u>/</u>	A töbt	bi importálás megszakítva	Összes Siker	en: 9 es: 3	Megszakítva: Hiba:
Rés:	zletek: Munkaelem	Allapot		Üzen	et
Ø	Bicske	Sikeresen végrehajtva. 9 240 sor átvitele befej	jeződött.		
Ø	Csurgó	Sikeresen végrehajtva. 18 810 sor átvitele bef	ejeződött.		
Ø	Fonyód	Sikeresen végrehajtva. 9 900 sor átvitele befej	jeződött.		
8	Hatvan	Hiba		Hiba r	észletes adatai
Δ	Kisbér	Megszakítva			6)
Δ	Sarkad	Megszakítva			
	Sárvár	Measzakítva			

7. ábra a segédprogram negyedik panelja

A bővítmény felülről lefelé haladva, egyenként dolgozza fel a kijelölt objektumokat. A név előtt megjelenített pipa, valamint az Állapot oszlopban álló Sikeres végrehajtás szöveg a tábla létrehozását jelenti.

Az adatok beolvasását az Importálás leállítása parancsgombbal szakíthatjuk meg. A leállítás a már létrehozott táblákra nincs hatással. Az importálást végző segédprogramból a Bezárás parancsgombbal léphetünk ki. A létrehozott táblák, az importálás sorrendjében, a PowerPivot-ablakban állnak.

az importálandó munkalap tartalma

A bővítmény a kiválasztott munkalap összes adatát beolvassa. A tábla első rekordja a munkalap második adatot tartalmazó sora, első mezője a munkalap első adatot tartalmazó oszlopa lesz. A bővítmény a táblázat első 255 oszlopát olvassa be, tehát a létrehozott tábla mezőinek száma is legfeljebb 255 lehet. Nézzünk egy példát!



8. ábra az importálandó munkalap tartalma

Munkafüzetünk egyetlen munkalapot tartalmaz. Neve: "raktár". A munkalapon két elnevezett tartomány találunk: "városok" és "alkatrészek". Mindkettő első sorában az oszlopnevek állnak. Az importáló varázsló második panelján kiválasztjuk Az első sor adatai alkossák az oszlopfejléceket jelölőnégyzetet, majd a következő lépésben kijelöljük mind a három felkínált objektumot.

1	F1 💽	F2		F	3		kód		készle	t		kategór	ia						
1	város							788		4	158	м							
2	Vác	20	005. 05. 05					448		6	554	Z		vár	25		dátu	m	
3	Paks	20	006. 06. 06	ι.				333		2	214	К	4	Van	05		uatu	2005 05	05
4	Ózd	20	003. 03. 03							_			1	vac				2005. 05	0. 05.
				1		∦ ko	od 💌	ke	szlet	×.	ka	tegoria	2	Pak	s			2006.06	5. 06.
•	alkatrészek r a	aktár	városok			1	788			458	м		3	Ózd				2003. 03	3. 03.
						2	448			654	z			alkat	részek	ra	ktár	városo	k
						3	333			214	К		<u> </u>						
						alka	trészek ra	ktár	város	ok									

9. ábra az importálással létrehozott táblák

táblák a bővítmény ablakában

Minden tábla külön lapon áll. A tábla nevét a lap fülén olvashatjuk. A táblákat válthatjuk egérrel, rámutatással és kattintással, vagy a Ctrl+Page Up/Page Down gyorsbillentyűkkel. Utóbbi lehetőség csak akkor érhető el, ha a kurzor a táblában áll. Miért hol állhatna máshol? Az ablak alsó részében, az úgynevezett számítási területen. De arról majd később lesz szó.

A rekordok között az Excel programban megszokott módokon navigálhatunk. A bővítmény állapotsorában a *Rekord* felirat után állnak a navigáció eszközei, négy parancsgomb az első, az előző, a következő és az utolsó rekordra ugráshoz, valamint egy mező az aktuális rekord sorszámával és az összes rekord számával. Ez a terület nem csak kijelző, de beviteli eszköz is. A bővítmény a beírt sorszámú rekordra ugrik az Enter leütése után.

Mint láttuk, a bővítmény nem akadályozza meg az üres vagy több adattartományt tartalmazó munkalapok importálását. Az üres munkalap egyetlen, F1 nevű mezőt tartalmazó, rekord nélküli, táblát eredményez. A több adattartományt tartalmazó munkalapból a bővítmény elemzésre alkalmatlan táblát hoz létre. Az üres vagy használhatatlan táblákat a tábla fülének menüjéből kiadott utasítással vagy a *Ctrl+D* billentyűparanccsal törölhetjük. Utóbbi lehetőség csak akkor érhető el, ha a kurzor a táblában áll.

lokális adatkapcsolatok

Az automatikusan létrehozott adatkapcsolatok, a többi PowerPivot objektummal együtt, a munkafüzetben tárolódnak. Ezek az adatkapcsolatok lokálisak, mert csak az őket tartalmazó munkafüzetben használhatók. A lokális adatkapcsolat segítségével újabb objektumokat vonhatunk be az elemzésbe, közösen frissíthetjük az azonos fájlból származó táblákat és módosíthatjuk az áthelyezett forrásállomány elérési útvonalát.

A lokális adatkapcsolatok listáját a bővítmény ablakában a *Kezdőlap, Külső adatok beolvasása, Létező kapcsolatok* utasítással megjeleníthető panel tartalmazza. Először ki kell jelölnünk a használni kívánt adatkapcsolatot, majd a *Megnyitás* gombbal vagy az adatkapcsolatra duplán kattintva jeleníthetjük meg a munkafüzet objektumainak jegyzékét. A *Szerkesztés* parancsgombbal az adatkapcsolat nevét és a munkafüzet elérési útvonalát módosíthatjuk. A *Frissítés* gomb az adatkapcsolat munkafüzetéből származó táblák közös aktualizálására szolgál.

Létező kapcsolat kiválasztása Válasszon ki egy olyan adatforrással kialakított kapcsolatot, amely tartalmazza az importálni kívánt adatokat. Adatforrás-kapcsolat kiválasztása: A PowerPivot adatkapcsolatai Image: Second S	Létező kapcsolatok	?	×					
Adatforrás-kapcsolat kiválasztása: A powerPivot adatkapcsolatai 2018 Data Source = E:\02 két éve.xlsx Image: Source = E:\02 három éve.xlsx Image: Source = E:\02.	Létező kapcsolat kiválasztása Válasszon ki egy olyan adatforrással kialakított kapcsolatot, amely tartalmazza az importálni kívánt adatokat.							
A PowerPivot adatkapcsolatai 2018 Data Source = E:\02 előző év.xlsx 2017 Data Source = E:\02 két éve.xlsx 2016 Data Source = E:\02 három éve.xlsx Vankafüzet-kapcsolatok 2016 Munkafüzet-kapcsolatok 2017 Munkafüzet: 02. adatkapcsolatok.xlsx 2017 Munkafüzet: 02. adatkapcsolatok.xlsx 2017 Munkafüzet: 02. adatkapcsolatok.xlsx 2017 Munkafüzet: 02. adatkapcsolatok.xlsx 2018 Munkafüzet: 02. adatkapcsolatok.xlsx V Togvábbiak tallózása	Adatforrás-kapcsolat kiválasztása:							
2018 Data Source = E:\02 két éve.xlsx 2017 Data Source = E:\02 két éve.xlsx 2016 Data Source = E:\02 három éve.xlsx Munkafüzet-kapcsolatok 2016 Munkafüzet: 02. adatkapcsolatok.xlsx 2017 Munkafüzet: 02. adatkapcsolatok.xlsx 2017 Munkafüzet: 02. adatkapcsolatok.xlsx 2017 Munkafüzet: 02. adatkapcsolatok.xlsx 2017 Munkafüzet: 02. adatkapcsolatok.xlsx 2018 Munkafüzet: 02. adatkapcsolatok.xlsx V Tgvábbiak tallózása Megnyitás Szerkesztés Frissítés Törlés	A PowerPivot adatkapcsolatai		^					
2017 Data Source = E:\02 két éve.xlsx 2016 Data Source = E:\02 három éve.xlsx Munkafüzet-kapcsolatok 2016 Wunkafüzet: 02. adatkapcsolatok.xlsx 2017 Munkafüzet: 02. adatkapcsolatok.xlsx 2017 Munkafüzet: 02. adatkapcsolatok.xlsx 2018 Munkafüzet: 02. adatkapcsolatok.xlsx Y Togvábbiak tallózása	2018 Data Source = E:\02 előző év.xlsx							
2016 Data Source = E:\02 három éve.xlsx Munkafüzet-kapcsolatok 2016 Munkafüzet: 02. adatkapcsolatok.xlsx 2017 Munkafüzet: 02. adatkapcsolatok.xlsx 2018 Munkafüzet: 02. adatkapcsolatok.xlsx Y Togvábbiak tallózása	2017 Data Source = E:\02 két éve.xlsx							
Munkafüzet-kapcsolatok 2016 Munkafüzet: 02. adatkapcsolatok.xlsx 2017 Munkafüzet: 02. adatkapcsolatok.xlsx 2018 Munkafüzet: 02. adatkapcsolatok.xlsx V Továbbiak tallózása Megnyitás Szerkesztés Frissítés Törlés	2016 Data Source = E:\02 három éve.xlsx							
2016 Munkafüzet: 02. adatkapcsolatok.xlsx 2017 Munkafüzet: 02. adatkapcsolatok.xlsx 2018 Munkafüzet: 02. adatkapcsolatok.xlsx Y Zotl8 Munkafüzet: 02. adatkapcsolatok.xlsx Y Zotl8 Munkafüzet: 02. adatkapcsolatok.xlsx Y Zotl8 Y Továbbiak tallózása Megnyitás Szerkesztés Frissítés Törlés	Munkafüzet-kapcsolatok							
2017 Munkafüzet: 02. adatkapcsolatok.xlsx 2018 Munkafüzet: 02. adatkapcsolatok.xlsx Továbbiak tallózása	2016 Munkafüzet: 02. adatkapcsolatok.xlsx							
2018 Munkafüzet: 02. adatkapcsolatok.xlsx Továbbiak tallózása Megnyitás Szerkesztés Frissítés Törlés	2017 Munkafüzet: 02. adatkapcsolatok.xlsx							
Továbbiak tallózása Megnyitás Szerkesztés Frissítés Törlés	2018 Munkafüzet: 02. adatkapcsolatok.xlsx		•					
	Továbbiak tallózása <u>M</u> egnyitás <u>S</u> zerkesztés Frissítés	<u>T</u> örlés						
Bezárás		Bezárás						

10. ábra a bővítmény lokális adatkapcsolatai

Az automatikusan létrehozott adatkapcsolat önálló objektum. A bővítmény még akkor is megőrzi, ha a munkafüzetéből származó valamennyi táblát töröltük. Eltávolítása felhasználói művelettel, a parancstábla Törlés gombjával történik.

a táblák mentése

A bővítmény ablakának munka közbeni bezárása nem jár adatvesztéssel. Az ablak újbóli megnyitását követően a PowerPivot objektumok ismét a rendelkezésünkre állnak. A bővítmény ablakát bezárhatjuk az ablak *Bezárás* nyomógombjával, a *Fájl, Bezárás* utasításával vagy az *Alt+F*4 billentyűparanccsal. A PowerPivot ablakában kiadott Fájl, Mentés vagy Mentés másként utasítások egyenértékűek az Excel ablakból kezdeményezett mentés parancsokkal. A munkafüzetet bezáró utasítás a bővítmény ablakát is bezárja.

univerzális adatkapcsolat létrehozása

Univerzális, azaz bármely munkafüzetből elérhető, adatkapcsolatot a kapcsolat tulajdonságainak fájlba mentésével hozhatunk létre. Vegyük sorra a megvalósítás lépéseit.

Egy tetszőleges munkafüzetben indítsuk el az adatkapcsolat varázslót a program ablakának Adatok, Külső adatok átvétele, Egyéb forrásból, Az adatkapcsolat varázslóból utasításával.

Adatkapcsolat varázsló	?	×
Üdvözli az Adatkapcsolat varázsló		
A varázsló segítségével távoli adatforráshoz kapcsolódhat.		
Milyen típusú adatforráshoz kíván csatlakozni?		
Microsoft SQL Server Microsoft SQL Server Analysis Services Windows Azure Piactér		
Adatosatorna ODBC adatforrásnév (DSNI)		
Microsoft Data Access – OLE DB-szolgáltató Oracle alkalmazásokhoz Egyéb/speciális		
Mégse < Víšsza <u>T</u> ovább >	Befeje	zés

11. ábra az adatkapcsolat varázsló első panelja

A felkínált listájából válasszuk az ODBC *adatforrásnév* (DNS) *adatforrás* elemet és lépjünk *Tovább*. Dupla kattintásos kiválasztással a segédprogram automatikusan megjeleníti a következő panelt.

A varázsló következő panelja két ODBC szoftver tartalmaz. Excel vagy Access? A konkrét fájl a Tovább gombra kattintva, tallózással választhatjuk ki. Ezen a panelon is működik, a dupla kattintásos kiválasztáskor, az automatikus továbblépés.

Adatkapcsolat varázsló			?	×					
Kapcsolódás ODBC-adatforráshoz Jelölje ki azt az ODBC-adatforrást, amelyhez kapcsolódni kíván. ODBC-adatforrások:									
ODBC-adatforrások: Excel Files	Munkafüzet választása				×				
MS Access Database	Adatbázis <u>n</u> eve *.xls* 02 előző év.xlsx 02 két éve.xlsx 02 három éve.xlsx	<u>K</u> önyvtár: e:\temp Protection e:\ Temp		 	DK igse igó asásra				
	Fájl <u>t</u> ípus: Excel fájlok (*.xls*) ▼	<u>M</u> eghajtó:	•	<u>H</u> áló	zat				
	Mégse ≺ <u>V</u> i	ssza <u>T</u> ovább >	Befe	jezés					

12. ábra az adatkapcsolat varázsló második panelja

Az Excel munkafüzet kiválasztása után nyomjuk meg az OK gombot, amely mindkét ablakot bezárja és megjeleníti az adatkapcsolat varázsló harmadik paneljét: Adatbázis és tábla kijelölése.

Adatkapcsolat varázsló						?	×		
Adatbázis és tábla kijelölése									
Jelölje ki a kívánt adatoka	t tartaln	iazó adatbázi	ist, illetve	táblát v	agy kockát.				
Vál <u>a</u> ssza ki a kívánt adatokat	t tartalm	azó adatbázi	st:						
E:\temp\02 előző év.xlsx		`	-						
<u>C</u> satlakozás egy adott táb	olához:								
Név	Leírás	Módosítva	Készült	Típus					
'április eé\$'				TABLE					
'augusztus eé\$'				TABLE					
🔠 'december eé\$'				TABLE					
🔠 'február eé\$'			-	TABLE					
imárcius eé\$				TABLE					
inovember eé\$'				TABLE					
🔠 'október eé\$'				TABLE					
III 'szeptember eé\$' TABLE									
		Mégse	< <u>V</u>	issza	<u>T</u> ovább >	Ве	feje <u>z</u> és		

13. ábra az adatkapcsolat varázsló harmadik panelja

A parancstábla *Csatlakozás egy adott táblához* feliratú vezérlőjével szabályozhatjuk, hogy az adatkapcsolat a listában kiválasztott egyetlen táblát vagy a munkafüzet összes tábláját tartalmazza. Ha az utóbbit szeretnénk, töröljük a jelölőnégyzet pipáját! A *Tovább* gombra kattintva az adatkapcsolatfájl tulajdonságait adhatjuk meg a varázsló utolsó parancstábláján: *Adatkapcsolatfájl mentése és befejezés*.

Adatkapcsolat varázsló	?	\times
Adatkapcsolatfájl mentése és befejezés		
Írja be az új adatkapcsolatfájl nevét és leírását, majd a mentéshez kattintson a Befejezés gombra.		
Fájl <u>n</u> év:		
C:\Users\Lúdas Matyi\Documents\Adatforrások\vevők - 2018.odc	<u>T</u> allóz	ás
☐ Jelszó mentése fájlba Leírás:		
2018 összes vevője, havi bontásban		
Rövid név:		
2018		
Kulcsszavak:		
<u>M</u> indig ezzel a fájllal kísérelje meg az adatfrissítést		
Excel Services: Hitelesítési beállítások		
Mégse < <u>V</u> issza Tovább >	Befe	eje <u>z</u> és

14. ábra a varázsló negyedik, utolsó panelja

Az adatkapcsolat-fájl nevét és helyét tallózással adhatjuk meg. A program az Adatforrások mappát ajánlja fel tárolásra, amely a Dokumentumok könyvtárban áll. Abszolút elérési útvonala: <rendszermegható>: \Users\<felhasználónév>\Dokuments\Adatforrások.

A varázsló utolsó parancstábláján leírást adhatunk meg a kapcsolt objektumról, nevet adhatunk a kapcsolatnak és az adatkapcsolat-fájl egyéb tulajdonságait állíthatjuk be. A *Befejezés* gombra kattintva a program még felajánlja a táblaválasztást, de ezt utasítsuk el a *Mégse* nyomógombbal.

A létrehozott univerzális adatkapcsolat a PowerPivot ablakból nyitható meg. Kattintsunk a Kezdőlap, Külső adatok beolvasása, Létező kapcsolatok vezérlőre, majd a Továbbiak tallózása... gombra, és a megnyíló Adatforrások mappából választhatjuk ki a fájlt.

Létező kapcsolatok	?	×
Létező kapcsolat kiválasztása Válasszon ki egy olyan adatforrással kialakított kapcsolatot, amely tartalmazza az importálni kívánt adatokat.		
Adatforrás-kapcsolat kiválasztása:		
A PowerPivot adatkapcsolatai		
Nem található kapcsolat.		
Helyi kapcsolatok		_
2018 C:\Users\Lúdas Matyï\Documents\Adatforrások\vevők - 2018.odc		
Munkafüzet-kapcsolatok		
Nem található kapcsolat.		

15. ábra az aktivált adatkapcsolat

A tábla importáló varázslót a kapcsolatot kiválasztva a *Megnyitás* paranccsal, vagy dupla kattintással indíthatjuk el. A segédprogram először a kapcsolat leírását tartalmazó, és az összeköttetés ellenőrzését lehetővé tévő panelt jelenít meg. A következő varázsló-panel az importálandó objektumok meghatározásának módjáról dönthetünk. Fogadjuk el a felajánlott *Választás a táblák és nézetek listájáról…* lehetőséget és lépjünk tovább! A következő panelt már ismerjük: *Táblák és nézetek kijelölése*.

Az adatok beolvasása után a kapcsolat átkerül a *Létező kapcsolatok* panel A *PowerPivot adatkapcsolatai* kategoriájába. Végül ne felejtkezzünk el arról a körülményről, hogy a program Adatvédelmi központjának beállításai blokkolhatják az adatkapcsolat-fájl megnyitását!

Létező kapcsolatok	?	×
Létező kapcsolat kiválasztása Válasszon ki egy olyan adatforrással kialakított kapcsolatot, amely tartalmazza az importálni kívánt adatokat.		
Adatforrás-kapcsolat kiválasztása:		
A PowerPivot adatkapcsolatai		_
2018 Initial Catalog = C:\temp\02 előző év.xlsx		
Helyi kapcsolatok		_
Nem található kapcsolat.		

16. ábra a bővítmény által módosított kapcsolati panel

adattípusok

a DAX adattípusai

A homogenizálás eredményeként a tábla egyes mezőiben már azonos adattípusú bejegyzések állnak. A felhasználó módosíthatja, a konvertálási lehetőségek szabta határokon belül, az egyes mezők automatikus adattípusát.

Szöveg (Text)	Egy Unicode-karakterekből álló karakterlánc. Maximális hossza 268 435 456 Unicode-karakter, másként 536 870 912 bájt.
IGAZ/HAMIS (TRUE/FALSE)	IGAZ és HAMIS (TRUE, FALSE) értékek.
Pénznem (Currency)	A pénznem adattípus -922 337 203 685 477,5808 és 922 337 203 685 477,5807 közötti érték, négy tizedesjegy pontossággal. A törtrész számjegyeinek száma legfeljebb négy lehet. HUF pénznem jelölővel.
Egész szám (Whole Number)	Egy -2^63 és 2^63-1 közé eső 64 bites egész szám.
Tizedes tört szám (Decimal Number)	Egy 64 bites valós szám, amelynek értéktartománya [1] negatív érték ese- tén -1,79E+308 és -2,23E-308 között, [2] a nulla és [3] pozitív érték esetén 2,23E-308 és 1,79E+308 között. Az értékes számjegyek száma legfeljebb tizenhét lehet, a további számjegyek nullák.
Dátum (Date)	Dátum időponttal

17. ábra a DAX adattípusai

A bővítmény a táblázat képleteit nem importálja, csak a kifejezések aktuális eredményét olvassa be a táblákba. Ha a forrás-kifejezés értéke módosul, akkor az új adat csak frissítés után fog megjelenni a PowerPivot ablakban.

az adattípus megállapítása

A bővítmény a mező adattípusának meghatározásakor csak a forrás oszlop első nyolc celláját vizsgálja. Az üres cellák az elemzés eredményét nem befolyásolják. A nyolc üres cellával kezdődő oszlop adattípusa Szöveg lesz. A forrás adatait a bővítmény saját adattípusai szerint kategorizálja.

szöveg	karakterlánc aposztróffal beírt szám szövegként formázott szám hibaérték
IGAZ / HAMIS	logikai érték
pénznem	pénznem
tizedes tört szám	egész szám tizedes tört természetes tört százalék normál alakú szám egyéni kóddal formázott szám, pénz és dátum
dátum	dátum időpont



Szöveg adattípusúnak tekinti a bővítmény a forrás oszlopban álló karakterláncokat, aposztróffal beírt számokat, szövegként formázott számokat és a hibaértékeket.

A bővítmény IGAZ/ HAMIS adattípusú mezőt hoz létre a logikai értékeket tartalmazó oszlopnak. Pénzem adattípusú mezőbe kerül minden olyan szám, amely az Excel valamely pénznem jelölőjével van formázva.

Az egész számokat, tizedes törteket, százalékokat, tört számokat és normál alakban felírt számokat tartalmazó oszlop tizedes tört szám adattípusú mezőt eredményez. Az összes egyéni kóddal formázott számot, pénzt és dátumot a PowerPivot szintén tizedes tört számként importálja. A bővítmény még akkor is tizedes tört szám adattípusúra állítja be a mezőt, ha a forrás oszlop egyetlen törtszámot sem tartalmaz.

A dátumokat és az időpontokat a PowerPivot dátum adattípusú mezőbe olvassa be.

A mező adattípusa az első nyolc cella leggyakoribb adattípusa lesz. Ha két adattípus egyenlő számban fordul elő, akkor az adattípusok erősorrendje dönt: Szöveg > IGAZ/HAMIS > Pénznem > Tizedes tört szám > Dátum.

a mező homogenizálása

Miután a bővítmény az első nyolc cella tartalma alapján meghatározta a létrehozandó mező adattípusát, az oszlop további adatait ennek megfelelően kell átalakítania. Tekintsük át, hogyan egységesíti a bővítmény az adattípusnak megfelelően a mező bejegyzéseit.

Szöveg adattípusú mező: minden adat eredeti formájában beolvasásra kerül, beleértve a hibaértékeket is. Egyetlen kivétel a nem vezérlőpult szerinti formátumú pénznem, amely pénznem jelölő nélkül importálódik. A karakterláncok a mezőben balra igazítva jelennek meg. A karakterláncot megelőző és az azt követő szóközök nem kerülnek importálásra.

IGAZ/HAMIS adattípusú mező: a pénzek, a számok és a dátumok IGAZ értékre konvertálódnak. A szövegek és a hibaértékek nem kerülnek importálásra. A logikai értékek a mezőben középre igazítva jelennek meg.

Pénznem adattípusú mező: a pénzek, a számok és a dátumok, beleértve az időpontokat is, négy tizedesjegyre kerekítve, a vezérlőpult szerinti pénznem formátumban kerülnek beolvasásra. A szövegek, a logikai értékek és a hibaértékek nem kerülnek importálásra. A pénzek a mezőben jobbra igazítva jelennek meg.

Tizedes tört szám adattípusú mező: a számok, a pénzek, a dátumok és időpontok csonkolás nélkül, számként kerülnek beolvasásra. A szövegek, a logikai értékek és hibaértékek nem kerülnek importálásra. A mezőben a számok jobbra igazítva jelennek meg.

Dátum adattípusú mező: a pénzek, a számok, a dátumok és az időpontok <rövid dátum><szóköz><hosszú idő> formátumban fognak megjelenni. A szövegek, a logikai értékek és hibaértékek nem kerülnek importálásra. A dátumok a mezőben jobbra igazítva jelennek meg.

eredeti adat	Szöveg	IGAZ/HAMIS	Pénznem	Tizedes tört szám	Dátum
papír	papír				
HAMIS	HAMIS	HAMIS			
3,145873 Ft	3,15 Ft	IGAZ	3,15 HUF	3,145873	1900-01-03 3:30:03
70,70 USD	70,70	IGAZ	70,70 HUF	70,7	1900-03-10 16:48:00
13	13	IGAZ	13,00 HUF	13	1900-01-13 0:00:00
13,54789	13,54789	IGAZ	13,55 HUF	13,54789	1900-01-13 13:08:57
1955.05.05	1955.05.05	IGAZ	20 214,00 HUF	20214	1955-05-05 0:00:00
13:50	13:50	IGAZ	0,58 HUF	0,576388888	1899-12-30 13:50:00
#ZÉRÓOSZTÓ!	#DIV/o!				
#NÉV?	#NAME?				

19. ábra

különböző cellatartalmak importálása meghatározott adattípusú mezőbe

A bővítmény a háttérben, a programhoz hasonlóan, a dátumokat sorszámként, az időpontokat nulla és egy közé eső törtszámként kezeli. A dátum adattípusú mezőbe beolvasott nulla 1900.01.01.

dátummá konvertálódik. A bővítmény kezelni tudja a történelmi dátumokat is, ezért a dátum adattípusú mezőbe importált negatív számok is dátummá lesznek átalakítva.

A mínusz egyet a PowerPivot 1899.12.30. dátumra konvertálja. Tehát az 1899.12.31. munkafüzetből származó adatok konvertálásakor nem fordulhat elő. Az egész rész nélküli tizedes törteket a bővítmény időpontra alakítja át, majd az 1899.12.30. dátummal egészíti ki. A legkorábbi képezhető dátum az időszámításunk kezdete, tehát 0001.01.01., ami -693 593-nak felel meg.



20. ábra számok importálása dátum adattípusú mezőbe

Az importálással létrehozott új tábla üres rekordokat is tartalmazhat. Ennek oka a beolvasott táblázat végén álló sorok korábbi törlése. A hiba csak akkor jelentkezik, ha az eltávolítás a *Tartalom törlése* utasítással vagy a *Del* billentyűvel történt. Mivel a PowerPivot ablakban rekordot nem törölhetünk, ezért a hibát a programban vagy az importálás során kell orvosolnunk. Töröljük a táblát! Hozzunk létre a forrásból adatbázis-táblázatot és ennek adatait olvastassuk be a bővítménnyel! A másik lehetőség a táblázat sorainak szűrése az importálás meghatározott fázisában.

1	Α	В	С				
1	betűk	számok	színek			-	
2	С	5	sárga		A	В	C
3	С	3	barna	1	betük	számok	szinek
4	н	1	fehér	2	С	5	sárga
5	1	1	cárna	3	С	3	barna
-	-	-	saiga	4	н	1	fehér
0	F	9	Saiga	5	1	1	sárga
/	E	6	zold	6	F	9	sárga
8	G	7	kék	7	F	6	zöld
9	F	9	barna	0	c c	7	kák
10	G	2	barna	0		,	NEN
11	В	3	kék	9	7	9	barna
12	н	1	fehér	10	G	2	barna
13	F	3	fehér	11			
14	- D	0	harna	12			
14	5	v	oania	13			
				14			

	betűk 💌	számok 🔽	színek 🔽
1	С	5	sárga
2	С	3	barna
3	н	1	fehér
4	1	1	sárga
5	F	9	sárga
6	E	6	zöld
7	G	7	kék
8	F	9	barna
9	G	2	barna
10			
11			
12			
13			

21. ábra a tartalom törlése művelet következtében kialakuló üres rekordok

szelektív importálás

a villámnézet ablaka

A Tábla importálása varázsló eszközeivel nemcsak a munkafüzet importálandó lapjait választhatjuk ki, de az egyes táblázatok oszlopait és sorait is szelektálhatjuk. A beolvasandó táblázat-elemek körének meghatározására a varázsló harmadik, Táblák és nézetek kijelölése feliratú panelján nyílik lehetőség. A szelektálás a Villámnézet és szűrés nyomógombbal megjeleníthető felületen történik.

bla	importálása varázsló			?	×
i vál A a:	asztott tábba villámnézete jelölőnégyzet segítségével v z oszlop legördülő listájának	válassza ki a kívánt oszlopo nyíl gombját, és válassza k	kat. Egy oszlop adatainak sz i a szűrésben szerepeltetni k	űréséhez használja cívánt értékeket.	
Táb	lázat neve: javítások\$				
V	🗹 munkatárs neve 🔽	🗹 születési dátum 🛛 🔽	🗹 személyi szám 🛛 🔽	🗹 anyja neve 🛛 🔹	1
5	Pados Patrícia	1986. 06. 09. 0:00:00	28606097261	Csányi Erzsébet	
6	Szigetvári Domonkos	1978. 04. 14. 0:00:00	17804143373	Gosztonyi Erzsébet	
7	Keszler Tódor	1980. 02. 04. 0:00:00	180020443 R	ekord: 5., összesen 50	
8	Márkus Ferenc	1983. 04. 08. 0:00:00	18304073183	Sasvári Dóra	Ŭ
9	Méhes Viktória	1982. 11. 03. 0:00:00	28211039837	Fitos Matild	
10	Gyulai Márkus	1993. 02. 03. 0:00:00	19302035874	Somogyi Renáta	
11	Petrás Vince	1996. 08. 04. 0:00:00	19608048545	Engi Emilia	
12	Bódi Laura	1983. 11. 25. 0:00:00	28311252056	Gyurkovics Zsófia	
<				:	>
So	rs <u>z</u> űrők törlése			OK Mégs	e

22. ábra a villámnézet ablaka

A forrás-táblázat első ötven rekordját mutató villámnézet panelja nem rendelkezik teljes méret nyomógombbal, sőt az ablak bal felső sarkának helyi menüjében is elérhetetlen ez a parancs. Az ablak nagyítása csak a szegélyek húzásával lehetséges. A navigálás az Excel ablakban megszokott eszközökkel történik. Egyedül a vízszintes lapozás Alt+Page Up/Page Down billentyűparancsa hiányzik. Kárpótlásul, kapjuk, a CTRL+egérgörgő funkciót, amely más Microsoft szoftverekben a nagyítás eszköze, itt viszont a kép vízszintes mozgatója.

az oszlopok szelektálása

Az oszlopok kijelölése az oszlopnevek felett megjelenő, lefelé mutató fekete nyíllal történik. Az öszszes oszlopot a Ctrl+A billentyűparanccsal jelölhetjük ki.

Az oszlopok szélességét megadhatjuk hozzávetőlegesen, az oszlopnevek jobb oldali szegélyének húzásával, vagy pontosan az oszlopnév menüjének parancsával. Az alkalmazott mértékegység a képpont. Az oszlopnév jobb oldali szegélyére duplán kattintva a bővítmény a legszélesebb tartalomhoz igazítja az oszlop szélességét.

A kijelölt oszlop nevén az egérmutató négyirányú nyíllá alakul át, jelezve az oszlop áthelyezhetőségét. Tehát a villámnézet nemcsak a sorok egyéni rendezését, de az oszlopok felhasználói sorrendjének kialakítását is lehetővé teszi. A felsorolt lehetőségek csak az adatok jó megjelenítését szolgálják, magára az importálásra és a tábla-készítésre nincsenek befolyással.

Az importálandó oszlopok kiválasztása az oszlopnevek előtt álló jelölőnégyzettel történik. A kipipált oszlopok a táblába kerülnek. Az aktuális oszlop jelölőnégyzetét a Ctrl+szóköz billentyűvel is vezérelhetjük. A panel bal felső sarkában álló jelölőnégyzet az összes oszlop pipáját törli, illetve az összes oszlopot kipipálja. A vezérlőben pici szürke négyzet jelenik meg, ha egy vagy több oszlopot kihagyunk az importálásból.

a sorok szelektálása

A villámnézet szűrés-arzenálja szerényebb, mint az Excelé. Az oszlopnevek cellájából lenyitható szűrőlisták felépítése, a forrás adattípustól függetlenül azonos: három rendező utasítás, az oszlop szűrőjének törlője, az oszlop adattípusának megfelelő reláció-lista és a gyorsszűrő az oszlop egyedi bejegyzéseivel.

A Szövegszűrők és a Számszűrők feliratú relációs-listák elemeire kattintva az Egyéni szűrők parancstábla jelenítik meg, ahol legfeljebb két szűrőfeltételt határozhatunk meg. A feltételek logikai ÉS vagy logikai VAGY viszonyban állhatnak egymással. Logikai ÉS viszonyban mindkét feltételnek, a VAGY viszonyban legalább az egyik feltételnek teljesülnie kell az importáláshoz.

A szövegszűrő karakterláncában nem csak konkrét karakterek, de karakter-csoportot meghatározó kód is állhat.

[a-z]	egy betű
[0-9]	egy szám
[a-z;0-9]	egy karakter
[d-n]	egy betű az ABC megadott tartományából
[3-7]	egy szám a számsor megadott tartományából
[aou]	egy betű a felsoroltak közül
[159]	egy szám a felsoroltak közül
[!d-n]	betűk kizárása
[!3-7]	számok kizárása

23. ábra

a szöveg adattípusú oszlop szűrőjében használható, karakter-csoportot meghatározó kódok

A lényeget összefoglalva. [1] A karakter-csoportot meghatározó kód szögletes zárójelek között áll. [2] A felkiáltójel a kizárás jele. [3] Egy karakterre több deklaráció is megadható. A meghatározásokat pontosvesszővel kell elválasztanunk. [4] Karakter-definícióként a szögletes zárójelet szögletes zárójelek között kell begépelnünk.

A szűrőlista alján olvasható Nem minden elem látható felirat azt jelzi, hogy az egyedi értékek magas száma miatt a bővítmény csak az első ezeregy egyedi bejegyzést jelenítette meg a gyorsszűrő listájában.

999		
✓ 1000		
✓ 1001		
A Nemmino	len el em látható	
		A DESCRIPTION AND A REPORT OF THE

24. ábra a gyorsszűrő alatt megjelenő figyelmeztetés

A villámnézet ablakában a kijelöléses szűrés funkció is használható. A rámutatással és kattintással kiválasztott cella tartalmával azonos bejegyzések sorait tudjuk leválgattatni a cella menüjének Szűrő, Szűrés a kijelölt cellaérték alapján parancsával.

ibla i vál A h é	importálása varázsló l asztott tábla villámn i jelölőnégyzet segítsé asználja az oszlop lej rtékeket.	ázete śgével válassza ki a kív gördülő listájának nyíl g	vánt oszlopokat. gombját, és vála:	Egy os ssza k	szlop adatainak szűrés i a szűrésben szerepe	? séhez Itetni kívánt	×
Táb	olázat neve: javításo	k\$					
 ✓ 	🔻 rendszám 🛛 💽	🗹 típus 🛛 🔽	🗹 szervizből		🗹 számla 🛛 🔽	🗹 sajáthibás 🔽	^
1	MPI-909	Opel Astra	2007. 07. 24. 0:	00:00	244900	IGAZ	
2	BHQ-472	Chovrolat Cruza	2007 07 24 0	00.00	390200	HAMIS	
3	UCU-59 Szűrő		•	×	Szűrő törlése a követk	ezőből: rendszám	
4	UYV-076	Opel Antara	2007. 07. 16. 0:		Szűrés a kijelölt cellaé	rték alapján	
5	IBY-325	Citroen C4	2007. 08. 03. 03	00:00	481900	HAMIS	
6	HYZ-268	Chevrolet Cruze	2007. 08. 11. 0:	00:00	983800	HAMIS	
7	OGH-586	Peugeot 508	2007. 08. 01. 0:	00:00	246700	HAMIS	
<						>	
So	orszűrők törlése					OK Mégse	

25. ábra kijelöléses szűrés a villámnézet ablakában

Egy adott oszlop szűrőjét törölhetjük az oszlop szűrőlistájában álló vagy az oszlop tetszőleges cellájában megjeleníthető menü utasításával. A villámnézet ablakának bal alsó sarkában álló Sorszűrök törlése gombbal a táblázat összes beállított szűrője eltávolítható.

A sorok szelektálásának feltételeit jóváhagyva, a Táblák és nézetek kijelölése panelen az Alkalmazott szűrők linkre kattintva, megtekinthetjük az importálás teljes deklarációját. A megjelenített szöveg nem szerkeszthető!

ájlnév	: E:\temp\feladatok\04 javítás	sok.xlsx	
áblák (és <u>n</u> ézetek:		
	Forrástábla	Felhasználóbarát név	Szűrő részletei
⊿ ⊞	javítások\$	javítások	Alkalmazott szűrők
	Szürő részletei Kiválasztott oszlopok: mi rendszám, szervizből, szi Alkalmazott szűrő: (([oszt OR (losztálv] = 'számvite	unkatárs neve, személyi szám, o ámla iály] = 'ellenőrzési') OR ([osztály] li')) AND (([típus] = 'Audi A4') OF	<pre>? X sztály, típus, = 'pénzügyi') t ([típus] =</pre>

26. ábra az importálás deklarációja

Az adatok beolvasása után, a PowerPivot ablakban a szűrök módosíthatók és az importálásból kizárt rekordok utólagosan is hozzáadhatók a táblához.

szövegfájl importálása

fogalmak

A táblázatos adatkezelő programok leggyakoribb adatcsere-formátuma a formázást nem tartalmazó szövegfájl. Kiterjesztése "txt". Tehát, van két programunk, amelyek közvetlenül nem tudják egymás adatait olvasni, ezért az egyik program a "táblázatát" txt-be exportálja, majd a másik program a txt-t importálással "táblázattá" alakítja vissza. A szövegfájl minden sora azonos számú adatot tartalmaz meghatározott karakterrel elválasztva, vagy szóközökkel valóságos oszlopokat képezve. A szövegfájl tartalmazhatja az oszlopneveket is. A bővítmény csak a karakterrel elkülönített adatokat tartalmazó állományok importálására alkalmas, a szóközökkel oszlopokba rendezett típust nem tudja kezelni.

beolvasás

A szövegfájl beolvasását is a bővítmény ablakából kezdeményezzük: Kezdőlap, Külső adatok beolvasása, Más forrásból, Szövegfájlok, Szövegfájl.

Az importáló varázsló szövegfájl modulja egyetlen lapon tartalmazza a kapcsolat elnevezésére, a beolvasandó állomány meghatározására, illetve a rekordok szelektálására szolgáló vezérlőket. A fájl kiválasztása a szokásos *Tallózás…* funkcióval történik. Az *Oszlopelválasztó* listából kell kiválasztanunk a text-fájlban alkalmazott szeparáló karaktert. Ha erről nincs információnk, akkor egyesével kell kipróbálnunk a lista elemeit és figyelni a villámnézetet, melyiknél kapunk azonos típusú adatokkal feltöltött oszlopokat. Amennyiben a forrás tartalmazza a oszlop-neveket, akkor jelöljük be az *Első sor elemeinek…* jelölőnégyzetet.

Tábla importá	ilása varázsló					?	Х
Kapcsolódá Adja meg	s strukturálatlan f az adatok struktu	lájlhoz ırálatlan fájlból való beolvas	ásához szükséges a	adatokat.			
Kap <u>c</u> solat fe	elhasználóbarát n	eve: adattípus meghatároza	ás				
<u>F</u> ájl elérési	útja:	E:\temp\05 adatt ipus	meghatározás.txt			Talló <u>z</u> ás	
<u>O</u> szlopelvá	lasztó:	Tabulátor (t)			\sim	<u>S</u> peciáli	s
✓ Első sor	elemeinek haszn	álata oszlopfejlécként					
🔽 🗹 ind	ex 🔽 🗹 szi	öveg1 🔽 🗹 szöveg2 📘	🛛 🗹 szám1 💽	🗹 szám2 🔽	🔽 pér	z 💌	^
1	1	papucs	123	654		123	
2	2	papucs	123	654		123	
3	3	papucs	123	654		123	
4	4	PROUCS	123			123	

27. ábra szövegfájl importálásakor megjelenített panel

Ha a villámnézetben a szövegek helytelenül jelennek meg, vagy a bővítmény egyetlen oszlopot sem ismer fel, akkor a *Speciális* gombra kattintva megadhatjuk a szövegfájl létrehozásakor alkalmazott, ország-függő kódolási rendszert. Ha nem ismerjük az állomány létrehozásának körülményeit, akkor sajnos csak a "próba-szerencse" módszert tudjuk alkalmazni.

Az oszlopok és a sorok szelektálása a munkafüzet importálásakor megismert módon és lehetőségekkel történik. A létrehozott tábla automatikusan a szövegfájl nevét kapja.

sajátosságok

Amennyiben van befolyásunk az importálandó szövegfájl típusára, akkor a tabulátorral tagolt formátumot válasszuk! Ez a típus okozza importáláskor a legkevesebb problémát. A továbbiakban csak az ebben a formátumban álló adatok beolvasását ismertetem. Természetesen a megállapítások nagy része más típusú szövegfájlra is érvényes.

Az adattípus megállapításánál a bővítmény az oszlop első huszonöt adatát elemzi. Megkülönböztet szöveget, egész és tört számot valamint dátumot. Az elemzéskor a logikai értékek szövegnek, a pénzek egész vagy tört számnak, az időpontok dátumnak minősülnek. A mező adattípusa az első huszonöt bejegyzésben leggyakrabban előforduló típus lesz. Ha két adattípus egyenlő számban fordul elő, akkor az adattípusok rangsora dönt: szám > dátum > szöveg. A hiányzó bejegyzések az adattípus-meghatározást nem befolyásolják. Az első huszonöt helyen adatot nem tartalmazó oszlop adattípusát a bővítmény Szöveg típusúra állítja be. Ha az oszlop első huszonöt számbejegyzése közül akár egy is tört szám, akkor a mező adattípusa Tizedes tört szám, különben Egész szám adattípusú lesz.

A csak TRUE és FALSE bejegyzéseket tartalmazó, szöveg adattípusú mező adattípusát IGAZ/ HAMIS adattípusra, a számként beolvasott adatokat Pénznem típusra állíthatjuk át a menüszalag Kezdőlap, Formátum csoportjának Adattípus listájával.

inde	ex	Egész szám	Tizedes tört szám	Dátum	Szöveg
1		5555	55,55	2005.05.05	ötösök
2		5555	55,55	2005.05.05	ötösök
3	~ _	5555	55,55	2005.05.05	ötösök
24	\sim	2225	55,55	2005.05.05	ötösök
25		5555	55,55	2005.05.05	ötösök
26		55,55	5555	5555	5555
27		2005.05.05	2005.05.05	55,55	55,55
28		ötör	ötör	ötör	2005 05 05
		0103	0105	0105	2003.03.03
	index 🔽	v egész szám ▼	vitis ✓ tizedes tört szám ▼	v dátum	v v szöveg v
22	index 🔽 22	vitos ✓ egész szám 5555	vius ✓ tizedes tört szám 55,55	√ dátum 2005. 05. 05. 0:0	v szöveg v 10:00 ötösök
22 23	index 22 23	✓ egész szám ▼ 5555 5555	vitos ✓ tizedes tört szám 55,55 55,55	√ dátum 2005. 05. 05. 0:0 2005. 05. 05. 0:0	 ✓ szöveg ▼ 10:00 ötösök 10:00 ötösök
22 23 24	index 22 23 24	 ✓ egész szám ▼ 5555 5555 	vtis ▼ tizedes tört szám 55,55 55,55 55,55	 ✓ dátum 2005. 05. 05. 05. 0:0 2005. 05. 05. 0:0 2005. 05. 05. 0:0 	 ✓ szöveg ▼ 0:00 ötösök 000 ötösök
22 23 24 25	index 22 23 24 25	 ✓ egész szám ▼ 5555 5555 5555 5555 	vtis ▼ tizedes tört szám 55,55 55,55 55,55 55,55	 ✓ dátum 2005. 05. 05. 05. 0:0 2005. 05. 05. 0:0 2005. 05. 05. 0:0 2005. 05. 05. 0:0 	 ▼ ✓ szöveg ▼ 0:00 ötösök 000 ötösök 0:00 ötösök
 ✓ ✓ ✓ 22 23 24 25 26 	index 22 23 24 25 26	v egész szám	vtos ▼ tizedes tört szám 55,55 55,55 55,55 55,55 55,55 55,55	 ✓ dátum 2005. 05. 05. 05. 0:0 2005. 05. 05. 0:0 2005. 05. 05. 0:0 2005. 05. 05. 0:0 	▼ szöveg ▼ 0:00 ötösök 0:00 ötösök 0:00 ötösök 0:00 ötösök 0:00 ötösök 5555 5555
 ✓ ✓ ✓ 22 23 24 25 26 27 	index 22 23 24 25 26 27	 ✓ egész szám ▼ 5555 5555 5555 5555 5555 5555 	 ✓ tizedes tört szám ▼ 55,55 55,55 55,55 55,55 55,55 55,55 	 ✓ dátum 2005. 05. 05. 05. 0:0 2005. 05. 05. 0:0 2005. 05. 05. 0:0 2005. 05. 05. 0:0 	▼ szöveg ▼ szöveg 0:00 ötösök 0:00 ötösök 0:00 ötösök 0:00 ötösök 5555 55,55

28. ábra adatok beolvasása különböző adattípusú mezőkbe (felül a forrás, alul a tábla)

A mező adattípusának meghatározása után a beolvasandó adatok homogenizálására kerül sor. Az ábra tanulságai szerint, [1] az Egész szám adattípusú mezőbe a bővítmény a tört számot egészre csonkolva olvassa be. A dátumot és a szöveget nem importálja. [2] A Tizedes tört szám adattípusú mezőbe csak egész számot tudunk beolvastatni, dátumot és szöveget nem. [3] A Dátum adattípusú mezőbe csak dátum kerülhet. [4] A Szöveg adattípusú mezőben a másik három adattípus szövegként lesz elhelyezve.

beolvasás vágólapról

fogalmak

Az importálásos tábla készítés első lépése a kopírozás: a bővítmény az elemzendő adatokról másolatot készít a számítógép ideiglenes tárjába. Ugyanezt a műveletet hajtja végre a program is, amikor kijelölünk egy táblázatot és kiadjuk a *Másolás* parancsot. Tehát nem meglepő, hogy a bővítmény a "vágólapon" álló adatokból is képes táblát készíteni.

Vegyük sorra a PowerPivot vágólap-műveleteit. [1] *Beillesztés új táblázatba*: a művelet egy új táblát hoz létre a vágólapon lévő adatokkal. [2] *Beillesztés hozzáfűzéssel*: a művelet a vágólapon lévő adatokat az aktív táblába helyezi, a már ott lévő rekordok után. [3] *Beillesztés cserével*: a művelet törli az aktív tábla rekordjait, majd az üres táblába helyezi a vágólapon lévő adatokat. [4] *Másolás*: a művelet vágólapra helyezi a kijelölt objektumot.

A PowerPivot ablakban nincs többes kijelölés, értsd: nem szomszédos területeket nem tudunk kijelölni. Az importálással létrehozott tábláknál a beillesztés hozzáfűzéssel és a beillesztés cserével utasítások nem elérhetők.



29. ábra a vágólap-műveletek vezérlői és leírásai

A vágólapos táblakészítés elsősorban különböző sor-számú, de azonos nevű oszlopokat, azonos sorrendben tartalmazó táblázatok, egyszerűen azonos felépítésű táblázatok egyesítésére alkalmazható. Másszóval a forrás-táblázatok adatait egyetlen táblába másoljuk, majd azt adatbázisba ágyazottan vagy önállóan elemezzük. Ebből következően a vágólap műveletekkel létrehozott táblát, általánosítva egyesítő táblának is nevezhetjük. Az egyesítő tábla és forrás-táblázatai tetszőleges munkafüzetben állhatnak. A vágólappal létrehozott tábla nem frissíthető!

az egyesítő tábla létrehozása

Az egyesítő tábla létrehozása egyet jelent az első forrás-táblázat beolvasásával. Másoljuk a táblázatot a vágólapra, majd adjuk ki beillesztés parancsot a bővítmény ablakában, amelynek hatására a PowerPivot megjeleníti a beillesztendő táblázat tizenkilenc soros villámnézetét.

A parancstáblán megadhatjuk a tábla nevét és jelölőnégyzettel deklarálhatjuk az első sorban álló oszlop-neveket. A beillesztés villámnézete nem tartalmazza a szűrés eszközeit, tehát a forrás-táblázatok rekordjait, ha szükséges, akkor a vágólapra helyezés előtt kell szelektálnunk.

Azonos adattípusú bejegyzéseket tartalmazó oszlopok esetén a bővítmény: [1] megkülönbözteti az egész számokat- és a tizedes törteket és ennek megfelelően állítja be a mező adattípusát, [2] a pénznem típusú forrás-adatokat négy tizedesjegyre kerekíti, [3] a dátumokat a o:oo:oo időponttal egészíti ki, [4] az időpontok elé a 1899.12.31. dátumot illeszti, és dátum adattípusú mezőt hoz létre számukra.

A vegyes adattípusú forrás-oszlop számára a bővítmény szöveg adattípusú mezőt hoz létre, ha abban akár csak egyetlen szöveg/logikai/dátum adattípusú adat áll. Ha a felsorolt három adattípus hiányzik a vegyes adattípusú oszlopból, akkor a mező adattípusa pénznem lesz, ha akár csak egyetlen adata pénznem. A számokat tartalmazó forrás-oszlop számára a bővítmény tizedes tört adattípusú mezőt hoz létre, ha akár csak egyetlen adata tizedes tört.

A PowerPivot a fenti szabályoknak megfelelően létrehozza a táblát, de ha valamelyik mezőjében az egyik adattípus bejegyzéseinek száma eléri a rekordok számának 95 százalékát, felfelé, egész százalékra kerekítve, akkor a bővítmény felajánlja egy számított mező automatikus létrehozását, amely az eredeti mező homogenizált bejegyzéseit tartalmazza.

=KEREK.LE(rekordok száma * 95% ; 0) <= azonos adattípusú bejegyzések száma
=KEREK.FEL(azonos adattípusú bejegyzések száma / rekordok száma ; 2) >= 95%

30. ábra homogenizáló mező felajánlásának egyetlen feltétele két különböző megfogalmazásban

A szolgáltatás nem terjed ki a döntően szöveg- és a többségében pénznem bejegyzéseket tartalmazó mezőkre. Az automatikus homogenizáló oszlop létrehozásának lehetőségére a mezőnév mellett álló jelzés figyelmeztet.

	logikai 🌼 🔽	egész 🛛 🔶 🔽	tört 🛛 🔶 🔽	dátum 🔶 🔽	idő 🛛 🔶 🔽
9	false Tov	ábbi információért katt	intson az oszlop bárme	lyik cellájára. :00:00	1899. 12. 31. 19:13:52
10	false	-38	-38.940194594391031	2004. 06. 28. 0:00:00	1899. 12. 31. 16:09:20
11	false	97	97.6715220258331	2001. 01. 12. 0:00:00	1899. 12. 31. 15:21:37
12	true	72	72.393020295397008	2007. 11. 13. 0:00:00	1899. 12. 31. 21:43:17
13	true	🔶 (Ctrl) 🔨	37.279309370113467	2004. 04. 29. 0:00:00	1899. 12. 31. 17:30:28
14	true	Számított oszloj	hozzáadása és logikai	adattípusúra konvertál	ása 1. 2:58:20
15	true	Az oszlop megt	artása szöveges adattíp	usúként	1. 7:59:55
16	false	55	55.069027978014461	2002. 02. 13. 0:00:00	1899. 12. 31. 9:27:43
17	true	-96	-96.494780655974324	2003. 12. 08. 0:00:00	1899. 12. 31. 20:16:54
18	true	-88	-88.856946562051874	2009. 01. 24. 0:00:00	1899. 12. 31. 15:02:15
19	false	89	89.319920835200946	2008. 04. 19. 0:00:00	1899. 12. 31. 16:44:36
20	true	27	-37.0100788300278	2000. 04. 08 0:00:00	1890 12 31. 0:02.11

31. ábra

a mezőnév mellett álló és a mező egy bejegyzésére kattintva megjelenő vezérlő

A homogenizáló mező létrehozását a jelzett mező egy bejegyzésére-, majd a bejegyzés mellett megjelenő, Ctrl feliratra kattintva, a Számított oszlop hozzáadása és logikai/numerikus/dátum adattípusúra konvertálása paranccsal kezdeményezhetjük.

átalakítandó	a többségben lévő adattípus							
adattípus	logikai	egész	tizedestört	dátum	idő			
szöveg	üres	üres	üres	üres	üres			
logikai		üres	üres	üres	üres			
pénznem	<>0 IGAZ 0 HAMIS	egészre lefelé kerekítve	azonos az eredetivel	üres	üres			
egész	<>0 IGAZ 0 HAMIS		azonos az eredetivel	üres	üres			
tizedestört	<>0 IGAZ 0 HAMIS	egészre lefelé kerekítve		üres	üres			
dátum	üres	dátum számértéke	dátum számértéke		o:oo:oo időponttal			
idő	üres	1	1 + az idő számértéke	1899.12.31. dátummal				

32. ábra a homogenizáló számított oszlop értékei az egyes adattípusok függvényében

A vegyes adattípusú, döntően idő bejegyzéseket (ó:pp:mm) tartalmazó mező automatikus homogenizáló képlete hibás! A képletet a mező tetszőleges bejegyzésére kattintva, a szerkesztőlécen javíthatjuk: =IF(NOT(ISERROR(TIMEVALUE([mezőnév]))) ; TIMEVALUE([mezőnév])). A többségében dátum-idő (éééé. hh. nn. ó:pp:mm) bejegyzéseket tartalmazó mező homogenizálására a bővítmény képlete nem alkalmas! A minden dátum-tartalomra használható képlet ez: =IF(NOT(IS-ERROR(DATEVALUE([mezőnév]))) ; DATEVALUE([mezőnév]) + TIMEVALUE([mezőnév])).

A pénznem és tizedes tört adattípusú bejegyzéseket az INT függvény, a számegyenesen balra elmozdulva, egészre alakítja át. Például a tizenkettő egész három (12,3) tizenkettő (12), a mínusz tizenkettő egész három (-12,3) mínusz tizenhárom (-13) lesz.

A homogenizáló számított mező adattípusát a bővítmény a Kezdőlap, Formátum, Adattípus: felirat után jeleníti meg "Automatikus (adattípus neve)" formában.

a táblázat előkészítése

Az oszlopok homogenizálását a táblakészítés előtt is elvégezhetjük. Az egyik megoldás lehet, egy segédoszlop alkalmazása, amelyben a kívánt adattípustól eltérő cellatartalmakat képlettel töröljük. A művelethez az Excel típusellenőrző függvényeit használhatjuk. A szöveg-, a logikai-, a pénznem-, az egész-, és a tizedes tört adattípusok detektálása ezekkel a függvényekkel nem okoz problémát, de a dátumok és az időpontok azonosítására alkalmas CELLA függvény, magyar felhasználói környezetben, erre a feladatra, csak erősen korlátozott mértékben alkalmas.

szöveg	=HA(SZÖVEG.E(A2) ; A2 ; "")
logikai	=HA(LOGIKAI(B2); B2 ; "")
pénznem	=HA(HAHIBA(NEM(SZÖVEG.KERES("c" ; CELLA("forma" ; A2)) > 0) ; IGAZ) ; "" ; A2)
egész szám	=HA(VAGY(SZÖVEG.E(A2) ; LOGIKAI(A2) ; ÜRES(A2) ; HAHIBA(MARADÉK(A2 ; 1) <> 0 ; HAMIS)) ; "" ; A2)
tizedes tört	=HA(VAGY(SZÖVEG.E(A2) ; LOGIKAI(A2) ; ÜRES(A2) ; HAHIBA(SZÖVEG.KERES("c" ; CELLA("forma" ; A2)) > 0 ; HAMIS) ; HAHIBA(MARADÉK(A2 ; 1) = 0 ; HAMIS)) ; "" ; A2)

33. ábra a kívánt adattípustól eltérő cellatartalmak törlése képlettel

A képletek nem tökéletesek, mert a fent ismertetett probléma miatt, [1] az egész számok között a dátumok, [2] a tizedes törtek között az időpontok is megjelennek.

rekordok hozzáadása az egyesítő táblához

A Kezdőlap, Vágólap, Beillesztés hozzáfűzéssel utasítással újabb rekordokat másolhatunk az elkészült táblába. Amennyiben a vágólapon álló táblázat oszlopainak száma azonos a kibővítendő tábla mezőszámával, akkor a bővítmény megjeleníti a Beillesztés villámnézete parancstáblát. A panel tartalmazza a bővítendő tábla első húsz rekordját (felül) és a hozzáfűzni szándékolt táblázat első tizenkilenc sorát (alul). A vágólapon lévő adatok első sorában álló mezőneveket a két listát követő jelölőnégyzettel deklarálhatjuk. A tábla rekordjait megjelenítő területrész magasságát a terület alsó szegélye alatti, láthatatlan vezérlővel szabályozhatjuk. Az eszközt csak az átalakuló egér-mutató jelzi.

Az új tábla létrehozásakor és az újabb rekordok hozzáadásakor megjelenített parancstáblák különbözőek, de feliratuk azonos: *Beillesztés villámnézete*. Megnyitáskor a panel alján megjelenített üzenettől, "Típuseltérés. A(z) <oszlopnév> oszlopba illesztendő adatok típusa nem helyes (<oszlop adattípusa>) legalább egy sor esetében.", nem kell megijedni mert az esetek többségében, a még nem deklarált oszlop-nevek generálják!



34. ábra a "beillesztés hozzáfűzéssel" panel a láthatatlan vezérlővel és a még nem deklarált oszlop-nevek okozta adattípus-hibával

A bővítmény csak a beillesztendő oszlopok számát és adataik típusát ellenőrzi. Mást nem. Az oszlopnevek deklarálását nem követi az oszlopok és a mezők sorrendjének összevetése! Tehát az azonos adattípusú oszlopok véletlen felcserélése hamis adatok tömegét eredményezheti a táblában!

A beillesztés hozzáfűzéssel utasításra a bővítmény a befogadó mező típusától eltérő adatokat a mező adattípusára konvertálja. Ha ez nem lehetséges, akkor erről a panel alján "Típuseltérés" kezdetű és a cél-mező nevét is tartalmazó szöveggel figyelmeztet. Több hiba esetén, jobbról balra haladva, az első megsértett adattípusú cél-mező nevét olvashatjuk az üzenetben. Az adattípus-hiba meghiúsítja az adatok beillesztését. Ha a hibát csak, a még nem deklarált, egyik oszlop-név okozza, akkor a jelölőnégyzetére kattintva az üzenet eltűnik.

a beillesztendő	a tábla mezőinek adattípusa						
adat típusa	szöveg	I	ogikai	pénznem	egész	tizedes	dátum
szöveg		AD	ATTÍPUS- HIBA	ADATTÍPUS- HIBA	ADATTÍPUS- HIBA	ADATTÍPUS- HIBA	ADATTÍPUS- HIBA
logikai	szövegként			ADATTÍPUS- HIBA	ADATTÍPUS- HIBA	ADATTÍPUS- HIBA	ADATTÍPUS- HIBA
pénznem	szövegként pénznem jelölő nélkül	<>0 0	igaz Hamis		e. kerekítve, pénznem jelölő nélkül	pénznem jelölő nélkül	ADATTÍPUS- HIBA
egész	szövegként	<>0 0	IGAZ HAMIS	pénznem jelölővel kiegészítve			ADATTÍPUS- HIBA
tizedes	szövegként	<>0 0	igaz Hamis	pénznem jelölővel kiegészítve	egészre kerekítve		ADATTÍPUS- HIBA
dátum	szövegként o:oo:oo időponttal kiegészítve	AD	ATTÍPUS- HIBA	ADATTÍPUS- HIBA	ADATTÍPUS- HIBA	ADATTÍPUS- HIBA	
idő	szövegként 1899.12.31. dátummal kiegészítve	AD	ATTÍPUS- HIBA	ADATTÍPUS- HIBA	ADATTÍPUS- HIBA	ADATTÍPUS- HIBA	

35. ábra

beillesztés hozzáfűzéssel műveletet megelőző konvertálás, adattípusok szerint, az üres cella az eredeti adat beillesztését jelenti

Az adattípus-hiba miatt meghiúsult műveletre, a villámnézet bezárása után megjelenő "Nem sikerült beilleszteni az adatokat" szövegű üzenet figyelmeztet. A balsikerű próbálkozás nem törli a vágólapot, az adatok továbbra is a rendelkezésünkre állnak.

további vágólap-műveletek

A Kezdőlap, Vágólap, Beillesztés cserével parancs törli az aktív tábla rekordjait és az üres táblába helyezi a vágólapon lévő sorokat. A művelet végrehajtása előtt a PowerPivot a beillesztés hozzáfűzéssel műveletnél már bemutatott parancstáblát jeleníti meg, eltérő szöveggel, de azonos képi elemekkel.

Az importálásos- és a vágólapos tábla-készítő módszer vegyíthető. Tehát egy adott PowerPivot munkafüzetben állhatnak importálással és vágólap-műveletekkel létrehozott táblák is.

Az adatok további feldolgozásához használhatjuk a *Kezdőlap, Vágólap, Másolás* parancsot is, amellyel vágólapra helyezhetjük a teljes táblát, illetve kijelölt rekordjait. Rekordok másolásánál a mezőnevek automatikusan a vágólapra kerülnek, de csak a bővítmény ablakában használhatjuk őket, az Excel ablakban csak a rekordok lesznek beillesztve. A másolás utasítás a kijelölt rekordok menüjéből is elérhető.

tábla adatbázis-táblázatból

fogalmak

Az adatbázis-táblázat egy az Excelben, a Beszúrás, Táblázatok, Táblázat- vagy a Kezdőlap, Stílusok, Formázás táblázatként utasítással vagy Ctrl+R billentyű-paranccsal létrehozott funkcionális egység, az adatbázis-jellegű műveletek egyszerű végrehajtására.

Az Excel adatbázis-táblázat létrehozásakor automatikusan, univerzális, tehát a teljes munkafüzetben használható nevet kap: Táblázat<sorszám>. Az automatikus név, hasonlóan a felhasználói nevekhez, felkerül a szerkesztőléc név mezőjének-, valamint a névkezelő-parancstábla listájára. Utóbbit a *Képletek, Definiált nevek, Névkezelő* utasítással jeleníthetjük meg.

Az automatikusan létrehozott nevet a Névkezelő parancstáblán, valamint aktív adatbázis-táblázat mellett a Táblázateszközök, Tervezés, Tulajdonságok csoportjában módosíthatjuk.



36. ábra az adatbázis-táblázat nevének módosítási lehetőségei

A névnek a következő előírásoknak kell megfelelnie. [1] Maximális hossza 255 karakter, [2] tartalmazhat kis- és nagybetűket valamint számokat, [3] nem tartalmazhat szóközt és a program foglalt karaktereit, [4] tagolása az _ (alsó vonal) és a . (pont) karakterekkel történik, [5] nem kezdődhet számmal, [6] nem lehet azonos egyetlen lehetséges hivatkozással, [7] önállóan nem lehet név a kis és nagy cé (c, C) és az er (r, R) betűk, [8] a csak kis- és nagybetűkben különböző neveket a program azonosnak tekinti.

A szövegében a közönséges táblázat elnevezést használom azokra a táblázatokra, amelyeket nem alakítottunk át adatbázis-táblázattá.

a tábla létrehozása

Kattintsunk az adatbázis-táblázatba, majd adjuk ki PowerPivot, Táblázatok, Hozzáadás az adatmodellhez utasítást. A parancs kiadása után a bővítmény megjeleníti a létrehozott táblát, amelynek neve megegyezik az adatbázis-táblázat nevével.

Amennyiben egy közönséges táblázatból adjuk ki a fenti utasítást, akkor először deklarálnunk kell a táblázat területét és oszlop-neveinek meglétét a megjelenített *Táblázat létrehozása* parancs-táblán és ezt követi az adatbázis-táblázat és a tábla létrehozása.

A bővítmény homogén adattípusú oszlopok esetén a vágólapos táblakészítésnél alkalmazott szabályokkal megegyezően állítja be a mezők adattípusát, kivéve a pénzeket, amelyek számára Tizedes tört szám adattípusú mezőt hoz létre. A vegyes adattípusú oszlopok szöveg adattípusú mezőket eredményeznek.

tábla a kimutatás forrásából

létrehozás

Mikor egy közönséges táblázatot szeretnénk kimutatással vizsgálni és kiadjuk a Beszúrás, Táblázatok, Kimutatás utasítást, a megjelenített Kimutatás létrehozása parancstábla bal alsó sarkában egy Adat felvétele az adatmodellbe feliratú vezérlőt találunk. A jelölőnégyzetet kiválasztva a bővítmény Tartomány néven táblát készít az aktuális táblázatból és ezt követően PowerPivot-kimutatást hoz létre az új tábla elemzésére.

Kimutatás létrehozása		?	\times	
Válassza ki az elemezni kívánt adatokat Táblázat vagy tartomány kijelölé <u>s</u> e				
Táblázat vagy tartomány: Munka 1!\$A	\$1:\$D\$11		Ť	
○ K <u>ü</u> lső adatforrás használata				
Kapcsolat választása				
Kapcsolat neve: A munkafüzet adatmodelljének használata				
Adja meg a kimutatás helyét				
		Kimut	atásr	mezők 🔹 🗙
Hel <u>v</u> :		Aktívak	Mind	
Válassza ki, hogy több táblázatot szeretne ☑ Adat felvétele az adat <u>m</u> odellbe	mezni OK	Válassza ki kívánt mez Keresés	a jelent zőket:	ésbe felvenni 😰 🔻
		▲ III Tart	tomány név szem. sz szül. dát fizetés	ám um

37. ábra "tábla-készítő" vezérlő és a létrehozott tábla a kimutatás segédablakában

Az Adat felvétele az adatmodellbe vezérlővel létrehozott tábla, más megfogalmazásban, egy segédtábla, amely biztosítja a korlátlan hozzáférést a PowerPivot szolgáltatásaihoz, beleértve a DAX nyelv használatát is. Ezt a magyarázatot erősíti az a tény is, hogy a kimutatás törlése a segéd-tábla automatikus törlését eredményezi. A kimutatással párhuzamosan létrehozott tábla kezelése mindenben azonos a más módon létrehozott táblákkal. Például összekapcsolhatjuk az adatbázis más táblájával.

táblák

az adatnézet felépítése

A táblákat a PowerPivot ablakban Adatnézetben látjuk. Ebben a nézetben tudjuk elvégezni a tábla-, a mező- és a rekordműveletek többségét. A másik megjelenítési mód a Diagramnézet, amely elsősorban a táblák közötti kapcsolatok megjelenítésére illetve kialakítására szolgál. Mivel a Diagramnézet semmilyen diagramot nem tartalmaz, ezért én ezt a megjelenítést a továbbiakban kapcsolatnézetnek fogam nevezni. A nézetek kapcsolói megtalálhatók a szalagon a Kezdőlap, Nézet csoportban, illetve az állapotsor jobb alsó sarkában. Utóbbiak feliratai: Rács, ez az adatnézet és Diagram, ez a kapcsolatnézet megjelenítője. Ebben a fejezetben az adatnézet felépítését és funkcióit ismertetem.

A nézet elemei felülről lefelé és balról jobbra haladva a következők. A szalag alatt a szerkesztő sor húzódik, a mezőlistával, a képletszerkesztő blokkal és a szerkesztő mezővel. Ezt követi a tábla kijelölésére szolgáló vezérlő, valamint a mezőneveket tartalmazó sor. Az utolsó mező után egy üres oszlop áll, *Oszlop hozzáadása* felirattal. Ez az oszlop a számított mező egyszerű létrehozását teszi lehetővé. A rekordok előtt, az ablak bal szélén futó keskeny, sorszámokat tartalmazó függőleges sáv a rekordok kijelölésére szolgál. A panel alsó harmadát a számítási terület foglalja el. Itt jeleníti meg a bővítmény a táblában tárolt összesítéseket. A két területet elválasztó borda egérrel mozgatható. A számítási terület a *Kezdőlap, Nézet, Számítási terület* paranccsal vagy az elválasztó bordán, dupla kattintással rejthető el. A görgetősávokat a bővítmény csak akkor jeleníti meg, ha a képernyőn kívül mezők illetve rekordok állnak.

		•					×
1	város 🔽	l negyedév 🔽	II negyedév 🔽	III negyedév 🔽	IV negyedév 🔽	Oszlop hozzáadása	^
1	Bácsalmás	71	54	72	81		
2	Balassagyarmat	98	-68	-69	34		
3	Balatonalmádi	-50	63	65	-20		
4	Balatonboglár	-62	-98	-84	-76		
5	Balatonföldvár	75	79	62	-47		
6	Balatonfüred	33	33	29	53		
7	Balatonfűzfő	-34	-23	-83	-65		
8	Balatonlelle	11	50	-14	10		
9	Balmazújváros	62	-91	83	-71		~
							^
							~
<							>
2	2005 összesen 2	006 összesen 20	007 összesen 🛛 200	08 összesen 2009	összesen 2011 ö	sszesen 🛄 🚽	
R	lekord: 🕶 🔹 1.	., összesen 14 🔹 🔸	 ▶1 			H	Ч.,:

38. ábra az adatnézet elemei

A vízszintes görgetősávot a füleket tartalmazó sor követi. A jobb szélén álló vezérlő, három pont és egy piciny háromszög, a nem látható táblák listáját jeleníti meg. A fülek alatt az állapotsor húzódik a rekord navigáció eszközeivel és a már említett nézetváltó parancsgombokkal.

Az adatnézetben a mezőnevek folyamatosan láthatók, a tábla függőleges görgetése csak a rekordokat mozgatja. Nagyítás funkcióval ez a nézet nem rendelkezik.

tábla-műveletek

Az importálással létrehozott táblák bővítésére vagy utólagos szelektálására szolgál a Tervezés, Táblázat tulajdonságai utasítással megjeleníthető a Táblázat tulajdonságai szerkesztése panel. A parancstáblán elérhető szolgáltatások közül csak egy kíván magyarázatot: Az oszlopnevek eredete felirat után álló Forrás vezérlő a táblázat oszlopneveit-, a Modell vezérlő a mezőneveket jeleníti meg. Természetesen, ha nem neveztük át a mezőket, akkor a nevek azonosak.

Táblázat tulajdonságain	k szerkesztése					
Táblázattulajdonságai Ezen a lapon módosi	blázat tulajdonságainak szerkesztése Ezen a lapon módosíthatók a táblázat-, oszlop- és sorszűrők leképezései.					
Tá <u>b</u> lázat neve:	2001 összesen					
Kap <u>c</u> solat neve:	Excel 09 eladások					
F <u>o</u> rrás neve:	'2001 összesen\$'	Villám <u>n</u> ézet frissítése				
Oszlopnevek eredete:	● <u>F</u> orrás ○ <u>M</u> odell					
🔽 🗹 város 💽	🗹 i negyedév 🛛 🔽 🛛 il neg	yedév 🛛 🔽 III negyed				
1 Bácsalmás	-17	-24				
2 Balassagyarmat	61	92				
3 Balatonalmádi	-72	-51				



Munkafüzetből illetve szövegfájlból importált adatok tábláinál a Váltás a következőre utasítás inaktív. A témakörünkben nem szereplő, lekérdezéssel létrehozott táblák esetén, ezzel a parancscsal válthatunk a tábla villámnézete és az őt létrehozó lekérdezés tervező nézete között.

A további tábla-műveleteket a tábla fülének menüjében találjuk. [1] Törlés: a tábla törlése, billentyűparancsa Ctrl+d. [2] Átnevezés: tábla átnevezése, billentyűparancsa Ctrl+r, egérrel dupla kattintás a fülön. [3] Áthelyezés: A Táblázat áthelyezése panel megjelenítése, billentyűparancsa Ctrl+m, egérrel a fül húzása. [4] Leírás...: adatnézetben a tábla fülére mutatva megjelenő rövid szöveg, billentyűparancsa Ctrl+e. A szöveg szerkesztése közben új sort a Ctrl+Enter billentyűparanccsal kérhetünk. [5] Elrejtés/Megjelenítés az ügyféleszközök elől: a tábla megjelenítésének tiltása a PowerPivot-mezőlistán/a tiltás feloldása. [6] Számítási terület megjelenítése: az az összesítések megjelenítésére szolgáló terület elrejtés/megjelenítése.

Az Elrejtés az ügyféleszközök elöl utasítás a szokásos beállítások mellett csak a PowerPivot-kimutatás segédablakából tünteti el a táblát, de a bővítmény ablakából nem. Ott csak a tábla fülecskéjének szürke színe jelzi a "rejtettséget". A tábla csak a Kezdőlap, Nézet, Rejtett elemek megjelenítése utasítás kiadása után tűnik el az ablakból.

rekord-navigáció

A táblában történő navigálás eszközei jórészt azonosak a munkalapon megszokottakkal. Vegyük számba az eltéréseket. [1] A függőleges görgetősáv csúszkáját húzva, a bővítmény az egérmutató bal oldalán megjeleníti a legelső látható rekord sorszámát és a tábla összes rekordjának számát. [2] Az állapotsor bal oldalán, a navigáló parancsgombok között az aktív rekord sorszámát látjuk a teljes rekordszám társaságában. Ez a vezérlő nem csak kijelző, hanem beviteli mező is: az "öszszesen" szóra kattintva begépelhetjük a megjelenítendő rekord sorszámát. Az *Enter* billentyűvel zárjuk le a bevitelt, amely egyben az utasítás kiadását is jelenti. [3] Az aktív rekord meghatározott mezőjére helyezhetjük a kurzort a szerkesztőléc elején álló, illetve az F5 billentyűvel megjeleníthető mező-listák segítségével. Az előbbi ABC sorrendben, utóbbi a táblában elfoglalt helyzetükkel azonos sorrendben tartalmazza a mezőket.

kijelölés

A tábla bal felső sarkában, a mezőnevek előtt álló kis négyszöggel, illetve a Ctrl+a billentyűparanccsal a teljes tábla kijelölhető, a mezőneveket is beleértve.

Adott mezőt a nevére kattintva, vagy a *Ctrl+szóköz* gyorsbillentyűvel jelölhetjük ki. A rekordok kijelölése a tábla bal szélén húzódó sáv elemeivel, vagy a *Shift+szóköz* billentyűparanccsal történhet. Az Excelben megszokott kijelölő billentyűparancsok a bővítmény ablakában is használhatók.

Többes kijelölés, azaz nem szomszédos elemek kijelölése, a táblában nem lehetséges.

mezők kezelése

A mezők sorrendjét a kijelölés után a mezőnevek húzásával módosíthatjuk. A mezőnevek menüjének Oszlopok rögzítése és a Tervezés, Oszlopok, Rögzítés, Rögzítés parancs a kijelölt mezőt a tábla jobb oldalára helyezi, a már ott lévő rögzített mezők után. A rögzített mezők a tábla vízszintes görgetésekor mozdulatlanok maradnak.

	\sim					\sim	
🖌 típus	🕅 rendszám 💌	szervizhe 🔽	szerv	izből		számla	🔽 sajáthibás 📘
1 Seat Leon	Kapcsolat létrehoz	tása		10. 08.	0:00:00	295 500 H	IUF IGAZ
2 VW Golf	Ugrás a kapcsoló	dó táblához		10. 15.	0:00:00	274 600 H	IUF IGAZ
3 Chevrolet (Másolás			11. 05.	0:00:00	255 800 H	IUF HAMIS
4 Opel Astra	🖞 Oszlop beszúrása			11. 07.	0:00:00	672 300 H	IUF IGAZ
5 Opel Corsa 📑	Oszlopok törlése			11. 05.	0:00:00	487 400 H	IUF HAMIS
6 Opel Corsa	Oszlop átnevezés	e		11. 05.	0:00:00	256 100 H	IUF HAMIS
7 Chevrolet A	Oszlopok rögzítés	e		10. 30.	0:00:00	296 600 H	IUF HAMIS
8 Chevrolet (Minden oszlop rög	zítésének feloldása		11. 12.	0:00:00	257 500 H	IUF IGAZ
9 Opel Vectra	Electés az ügyféle	ezközök elől		11. 21.	0:00:00	249 600 H	IUF HAMIS
10 Opel Antar		SZROZOR EIDI		10. 31.	0:00:00	255 500 H	IUF HAMIS
11 Opel Vectra	Oszlopszélesség			11. 07.	0:00:00	267 100 H	UF HAMIS
12 Opel Vectra	Szűrő		•	×	Szűrő tör	lése a követke	zőből: típus
13 Chevrolet A	Leírás	~		11. 19.	0:00:00	295 300 H	IUF IGAZ
14 Opel Vectra	ZPQ-733	2007. 11. 12. 0:00:00	2007.	11. 21.	0:00:00	273 000 H	IUF IGAZ
15 Opel Vectra	YSO-468	2007 11. 19. 0:00:00	2007.	12, 10	0:00:00	288 500 H	IUF IGAZ

40. ábra a mezőnév menüje

A mezőnév menüjének Elrejtés az ügyféleszközök elől utasításával letilthatjuk a mező megjelenítését a PowerPivot-kimutatás mezőlistáján. A letiltott mezőt a bővítmény szürkére színezi, de kezelhetőségét adatnézetben nem korlátozza. A PowerPivot-mezőlista frissítés után, ismét mutatja a mezőt, ha a blokkolást feloldjuk a mezőnév menüjének Megjelenítés az ügyféleszközök számára parancsával. A szokásos beállítások mellett a "rejtett" mező látható marad a bővítmény ablakában, de a Kezdőlap, Nézet, Rejtett elemek megjelenítése utasítással a táblából is eltüntethető.

A mezők szélességének megadása történhet: [1] hozzávetőlegesen, a mezőnév jobb oldali szegélyének húzásával, [2] a megjelenített legszélesebb mezőbejegyzéshez igazítva, a mezőt kijelölve, a mezőnév jobb oldali szegélyen duplát kattintva, vagy [3] pontosan, a mezőnév menüjének *Oszlopszélesség…*, illetve a *Tervezés, Oszlopok, Szélesség* paranccsal, képpontban megadva. Utóbbi mértékegység nem megszokott az Excelben. Hogyan használjuk? Nézzük meg a vezérlőpultban monitorunk felbontását! Például 1152*864. Az első szám a vízszintesen megjelenített képpontok számát jelenti. Ehhez a számhoz arányíthatjuk, mondjuk a 150 képpontot, amely körülbelül nyolcada a teljes szélesség. Mindhárom méretező-művelet több mező kijelölésével is működik. Az összes mező szélességének dupla kattintásos megadása, a tábla kijelölése után, tetszőleges mezőnév elválasztó vonalán elérhető. A bővítmény a dupla-kattintásos szélesség-beállításkor csak a megjelenített bejegyzések hosszát vizsgálja.

	[javítá Z]	•	1					
4		munkat 💌	születé 🚱	anvia neve		ó 💌	város 💽	utca, sz 💌
1		1 Karikás Haj	32303	(Az összes kije	lölése)	1041	Budapest,	Mády Lajos
2	2	2 Cseke Piros	32420	Kovács Linda		1157	Budapest,	Árendás kö
3	3	3 Szolnoki G	22191	Megyesi Ba		1016	Budapest,	Gellérthegy
4	4	4 Olajos Emma	25097	Ujvári Emma		1222	Budapest,	Bálvány utc
5	5	5 Kardos Nán	23982	Puskás Giz		1097	Budapest,	Vaskapu ut
6	6	6 Pandúr Péter	22107	Kövér Gertr		1013	Budapest,	Feszty Árpá
7		7 Alföldi Anikó	32558	Sajó Kinga	_	1138	Budanest,	Csaverevár

41. ábra

összes mező szélességének dupla kattintásos megadása, a megjelenített bejegyzések alapján

Amint láttuk a *Elrejtés az ügyféleszközök elöl* művelet funkciója a mező adatainak kivonása a pivot-táblás elemzésből. A bővítmény ablakában nincs lehetőségünk a mezők, az Excelben megszokott, rejtésére, de a mezőszélesség nullára állításával helyettesíthetjük a hiányzó műveletet. A kijelölhetőség megtartása érdekében a nulla szélességű mezőket a bővítmény öt képpont szélességűre állítja be.

	[sajáthibás]	▼ IGAZ			
	javítás AZ 🔽	munkatárs neve 🔽		osztály 💽	típus
1	1	Karikás Hajnalka	06093173	IT	Seat Leon
2	2	Cseke Piroska	. 28810049380	beruházási	VW Golf
3	3	Szolnoki Gabriella	26010027243	beruházási	Chevrolet Captiva
4	4	Olajos Emma	26809162865	IT	Opel Astra
5	5	Kardos Nándor	16508289266	jogi	Opel Corsa
6	6	Pandúr Péter	16007106308	kommunikációs	Opel Corsa
7	7	Alföldi Anikó	28002191763	kommunikációs	Chevrolet Aveo

42. ábra "nulla" szélességűre állított mezők

A mezőnév menüjének *Leírás…* parancsával a mező tartalmáról adhatunk információt a munkafüzettel dolgozóknak. A kijelölt mező nevének módosítása illetve a mező törlése a mezőnév menüjének parancsaival történhet. Az aktív mezőt a *Delete* billentyűvel is törölhetjük.

rekordok automatikus rendezése

A bővítmény az importálással létrehozott táblák rekordjait rendezetten jeleníti meg. Az automatikus rendezés algoritmusa a következő. [1] A bővítmény megszámlálja a mezők egyedi bejegyzéseinek előfordulásait. [2] Az a mező lesz a rekordok rendező mezője, amelynek valamelyik bejegyzése legalább hatvannégyszer előfordul, a mezőket balról jobbra vizsgálva. [3] Ha több mezőnek is van ilyen bejegyzése, akkor a legmagasabb előfordulás-számú bejegyzés mezője lesz a rendezés alapja. [4] Ha a legmagasabb előfordulás-szám egy vagy több mezőben azonos, akkor az a mező lesz a rendezés alapja, amelynek második legnagyobb előfordulás-száma a legnagyobb. [5] Két egyedi bejegyzést tartalmazó mezőt, például logikai adattípusú mezőt, csak akkor választ rendező mezőnek a bővítmény, ha a tábla több egyedi bejegyzésű mezői nem teljesítik a fenti feltételeket. [6] A rekordok sorrendjét a rendező mező bejegyzéseinek előfordulás-száma határozza meg. Tehát a legtöbbször előforduló bejegyzést tartalmazó rekordok lesznek a táblázat tetején, majd a második leggyakrabban előforduló bejegyzést tartalmazók következnek. És így tovább. [7] A határértéket el nem ért bejegyzéseket tartalmazó rekordokat a bővítmény a táblázat alján rendezetlenül jeleníti meg.

rekordok felhasználói rendezése

A rekordok csak egyetlen mező bejegyzései alapján rendezhetők. A növekvő és a csökkenő rendezés parancsait a *Kezdőlap, Rendezés és szűrés csoport* valamint a mezők szűrőlistái tartalmazzák. A műveletet előtt a rendezés alapjának szánt mezőt, egy bejegyzésére kattintva ki kell választanunk.

A rekordok eredeti sorrendjét a Kezdőlap, Rendezés és szűrés csoport Rendezés kikapcsolása vagy a rendező mező szűrőlistájának Rendezés törlése a következőből: <mezőnév> paranccsal állíthatjuk vissza.

a mező adattípusa és számformátuma

A mező adattípusának megállapítása, mint láttuk, automatikus. Felhasználói módosítása a Kezdőlap, Formátum, Adattípus listájával történhet. A felajánlott konvertálási lehetőségek végrehajthatóságát az adott mező bejegyzései határozzák meg.

Az adatok megjelenítését, az Excelben használatos kifejezéssel számformátumát, a *Kezdőlap, Formátum, Formátum* legördülő menüből választhatjuk ki. A bővítményben egyéni formátumkódot közvetlenül nem definiálhatunk, csak számított mező segítségével. A dátum adattípusú mező számformátum-listájában szereplő, csillaggal jelzett elemek a vezérlőpult szerinti formátumok. A dátum-lista utolsó elemével, *További dátumformátumok…* a teljes formátum-listát és a kiválasztott formátum kódját kapjuk.

A numerikus adattípusú, de nem dátum adattípusú mezőkhöz választható megjelenítések a következők. [1] *Általános*: ezres elválasztó nélkül, az összes tizedesjeggyel. Pénznem adattípusú mezőben ezres elválasztóval, két tizedesjeggyel. [2] *Tizedes tört szám*: ezres elválasztó nélkül, két tizedesjeggyel. Pénznem adattípusú mezőben ezres elválasztóval. [3] *Egész*: Ezres elválasztó és tizedesjegyek nélkül. [4] *Pénznem*: Ezres elválasztóval, két tizedesjeggyel és forint jelölővel. [5] *Százalék*: A bejegyzés százszorosának megjelenítése, ezres elválasztó nélkül, két tizedesjeggyel, százalék jelölővel. [6] *Tudományos*: Megjelenítés a számok normál alakjában, a mantissza két tizedesjeggyel, a karakterisztika három számjeggyel megjelenítve.

A bejegyzések megjelenítését a szalag *Kezdőlap, Formátum* csoportjának parancsgombjaival is szabályozhatjuk. A funkciók, sorrendben: [1] pénznem formátum, listából választható pénznem jelölővel, [2]százalékos formátum, [3] a szám egészrészének megjelenítése ezres csoportosításban, [4] tizedesjegyek számának növelése és [5] csökkentése.



43. ábra a megjelenítés parancsgombjai a szalagon

műveletek visszavonása ismétlése

Az adatnézetben elvégzett műveletek a *Tervezés, Szerkesztés, Visszavonás/Mégis* parancsokkal vonhatók vissza illetve a visszavont műveletek újra végrehajthatók. A parancsgombok folyamatosan frissülő listáit a visszavonhatatlan műveletek törlik. Melyek ezek a műveletek? A *Táblázat tulajdonságainak szerkesztése* parancstáblán végrehajtott műveletek, mező törlése, tábla frissítés és a tábla törlése. Nem feljegyzett műveletek: mező vagy tábla leírásának módosítása, táblák sorrendjének módosítása.

tábla frissítése

Az esetleges adatvesztések elkerülése érdekében a tábla aktualizálása három lépésben történik. [1] A forrás-adatokat beolvassa az operatív tárba. [2] A beolvasott adatok homogenizálása. [2] Sikeres konvertálás esetén, a rekordok felülírása.

Az importálással létrehozott táblákat a Kezdőlap, Frissítés listájának parancsaival aktualizálhatjuk. A Frissítés utasítás az aktív táblát, az Összes frissítése utasítás a munkafüzet minden tábláját aktualizálja. A művelet végrehajtásának pillanatnyi állapotát, az automatikusan megjelenített, Adatfrissítés parancstáblán látjuk.

A táblák csoportos frissítésekor bármely tábla aktualizálási hibája az összes tábla frissítését meghiúsítja. Ez látható a következő képen: mind a tizennyolc táblánál a bővítmény hibát jelez, holott csak a 2003-as tábla aktualizálása nem lehetséges. A *Hiba részletes adatai* link egy semmitmondó szöveget jelenít meg.

Adat	Adatfrissítés ? X							
Adatfrissítés előrehaladása A frissítés beolvassa a módosított adatokat az eredeti adatforrásokból.								
6	Hiba		18 Összesen 0 Megszakítva					
	Tiba		0 Sikeres 18 Hiba					
Rés	zletek:							
	Munkaelem	Állapot	Üzenet					
8	2007 összesen	Hiba	Hiba részletes adatai					
8	2008 összesen	Hiba	Hiba részletes adatai					
8	2009 összesen	Hiba	Hiba részletes adatai					
8	2010 összesen	Hiba	Hiba részletes adatai					
8	2011 összesen	Hiba	Hiba részletes adatai					
8	2012 összesen	Hiba	Hiba részletes adatai					
8	2013 összesen	Hiba	Hiba részletes adatai					
8	2014 összesen	Hiba	Hiba részletes adatai					
8	2015 összesen	Hiba	Hiba részletes adatai					
8	2016 összesen	Hiba	Hiba részletes adatai					
8	2017 összesen	Hiba	Hiba részletes adatai					
8	2018 összesen	Hiba	Hiba részletes adatai					
			Frissítés leállítása Bezárás					

44. ábra a táblák csoportos frissítésekor előforduló hiba jelzése

A munkafüzetből importált táblák frissítésekor a bővítmény törli azokat az adatokat, amelyek homogenizálással nem hozhatók a befogadó mező adattípusára.

Az aktualizálást alatt a bővítmény, táblák létrehozásának módjától függetlenül, mindig megjeleníti az Adatfrissítés panelt! A kimutatás forrásából létrehozott tábla is frissíthető, sőt a két objektum, a kimutatás és a tábla, kölcsönösen frissítik egymást. Magyarul, ha frissítjük az egyiket a bővítmény automatikusan átvezeti a másik objektumba az új adatokat. A kimutatás forrásának bővítése utasítás (Kimutatáseszközök, Elemzés, Adatok, Más adatforrás megadása, Más adatforrás megadása) automatikusan a táblát is bővíti.

A vágólappal létrehozott tábla nem frissíthető.

A bővítmény még a PowerPivot munkafüzet megnyitásakor sem frissíti automatikusan a táblákat, ezért a táblák aktualizálása, teljes egészében, a felhasználó feladata!

kapcsolatok beolvasása

fogalmak

A kapcsolat két tábla közötti logikai viszony, amely a rekordok szintjén valósul meg. Az összetartozó rekordok visszakereshetőségét a két tábla kapcsoló mezőiben álló, azonos bejegyzések biztosítják. Például megszámláltathatjuk az egyes megrendelők megrendeléseit a "megrendelők" és a "megrendelések" táblák közötti kapcsolat segítségével.

Ha a két tábla között nincs kapcsolat meghatározva, akkor a bővítmény minden megrendelőhöz az összes megrendelést társítja és a statisztikai vizsgálat eredményeként kapott megrendelés-szám minden megrendelőnél a megrendelések tábla rekordszámával lesz azonos.

A kapcsolat létrehozható [1] a definíciókat a forrás-adatbázisból beolvasva, [2] automatikusan a bővítmény algoritmusaival, vagy [3] felhasználói deklarációval. A bővítmény ablakában álló táblák és kapcsolataik alkotják a PowerPivot adatbázist.

Két logikailag összetartozó tábla egyikének rekordjaihoz általában a másik tábla több rekordja tartozik. Például: megrendelő-megrendelések, biztosítók-biztosítottak. Viszonylag ritka, de előfordulhat, hogy a két tábla tetszőleges rekordja a másik tábla egyetlen rekordjával tartozik össze. Például országok-fővárosok, megyék-megyeszékhelyek. Felmerülhet a kérdés, miért nem egy táblában állnak az összetartozó rekordok? Az elkülönített tárolás oka leggyakrabban: a nagy táblák nehéz kezelhetősége, az állandó és a gyakran változó-, vagy a nyilvános és bizalmas adatok elkülönített tárolásának igénye...

A bővítmény nem tesz különbséget az "egy az egyhez" és az "egy a többhöz" kapcsolat-típus között, de megköveteli, hogy a kapcsoló mezők egyike [1] csak egyedi bejegyzést tartalmazzon és [2] ne legyenek üres bejegyzései. A kapcsolatok tárgyalásánál ezt a mezőt és tábláját egy oldali táblának, illetve egy oldali kapcsoló mezőnek fogom nevezni. A kapcsolat másik oldalán álló tábla illetve kapcsoló mezője a több oldali tábla, illetve a több oldali kapcsoló mező.



45. ábra kapcsolatok és kapcsoló mezők ábrázolása kapcsolatnézetben

A PowerPivotban két tábla között több kapcsolat is létrehozható, de a bővítmény mindig csak az egyiket, az "aktív" kapcsolatot veszi figyelembe. Képletben függvénnyel aktiválhatjuk, a számítás idejére, az inaktív kapcsolatot.

A PowerPivot-kimutatás mindig a teljes PowerPivot adatbázist vizsgálja. Ez a szokásos működés. A felhasználó az "elrejtés az ügyféleszközök elől" művelettel vonhat ki táblát az elemzésből.

kapcsolatok importálása

Access adatbázisokból, tehát az "mdb" és a "accdb" kiterjesztésű fájlokból, nem csak az adatbázis tábláit, de a köztük lévő kapcsolatokat is beolvastathatjuk a *Kezdőlap, Külső adatok beolvasása, Adatbázisból, Az Access programból* paranccsal. A művelet végrehajtását a *Tábla importálása varázsló* segíti.

A segédprogram első, Kapcsolódás Microsoft Access-adatbázishoz feliratú ablakában meg kell adnunk az adatkapcsolat nevét és tallózás funkcióval ki kell választanunk a forrás-fájlt. Ezután célszerű a Kapcsolat tesztelése funkcióval ellenőrizni az adatbázis megnyithatóságát: nincs-e jelszóval
védve vagy aktuálisan zárolva. Az első esetben be kell jelentkeznünk az adatbázisba a *Felhasználó*név és a *Jelszó* mezők kitöltésével. A bővítmény felkínálja a jelszó mentését.

A következő lépésében Az adatimportálás módjának kiválasztása parancstáblán döntenünk kell, hogy a segédprogram grafikus eszközeivel, vagy egy SQL nyelvű lekérdezéssel határozzuk meg az importálandó adatokat. A Választás táblák és nézetek listájáról... utasításra a Táblák és nézetek kijelölése feliratú, Az importálandó adatokat megadó lekérdezés készítése parancsra pedig az SQL-lekérdezés megadása panel jelenik meg.

A segédprogram Táblák és nézetek kijelölése feliratú ablaka már ismerős számunkra. A már látott és a most megjelenített lista között csupán annyi a különbség, hogy itt az adatbázis táblái, míg ott a munkafüzet lapjai közül választhattunk. A kiválasztott táblához közvetlenül csatlakozó táblákat A kapcsolódó táblák kijelölése utasítással választhatjuk ki. Az importáló varázsló nem csak a kiválasztott táblákat, de a köztük lévő kapcsolatok is beolvassa.

A Befejezés gombra kattintva a varázsló utolsó, *Importálás* feliratú ablakában nyomon követhetjük a beolvasás fázisait. A program először a kijelölt táblákat importálja, majd a parancstáblán megjelenik az Adatok előkészítése felirat, amely a kapcsolat-deklarációk beolvasását jelenti. A bővítmény a határozatlan, tehát a "hivatkozási integritás megőrzése" szolgáltatással nem védett kapcsolatokat is beolvassa. Amikor a felirat előtt álló zöld nyíl pipára vált, megjelenik a lista harmadik oszlopában a *Részletek* link, amellyel a beolvasott kapcsolatok leírását jeleníthetjük meg.

C	Sikoros	Összesen: 3 M	egsz	akítva: 0			
4	Sikeres	Sikeres: 3	R	lészletek		?	Х
\és	zletek:		ſ	Kananak	t a fle	R421 × 642-1, R421	
	Munkaelem	Állapot		- Állapot:	Sikere	grenij->remak grenij es, Aktív	^
2	férfiak	Sikeresen végrehajtva. 19 sor átvitele befejeződött.	Kapcsolat: gverekek [nő] -> nők [nő]		rekek ínől ->nők ínől		
2	gyerekek	Sikeresen végrehajtva. 8 sor átvitele befejeződött.		- Allapot:	Sikere	es, Aktív	
0	nők	Sikeresen végrehajtva. 19 sor átvitele befejeződött.			<u> </u>		
2	Adatok előkészítése	Befejezve	Ré	szletek			

46. ábra Access adatbázis-táblák és kapcsolataik beolvasása

A Részletek parancstáblán kis nyilacska mutatja a beolvasott kapcsolat "irányát": a nyíl a kapcsolat több oldalán álló mezőtől indul és a kapcsolat egy oldalán álló mezőre mutat.

Adatnézetben a beolvasott táblák, nevük szerint, ABC sorrendben állnak.

A bővítmény az Access számláló adattípusú, valamint bájt, egész és hosszú egész méretű, szám adattípusú mezőinek Egész szám, az egyszeres, a dupla és a decimális méretű szám adattípusú mezőinek Tizedes tört szám adattípust állít be.

A bővítmény adatnézetben a mezőnév után álló piciny ábrácskával jelöli meg a tábla kapcsoló mezőit. A csak egyedi bejegyzéseket tartalmazó-, tehát a kapcsolat egy oldalán álló mező ábrácskája két, sorszámozott tábla, míg a több oldalon álló mező ikonja ugyanez az ábrácska egy nagyítóval kiegészítve. A mezőnévre mutatva a bővítmény szövegdobozt jelenít meg a kapcsolat másik oldalán álló tábla- és kapcsoló mezőjének nevével.

1	nő 📲 🖬	férfi 👘 🖬	szszám 🔽
1	Udvardi Emese	Varga Levente	26003035101
2	Nyéki Mária		26405097746
3	Sasvári Ágnes	Kovács Pál	28010071226
4	Keleti Boriska	Pásztor Vajk	28903257873
5	Gyulai Debóra	Kontra Zoltán	26102176064
6	Szebeni Dóra	Rédei Örs	28305077092

^{47.} ábra kapcsoló mezők jelzése adatnézetben, a mezőnevek mellett

kapcsolatnézet

A beolvasott táblák és kapcsolatok grafikus megjelenítését a bővítmény ablakának *Kezdőlap, Nézet, Diagramnézet* utasításával vagy az ablak jobb alsó sarkában álló *Diagram* parancsgombbal kezdeményezhetjük. A táblákat feliratozott téglalapok-, a kapcsolatokat vonalak szimbolizálják. A vonal közepén álló nyíl a kapcsolat több oldalán álló táblára mutat. Az egy oldali táblát a vonal végén álló egyes (1), a több oldalit a vonal másik végén álló csillag (*) jelzi. A táblák és kapcsolataik rajzos megjelenítése a kapcsolat-háló, a megjelenítő felület pedig a kapcsolatnézet.

A kapcsolatnézet megjelenítési eszközeit az állapotsorban találjuk. Balról jobbra haladva az első vezérlő a Képernyőhöz igazítás feliratú négy-irányú nyilas ábrácska, melynek funkciója az ideális megjelenítési méretarány automatikus beállítása. A nagyító csúszkát a *Megjelenítés* feliratú kis menü követi, a táblák megjelenítését szabályozó jelölőnégyzetekkel.



48. ábra a kapcsolatnézet megjelenítési eszközei az állapotsorban

A táblák a címsorukra mutatva mozgathatók, illetve a szegélyüket "megfogva" méretezhetők. A kijelölt tábla a Ctrl+nyíl billentyűparanccsal is mozgatható. A teljes nézet nagyíthatósága mellett az egyes táblák külön-külön is nagyíthatók a jobb sarkukban álló, kattintásra vagy az egérmutató közelítésére megjelenő *Teljes méret* vezérlővel. Nagyításban a vezérlő felirata *Visszaállításra* változik.

A PowerPivot-kimutatás mezőlistájáról az Elrejtés az ügyféleszközök elöl utasítással letiltott mezőket és táblákat a kapcsolatnézetben halvány színekkel formázva látjuk. Ez a szokásos megjelenítés. Teljes rejtésük, majd ismételt megjelenítésük a Kezdőlap, Nézet, Rejtett elemek utasítással kezdeményezhető.

A kapcsolat vonalára mutatva a bővítmény a kapcsoló mezőket szegéllyel emeli ki.

A kapcsolat-hálón végzett módosításainkat a program a PowerPivot munkafüzet részeként menti, ha azonban elégedetlenek vagyunk az átalakítással, akkor a kapcsolatnézet eszköztárának *Elrendezés alaphelyzetbe állítása* parancsával visszatérhetünk a bővítmény elrendezéséhez.

műveletek kapcsolatnézetben

A kapcsolatnézet a kapcsolatok megjelenítésén és kezelésén felül, még néhány alapvető tábla- illetve mező-művelet elvégzésére is alkalmas. A táblát a címsorára-, a mezőt a nevére-, a kapcsolatot a vonalára kattintva jelölhetjük ki.

A tábla- és a mező menüje tartalmazza az átnevezés, a rejtés és a törlés parancsát. Az Ugrás utasítással a kijelölt objektumot jeleníthetjük meg adatnézetben. A kijelölt objektumok, beleértve kapcsolatot is, a *Delete* billentyűvel távolíthatók el a PowerPivot adatbázisból.

A kapcsolatnézetben elvégzett műveletek nem vonhatók vissza és nem ismételhetők.

kapcsolatok automatikus létrehozása

fogalmak

A PowerPivot-kimutatás lehetőséget teremt az elkülönítve tárolt adatok egységes elemzésére. Kapcsolatok hiányában azonban ez a szolgáltatás nem működne, ezért a bővítmény felhasználói kérésre megkísérli önállóan létrehozni a kapcsolatokat. A táblák kapcsoló mezőit a PowerPivot a mezők nevének, adattípusának és bejegyzéseinek vizsgálatával próbálja meghatározni. A felhasználói beavatkozás nélkül létrehozott kapcsolat mindenben egyenértékű a beolvasott illetve a manuálisan létrehozott kapcsolattal.

a modul működése

PowerPivot-kimutatás készítésekor, ha nem a sor- illetve az oszlopmezők tábláiból helyezünk mezőt az értékek területre, üzenet figyelmeztet a mezőlistán a kapcsolatok hiányára.

Kimutatásmezők	- ×	ré	gió	-	Elemszám - város
		D	él-Alföld		256
Válassza ki a jelentésbe felvenni kívánt mezőket:	5 +	D	él-Dunántúl		256
	~	És	szak-Alföld		256
Contradict light at a stick light large static large searches have		És	szak-Magyarorszá	ig	256
Szükseg lehet a tablak közötti kapcsolatokra.	×	Kč	özép-Dunántúl		256
Automatikus felismerés LÉTREHOZÁS		Kč	özép-Magyarorsz	ág	256
		N	yugat-Dunántúl		256
Keresés	P	V	égösszeg		256

49. ábra kapcsolatok nélküli adatbázis mezőlistája és kimutatása

A képen látható kapcsolatok nélküli adatbázis táblái: "régiók", "megyék", "városok". Kimutatással meg kell határoznunk a régióik városainak számát. A kapcsolatok hiánya miatt azonban a kimutatás minden régióhoz megjeleníti a városok tábla rekordszámát, azaz az összes város darabszámát.

Az automatikus-kapcsolat modult a figyelmeztető üzenet alatt megjelenített Automatikus felismerés... vezérlővel indíthatjuk el. A művelet fázisait a megjelenített Kapcsolatok automatikus felismerése panelen követhetjük nyomon, amelyen eleinte csak egyetlen, a művelet megszakítására szolgáló, Mégse parancsgomb áll. Az eljárás befejezését követően a bővítmény ezen a panelen tájékoztat a művelet eredménytelenségéről vagy a létrehozott kapcsolatok számáról.

Kapcsolato	k autom	atikus felisr	nerése		?	×		
Kapcsol A művelet	atok f az adat	elismer tok összete	ése és létrehozása ettségétől függően több	percet is igén	ybe vehe	et.		1
Kaj	pcsolat	észlelése:	folyamatban				×	
					Még	Ise	vehet.	
		Kapcsola	t észlelése: kész					×
	-	Nem talá	hatók új kapcsolatok	K <u>a</u> pcsolatok	kezelése.		Bezárás	vehet.
		\checkmark	Kapcsolat észlelése: ké	ész				
			2 uj kapcsolat létrehoz	zva	K <u>a</u> pcsola	tok keze	elése	Bezárás

50. ábra az automatikus-kapcsolat modul működését mutató parancstábla három lehetséges állapota

A *Kapcsolatok kezelése…* gombra kattintva megtekinthetjük a létrehozott kapcsolatokat. A második oszlopban az egy oldali tábla nevét és zárójelben a kapcsoló mező nevét olvashatjuk. A több oldali tábla és kapcsolómezőjének neve a harmadik, *Kapcsolódó keresési tábla* feliratú oszlopban áll. A panel jobb oldalán lévő parancsgombok a kijelölt kapcsolat kezelésére, illetve az automatikuskapcsolat modul indítására szolgálnak.

Aktív kimutatás mellett az automatikus kapcsolat-felismerést a PowerPivot, Kapcsolatok, Észlelés paranccsal is elindíthatjuk, de ebben az esetben nem a fent ismertetett panelt jeleníti meg a bővítmény.

Kapcsolat		? ×
Észlelés és létrehozás		
Folyamatban van a kapcsolatok észlelése és létr Az adatok összetettségétől függően ez több perce	Létrehozott kapcsolatok: t. [városok].[megye] -> [megyék].[megye] [megyék].[régió] -> [régiók].[régió]]	
Befejezve		-
Részletek:		
Munkaelem	Összegzés	
Kapcsolatok észlelése: kész	<u>Részletek</u>	1
Kapcsolatok létrehozása: kész	Résziptek	ОК
	Г	
raibeszeupanei automatikus bezarasa	L	bezaras Megse

51. ábra az automatikus-kapcsolat modul másik parancstáblája

Az Észlelés paranccsal indított kapcsolat-felismerés parancstábláján a Részletek hivatkozásra kattintva tekinthetjük meg a létrehozott kapcsolatot. A megjelenített definíció nem szerkeszthető!

A kapcsolatok automatikus létrehozása, még kis rekordszámú táblák esetén is, rendkívül időigényes, ezért a hatékonyság érdekében, tartsuk szem előtt a szolgáltatás korlátait.

A bővítmény mindig csak az aktuális kimutatás-elrendezés statisztikai vizsgálatához szükséges táblákat próbálja összekapcsolni! Ez általában egy kapcsolat létrehozását jelenti, de a csak közvetett logikai viszonyban álló táblák vizsgálatakor az összes szükséges kapcsolat létrejön. Folytassuk az előbb felvázolt példát!



52. ábra

a közvetett logikai viszonyban álló "régiók" és "városok" táblák

A bővítmény közvetlenül nem tudja összekapcsolni a "régiók" táblát a "városok" táblával, ezért először kialakítja a városok-megyék, majd a megyék-városok kapcsolatot.

A bővítmény csak szöveg vagy egész szám adattípusú mezőt választ kapcsoló mezőnek, ezért a munkafüzetből importált, egész számokat tartalmazó mezők adattípusát Tizedes tört szám típusról át kell állítani Egész szám típusúra!

Ha a vizsgált mező minden bejegyzése egyedi és nincs üres bejegyzése, akkor ez lehet az egy oldali kapcsoló mező. Ha a modul talál egy olyan mezőt, amelynek bejegyzései több mint kilencvenöt százalékban szerepelnek az először kiválasztott mezőben, akkor ez lesz a több oldali kapcsoló mező. Az automatikus kapcsolat-létrehozás szolgáltatás a mezők nevét is vizsgálja. A kapcsolat csak akkor jön létre, ha a következő feltételek teljesülnek.

[1] A két név azonos. A modul a kis- és nagybetűket valamint az elválasztó karaktereket (szóköz, kötőjel, alsóvonal) nem különbözteti meg. A különböző sorrendű, de megegyező szavakat tartalmazó neveket a modul azonosnak tekinti.

[2] Az egy oldali mező egyszavas nevének egy darabja a több oldali név egyik "szava". A kapcsolat létrehozásához szükséges egyező karakterek száma legalább az egy oldali név hosszának tizede, felfelé egészre kerekítve: KEREK.FEL(HOSSZ(<egy oldali név>)/10; 0). Az egyező karakterláncnak az egy oldali név elejéről kell származnia, a következő szabály szerint: BAL(<egy oldali név> ; PADLÓ(HOSSZ(<egy oldali név>); 2)/2 - 1 + KEREK.FEL(HOSSZ (<egy oldali név>)/10; 0)). Lássunk egy példát! Az egy oldali név "termékazonosító". A kapcsolat létrehozásához szükséges egyező karakterek száma: KEREK.FEL(HOSSZ("termékazonosító")/10; 0) \rightarrow 2. Az egy oldali név eleje, ahonnan a két egymást követő karakternek származnia kell: BAL("termékazonosító" ; PADLÓ(HOSSZ("termékazonosító"); 2)/2 - 1 + KEREK.FEL(HOSSZ ("termékazonosító")/10; 0)) \rightarrow "termékaz". Tehát, ha ebből a karakterláncból a több oldali név önállóan tartalmaz, két tetszőleges, egymás után álló karaktert, akkor a kapcsolat létrejön. Például a több oldali név lehet: "árú er" vagy "árú az".

[3] Ha az egy oldali név több szóból áll, akkor a több oldali névnek sorrendben tartalmaznia kell az egy oldali név minden szavának egy részét, az előző szabályban előírt módon.

egy oldali	azonosító	termék azonosító	alkatrész kód	termékazonosító	belső raktárkészlet
több oldali	AZONOSÍTÓ	termék_azonosító	kód alkatrész	árú az	br

53. ábra

néhány példa az automatikus kapcsolat név-követelményeinek szemléltetésére

Egy az egyes automatikus-kapcsolat esetén a statisztikai mezőt tartalmazó tábla lesz a "több oldali tábla", azaz a létrehozott kapcsolat a statisztikai mező táblájára mutat.

kapcsolat manuális létrehozása

fogalmak

Az eddig megismert két módszer, a beolvasás és az automatikus létrehozása mellett, természetesen a kapcsolat felhasználói deklarációval is létrehozható. A kapcsolat meghatározása a kapcsoló mezők megadásából áll. A PowerPivot szóhasználatában a kapcsolat egy oldalán álló tábla és kapcsoló mezőjének neve "keresési táblázat" illetve "keresési oszlop".

Kapcsoló mezőnek tetszőleges nevű és adattípusú mezőt választhatunk, de a kapcsolat csak akkor jön létre, ha a kapcsoló szerepre kijelölt mezők egyike nem tartalmaz ismétlődő- és üres bejegyzéseket. Ha ez a két kritérium teljesül, akkor a kapcsolat létrejön. Ebből a tényből azonban az is következik, hogy a bővítmény nem akadályozza meg a funkciótlan kapcsolat kialakítását. Például kiválaszthatunk olyan több oldali kapcsoló mezőt, amelynek egyetlen bejegyzése sem szerepel az egy oldali kapcsoló mezőben.

A bővítmény a kapcsolatokkal összefüggésben több parancstáblán az adatbázis-kezelés terminológiájával él: a kapcsolat egy oldalán álló mezőt "elsődleges kulcsnak" míg a több oldalit, "külső kulcsnak" nevezi. Utóbbi kifejezés hibás: a kapcsolat több oldalán álló mező az "idegen kulcs" mező.

Két tábla között több kapcsolat is létrehozható. Természetesen a számítások végrehajtásakor a bővítmény csak egyetlen kapcsolatot vesz figyelembe. A használatra kijelölt kapcsolat az aktív kapcsolat. A két tábla között létrehozott első kapcsolat automatikusan aktív.

nem kapcsolódó rekordok

A különböző táblákban álló, logikailag összetartozó rekordok visszakereshetőségét a kapcsoló mezők azonos bejegyzései biztosítják. A bővítmény azonban nem követeli meg, hogy a több oldali tábla kapcsoló mezőjének bejegyzései szerepeljenek az egy oldali tábla kapcsoló mezőjében. Másként fogalmazva a több oldali táblában állhatnak olyan rekordok is, amelyeknek nincs kapcsolódó rekordjuk az egy oldali táblában. Ezek a nem kapcsolódó rekordok. A kimutatásban a bővítmény a nem kapcsolódó rekordokat egy kalap alá veszi és az "(üres)" feliratú "tételben" jeleníti meg, amely mindig a valós tételeket követően, az utolsó helyen áll a kimutatásban.





54. ábra nem kapcsolódó rekordok adatainak összesítése a kimutatásban

A képen látható kimutatás a kéttáblás (szervizek-javítások) adatbázis elemzésére készült: a számla mező értékeit összegzi a szerviz mező egyedi értékei szerinti csoportosításban. A szervizek táblában nem szereplő cégek számláinak összegét az (üres) címkéjű tétel tartalmazza.

Nem kapcsolódó rekordokat találunk, a már ismertetett, férfiak-nők-gyerekek adatbázis elemzésére létrehozott kimutatásban is.



nem kapcsolódó rekordok adatainak összesítése a kimutatásban

A kép tanúsága szerint két gyereknek az apukája nem található a férfiak táblában. Nem kapcsolódó rekordok tehát közvetett kapcsolatban álló táblák elemzésekor is megjelenhetnek a kimutatásban.

a kapcsolat létrehozása

Adatnézetben a Tervezés, Kapcsolatok, Kapcsolat létrehozása utasítással kezdeményezhetjük a műveletet. A megjelenő panel a táblák és a kapcsoló mezők egyszerű kiválasztását teszi lehetővé.

Kapcsolat le	étrehozása							?	×
Jelölje ki a	z egymás	hoz kapcs	solódó táb	olákat és	oszlop	okat.			
jav ítások						~			
javítás	rendszám	sajáthibás	s számla	szerviz		szervizbe	szervizből		
1	JAF-794	False	583200	Répássy I	⟨ft.	1999. 09. 08. 0:00:00	1999. 09. 18	. 0:00:00	
2	WYA-469	True	568000	Répássy I	Kft.	1999. 09. 08. 0:00:00	1999. 09. 19	. 0:00:00	
3	WYA-471	False	473900	Budai Aut	ójav ító	1999. 09. 18. 0:00:00	1999. 09. 25	. 0:00:00	
4	JAF-796	False	592900	Budai Aut	ójav ító	1999. 09. 26. 0:00:00	1999. 10. 07	. 0:00:00	
5	SBA-444	True	599500	Budai Aut	ójav ító	1999. 10. 02. 0:00:00	1999. 10. 12	. 0:00:00	
szerviz						~			
irány ítószár	n szerviz		utca-szám	1	város				
1213	Tóth & Fi	ai	Róka utca	13.	Budape	est			
1133	Fekete A	utószerviz	Árbóc utca	a 29.	Budape	est			
1039	FCA		Almos utca	a 27.	Budape	est			
1174	Répássy	Kft.	Dobos Istv	án utca 2.	Budape	est			
1038	M5 Autós	szerviz	Meggy utc	a 8.	Budape	est			
<u>⊿</u> ktív							<u>о</u> к	<u>M</u> é	gse

56. ábra a Kapcsolat létrehozása parancstábla

A panel két-két tábla-listát és a tábla első öt rekordjának megjelenítésére szolgáló területet foglal magába. Utóbbin kattintással jelölhetjük ki a kapcsoló szerepre szánt mezőt. A táblákat, a kapcsolatban betöltött funkciójuktól függetlenül, tetszőleges sorrendben adhatjuk meg.

A Kapcsolat létrehozása parancstábla megnyitásakor a bővítmény a panel felső részébe az aktív táblát automatikusan betölti és kijelöli a kurzort tartalmazó mezőjét. A második tábla kiválasztását követően a bővítmény automatikusan kijelöli azt a mezőt, amelynek neve azonos az első táblában kiválasztott mezőével. Ha nem talál ilyet, akkor az első mezőt.

A kapcsolat létrehozását adatnézetben a kapcsoló szerepre szánt mező nevének menüjéből is kezdeményezhetjük. A *Kapcsolat létrehozása…* utasítás kiadását követően a bővítmény betölti a fent ismertetett panelt az aktív táblával és a kijelölt kapcsoló mezővel.

A bővítmény rendelkezik egy kapcsolat-kezelő parancstáblával is, amelyet a *Tervezés, Kapcsolatok, Kapcsolatok kezelése* utasítással jeleníthetünk meg. A panel funkciói: új kapcsolat létrehozása, a kijelölt kapcsolat szerkesztése és törlése. Az erről panelról indított létrehozás-deklaráció is a *Kapcsolat létrehozása* parancstáblán történik.

Kapcsolato	ok kezelése			_		\times
Létrehozá	s Szerkesztés Törlés					
Aktív	1. tábla 🔺	Számosság	Szűrő iránya	2. tábla		
lgen	autohasználat [rendszám]	Több-az-egyhez (*:*)	<< lde: autohasználat	autók [rendszá	m]	
lgen	autohasználat [személyi szám]	Több-az-egyhez (*:*)	<< lde: autohasználat	munkatársak [s	személyi	szám]
Igen	javítások [rendszám]	Több-az-egyhez (*:*)	<< lde: jav ítások	autók [rendszá	m]	
lgen	javítások [szerviz]	Több-az-egyhez (*:*)	<< lde: jav ítások	szerviz [szerviz]	
					Bez	árás

57. ábra a kapcsolat-kezelő panel

Adatnézetben a több oldali kapcsoló mező nevének menüjében megjelenő Ugrás a kapcsolódó táblához utasítással a kapcsolat egy oldali mezőjét aktiválhatjuk.

Kapcsolatnézetben a kapcsolat létrehozása egérrel történik: az egy oldali tábla kapcsoló szerepre kiválasztott mezőjét áthúzzuk a több oldali tábla kapcsolónak szánt mezőjére. Ha az egy oldali kapcsoló mező minden bejegyzése egyedi és nem tartalmaz üres bejegyzéseket, akkor a kapcsolat létrejön, amelyet a táblák között megjelenített kapcsolat-szimbólum mutat.



58. ábra a nem megfelelő kapcsolómező gerjesztette hibaüzenetek kapcsolatnézetben

Ha elfeledkeztünk a kapcsolatokról, akkor kimutatás készítésekor, a segédablak LÉTREHOZÁS... vezérlőjével pótolhatjuk a mulasztást. Az utasítás kiadása után egy kétsoros panel felső sorában a több oldali-, alsó sorában az egy oldali táblát és kapcsoló mezőjét kell listából kiválasztanunk. Ha a sorrend nem megfelelő a bővítmény megcseréli a definícióban a táblákat. Erről a korrekcióról a parancstábla alján kapunk egy félrevezető információt.

		?	Х
zlopok kivála	asztása		
	Oszlop (külső):		
\sim	nő		\sim
	Kapcsolódó os <u>z</u> lop (elsődleges):		
\sim	nő		\sim
smétlődő ért oszlopban c	ékeket tartalmaz. Ha kapcsolatot sz sak egyedi értékek lehetnek.	eretne lét	rehozni
	ОК	Mé	ase
	smétlődő ért	zlopok kiválasztása Oszlop (külső): Nő Kapcsolódó os <u>z</u> lop (elsődleges): Nő smétlődő értékeket tartalmaz. Ha kapcsolatot sz oszlopban csak egyedi értékek lehetnek.	? zlopok kiválasztása Oszlop (külső): Nő Kapcsolódó os <u>z</u> lop (elsődleges): Nő smétlődő értékeket tartalmaz. Ha kapcsolatot szeretne lét oszlopban csak egyedi értékek lehetnek.

59. ábra a segédablak LÉTREHOZÁS... vezérlőjével megjeleníthető panel

A panel bal sarkában álló Kapcsolatok kezelése... utasítással egy újabb kapcsolat-kezelő parancstáblával ismerkedhetünk meg, amelynek funkciói azonosak az előbb ismertetett paneléval, kiegészítve az automatikus-kapcsolat modult indító Automatikus felismerés... parancsgombbal.

Kapcsolato	ok kezelése		? ×
Állapot	Tábla 🔺	Kapcsolódó keresési tábla	Új
Aktív	nők (férfi)	férfiak (férfi)	A <u>u</u> tomatikus felismerés
			<u>S</u> zerkesztés
			<u>A</u> ktiválás
			<u>I</u> naktiválás
			Törlés

60. ábra a 59. ábra parancstáblájának Kapcsolatok kezelése... vezérlőjével megjeleníthető panel

Összegezve tapasztalatainkat, a kapcsolat létrehozására és kezelésére két különböző panel-páros áll a rendelkezésünkre, egy bővítmény ablakában és egy másik a program ablakában.

a kapcsolat törlése és megszakadása

A kapcsolat törléséről beszélünk, ha a felhasználó távolítja el a létező kapcsolatot. Ezt megteheti a kapcsolat-kezelő parancstáblákon, vagy kapcsolatnézetben a kapcsolat vonalának menüjéből. Kapcsolatnézetben a kijelölt kapcsolat a *Delete* billentyűvel is törölhető.

Ha a kapcsolatot a bővítmény távolítja el, akkor a kapcsolat megszakadásáról beszélünk.

Mező törlésekor a bővítmény a művelet végrehajtásához megerősítést kér, de nem figyelmeztet a mező esetleges kapcsoló szerepére. Így a felhasználó, akarva-akaratlanul, használhatatlanná teheti az összeköttetést. A PowerPivot a mező hiányának észlelését követően, eltávolítja a sérült kapcsolatot.

Ha az importált tábla kapcsoló mezőjét nem a törlés funkcióval, hanem a tábla tulajdonságainak módosításával távolítjuk el, akkor a bővítmény figyelmeztet: az érintett kapcsolat törölve lesz a tulajdonságok frissítése követően.

Data Analysis Expressions

a DAX képlet jellemzői

A bővítmény képleteiben használandó operátor- és függvény-készletet, valamint a használatukat leíró szemantikai- és szintaktikai szabály-rendszert "Data Analysis Expressions", röviden DAX-nak nevezzük. Az angol kifejezés jelentése "adatelemző kifejezések".

A DAX képletnek is egyenlőségjellel (=) kell kezdődnie, de az előjel (-, +), mint képlet-jelző a bővítményben hibát eredményez!

A bővítmény és a program aritmetikai (^, *, /, +, -), összehasonlító (<, <=, =, >=, >, <>) és szövegösszefűző (&) operátorai azonosak, de a logikai műveletek deklarálása már eltérő.

operátor	művelet
NOT	logikai tagadás
&&	logikai ÉS
I	logikai VAGY
IN	VAGY láncolás



Az IN szócskával csoportot határozhatunk meg, amelynek elemei (konstansok és kifejezések) logikai VAGY viszonyban állnak egymással. Az elemeket kapcsos zárójelek között, pontosvesszővel elválasztva kell felsorolni. A szövegeket és a dátumokat idézőjelzni kell. Például, IN { "Vác" ; "Érd" ; "Ózd" ; "Fót" }. Az IN műveletet VAGY láncolásnak is nevezhetjük, mert a csoportot a VAGY operátorral is képezni tudjuk: "Vác" || "Érd" || "Ózd" || "Fót". Tehát a DAX-ban van NOT és van IN operátor, de NOT IN operátor nincs! Hiába logikus, a NOT IN{ "Vác" ; "Érd" ; "Ózd" ; "Fót" } kifejezés a DAXban érvénytelen! De erről majd a szűrő függvényeknél részletesen beszélek. Majd látni fogjuk azt is, hogy a logikai operátorok mellett a DAX-nak is vannak logikai függvényei.

DAX-képletben a százalék jel (%) nem használható.

A dátum-konstansokat, mint az Excelben, idézőjelek között kell beírnunk. A bővítmény képleteiben is használhatjuk a perjeles formátumot: évszám két számjeggyel, perjel, hónapszám vezető nulla nélkül, perjel, napszám vezető nulla nélkül (éé/h/n). A perjel helyett kötőjelet is alkalmazhatunk.

művelet	Excel	DAX
= üres + üres	0	
= üres * 15	0	
= üres * üres	0	
= 15 / 0	#ZÉRÓOSZTÓ (#DIV/o!)	œ
= 15 / üres	#ZÉRÓOSZTÓ (#DIV/o!)	ω
= üres / 15	0	
= üres / o	#ZÉRÓOSZTÓ (#DIV/o!)	
= o / üres	#ZÉRÓOSZTÓ (#DIV/o!)	NaN
= üres / üres	#ZÉRÓOSZTÓ (#DIV/o!)	
= üres üres	#ÉRTÉK! (#VALUE!)	HAMIS (FALSE)
= üres && üres	#ÉRTÉK! (#VALUE!)	HAMIS (FALSE)

A képletben szereplő üres bejegyzések kezelése több ponton eltér a programban megszokottól.

A táblázatból leszűrhető legfonosoabb következtetés: a DAX képlet kiértékelése "üres eredmény"re is vezethet. És ez nem csak üres bejegyzésű rekordok feldolgozásakor fordulhat elő! Másként fogalmazva: nem minden DAX képlet ad eredményt.

^{62.} ábra eltérések az üres cellák illetve az üres bejegyzések kezelésében (a DAX oszlopban álló üres cellák üres eredményt jelentenek)

A műveletek végrehajtási sorrendje megegyezik a programban megszokottal. A bővítmény ablakában nem működik az egyetlen függvényt tartalmazó képletekben a függvény második zárójelének automatikus bevitele.

nevek a DAX képletében

A DAX képletében névvel azonosított objektumok a táblák, a mezők és az egyéni összesítések. Az objektum-nevek nem tartalmazhatnak vezető, illetve záró szóközt, vezérlő karaktereket és a . , ;':/*|?&%\$!+=()[]{}<> karaktereket.

A tábla és az egyéni összesítés nevének az adatbázisban, a mező nevének a táblában kell egyedinek lennie. A táblaneveket aposztrófok között, a mezőneveket és az egyéni összesítések neveit szögletes zárójelek között kell a képletbe beírni. Ha a táblanév nem tartalmaz ékezetes karaktert és szóközt, akkor az aposztrófok elhagyhatók.

A képletben a mezőnevek állhatnak önállóan vagy a táblanévvel együtt: [mezőnév] vagy, 'táblanév'[mezőnév]. Utóbbi forma használata, számos függvény esetében, előírás. A bővítmény ezt az azonosítót "minősített névként" emlegeti.

A bővítmény nem tartalmaz név-követő szolgáltatást, ezért egy név módosítást követően a felhasználónak kell az objektum minden előfordulásában az új nevet megadnia!

adattípusok konvertálása a képletben

A képlet kiértékelése közben a bővítmény kísérletet tesz a nem megfelelő típusú adat átalakítására. Ha a konvertálás sikeres, akkor a művelet végrehajtásra kerül, különben hibaüzenetet kapunk.

		Egész szám	Tizedes tört szám	Pénznem	Dátum
Egész szám	+		Tizedes tört szám	Pénznem	Dátum
	-		Tizedes tört szám	Pénznem	Egész szám
	*		Tizedes tört szám	Pénznem	Egész szám
	1		Tizedes tört szám	Tizedes tört szám	Tizedes tört szám
Tizedes tört szám	+			Tizedes tört szám	Dátum
	-			Tizedes tört szám	Tizedes tört szám
	*			Pénznem	Tizedes tört szám
	1			Tizedes tört szám	Tizedes tört szám
Pénznem	+				Dátum
	-				Tizedes tört szám
	*				Pénznem
	1	Pénznem	Pénznem		Pénznem
Dátum	+				
	-	Dátum	Dátum	Dátum	
	*				
	1			Tizedes tört szám	

63. ábra

különböző típusú adatokkal végzett aritmetikai műveletek eredményének adattípusa

A táblázat a különböző típusú adatokkal történő aritmetikai műveletek eredményének típusát mutatja. A műveleti jel bal oldalán a sor adattípusa, jobb oldalán az oszlop adattípusa áll. Tehát a táblázat első sora mutatja, milyen adattípusú eredményt kapunk, ha egészhez törtet adunk, ha egészhez pénzt adunk, ha egészhez dátumot adunk. A táblázat második sorában az áll, milyen eredményt kapunk ha egészből törtet, ha egészből pénzt, ha egészből dátumot vonunk le...

A táblázat tehát minden lehetséges variációt kétszer tartalmaz. Ha az eredmény típusára a két elem felcserélése nincs hatással, akkor az adott páros második előfordulásában, az áttekinthetőséget javítva, az eredményt már nem tüntettem fel. A szövegként megadott számok, egésznek vagy törtnek minősülnek. A logikai értékek ebből a szempontból egész számoknak tekinthetők.

számított objektumok

számított mező fogalma

A bővítményben három, DAX képleten alapuló, objektumot hozhatunk létre: [1] számított mezőt, [2] automatikus összesítést és [3] egyéni összesítést. Mindhárom objektumot névvel azonosítjuk. Az összesítések statisztikai objektumok, egy meghatározott rekord-csoport adataival kalkulálnak, a számított mező képlete viszont csak egyetlen rekord adatait dolgozza fel.

A számított mező egy névvel azonosított, felhasználói képleten alapuló, származtatott objektum a táblában. Másként fogalmazva, a számított mező egy DAX képlet, amelyet a bővítmény a tábla minden egyes rekordjában elhelyez és kiszámol. A számított mező egyenrangú a tábla többi mezőjével, így másik számított mező képletében is szerepelhet.

Megjelenítésükben a bővítmény nem tesz különbséget a tábla forrás- és származtatott mezői között. Ennek ellenére létezhet olyan helyzet, amelyben fontos lehet a megkülönböztetés, ezért a névben, például csupa nagybetűket használva, jelezhetjük a számított mezőt.

Ha a számított mező képletében egy másik tábla is szerepel, akkor a bővítménynek a tábla minden egyes rekordjához meg kell keresnie a másik tábla kapcsolódó rekordjait. Ezt a funkciót látja el a RELATED függvény, amelynek egyetlen argumentuma a másik tábla egyik mezője. Természetesen a táblák közötti kapcsolatok deklarálásával biztosítani kell a függvény számára az összetartozó adatok kikereshetőségét. Nézzünk egy példát! Az alábbi ábra, "városok" táblájának, BERUHÁZÁS mezője számított mező. Képlete: = [szorzó] * RELATED('régiók'[összeg]).



64. ábra a példákban szereplő "BERUHÁZÁS" és a "VÁROSOK SZÁMA" számított mezők

A RELATED függvény tehát kikeresi a "régiók" táblából az aktuális "város" régiójában rendelkezésre álló "összeget". Ha azonban a régiók városainak számát akarjuk meghatározni, akkor már a RELATED-TABLE függvényt kell alkalmaznunk a VÁROSOK SZÁMA számított mezőben. Képlete: =COUNTROWS (RELATEDTABLE('városok')). A RELATEDTABLE függvény nem egyetlen adatot, hanem egy egész táblázatot ad eredményül: az aktuális "régió" a városok táblában álló, összes kapcsolódó rekordját tartalmazó táblát. Ennek a virtuális táblának a rekordjait számlálja meg a COUNTROWS függvény.

számított mező létrehozása

Adatnézetben a mezők után álló, üres cellákat tartalmazó oszlop a számított mező létrehozásának eszköze. Az oszlop egy cellájára, majd a szerkesztőlécre kattintva, hozzáláthatunk képlet összeállításához. A kurzor az F2 funkcióbillentyűvel is elhelyezhető a szerkesztőlécen. A szalagról a Tervezés, Oszlopok, Hozzáadás utasítással indíthatjuk a műveletet.

A nevek beírását a bővítmény névkiegészítője segíti. Listájában a le, fel nyilakkal lépegethetünk. A kiválasztott elemet a *Tab* billentyűvel vagy rámutatással és dupla kattintással tudjuk beíratni a képletbe. A listában felül állnak a begépelt karakterekkel kezdődő függvénynevek és őket követik az adott betűkkel kezdődő objektum-nevek.

				X	\checkmark	f _x =	=H					/				
J.	régió	ę.		ös	szeg		(k)	HASONEFIL	TER		Oszlop h	oz				
1	Dél-Alföld					3,9		HASUNEVA	LUE	43		7				
2	Dél-Dunántúl					3,9	00	IOOK		34		1				
3	Észak-Alföld						-	× 🗸 .	f _* ='						[
		1	rég	ιó			Pa 💽	összeg		·	megyék'	1A	*	Oszlop ho	02	
		1	Dé	I-Alt	öld				3,9		régiók' vézezek		43		7	
		2	Dé	l-Du	nánt	túl			3,9		Valusuk		34		(
		3	Ész	ak-	Alföl	d_				*	X 🗸	fx =[
							/ ré 1 Dé 2 Dé	gió Èl-Alföld Èl-Dunánt	ا úا	6	összeg	3,9 3,9 3,9	ן ני ני	régió] VÁROSOK összeg]	SZÁMA] J+	szlop hoz
							3 És	zak-Alföld	<u> </u>	_		3,0	_	~	56	

65. ábra a függvények, a táblák és a mezők jelzése a névkiegészítő listájában

A bővítmény ablakában használhatjuk a kattintásos névbevitelt is. A tábla nevét a tábla fülére-, a mező nevét tetszőleges bejegyzésére-, az egyéni összesítés nevét a számítási területen álló képletére kattintva írathatjuk be a szerkesztés alatt álló kifejezésbe.

A bővítmény már az összeállításakor elemzi a születő képletet. Meghatározott karakterek (szóköz, pontosvessző, zárójel...) bevitelét követően piros hullám-vonallal aláhúzza a hibás elemeket és a függvény-neveket zöld színnel írja át, sőt bizonyos esetekben, hibaüzenet megjelenítésével, még a szerkesztést is megszakítja. A névkiegészítő listáját nem csak az aktuálisan beírt karakterek, de a kurzor pozíciója is meghatározza. A bővítmény csak azoknak az objektumoknak a nevét jeleníti meg, amelyek a kurzor pozíciójában elfogadhatók.

f _X EÖ1:=COU	f_X EÖ1:=COUNTROWS(CONTAINSROW(D ; { [kategóna] }							
tegória 🔽 osztály 🔽 érték 🔽 Számított oszlop: 1 🔽 Oszlop hozzáadása								
)	1	35	HAMIS					
Powe	Power Pivot bővítmény az Excel programhoz X							
A kifejezés nem érvényes, vagy hiányosnak tűnik. Ellenőrizze és javítsa ki a kifejezést. A program elérte a bemenet végét.								
	3	53	IGAZ					
	3	27	HAMIS					

66. ábra hibajelzés a szerkesztés közben és a szerkesztést megszakító panel

A szerkesztőlécen dolgozva használhatjuk a kijelölés (Shift+Jobbra nyíl/Balra nyíl, Shift+Home/End, Ctrl+a) és a vágólap-műveletek (Ctrl+x, Ctrl+c, Ctrl+v) billentyűparancsait. Utóbbi művelet-csoport a szerkesztőléc menüjében is megtalálható, Az összes kijelölése utasítással együtt. A szerkesztőléc magasságát a felhasználó, az Excelben megismert módokon szabályozhatja. Egyedül a dupla kattintásos magasság-állítás nem működik az alsó szegélyen.

A szerkesztés az Esc billentyűvel vagy a képletszerkesztő blokk Mégse nyomógombjával szakítható meg. A képlet kiértékelését az Enter billentyűvel vagy a képletszerkesztő blokk OK gombjával kezdeményezhetjük. Az egyetlen függvényből álló képletek záró zárójelének automatikus bevitele a bővítmény ablakában nem működik, de az argumentum nélküli függvények név-kiegészítős bevitelekor a PowerPivot mindkét zárójelét beírja a képletbe.

				📰 adatok	
				📰 szöveg	
				💷 szám	
[ÖSSZEG]	• <i>f</i> _x	=[szöveg] + [szár	n]	📰 ÖSSZEG	
🖌 szöveg 💽	szám 🔽	ÖSSZEG 📣 💌	Oszlop hozzáadása		
1 koko	4	#HIBA	🔶 (Ctrl) 🔻	(
2 lili	8	#HIBA	Hiba megjelenítése.		
3 bubu	8	#HIBA	Ugrás az első hibára		
4 fefe	3	#HIBA		-6-	
5 vivi	2	#HIBA			
városok megyél	k 🛛 régiók 🚺 🕹 ad	atok			
Rekord: 11 1	1., összesen 7	+ +1			

67. ábra kiértékelhetetlen képlet jelölése adat- és kapcsolatnézetben

A kiértékelhetetlen kifejezés #HIBA bejegyzést eredményez a tábla minden rekordjában. A hibát pici rombusz jelöli [1] a tábla fülén, [2] a mezőnévben és [3] a hibás mező aktív bejegyzése mellett. A színes rajzocskára mutatva a bővítmény figyelmeztető üzenetet jelenít meg. A hiba leírását a számított mező tetszőleges cellájára kattintva, a bejegyzés jobb oldalán álló gomb *Hiba megjelenítése…* parancsával jeleníthetjük meg. A menü másik, *Ugrás az első hibára* utasítása a mező első bejegyzését aktiválja.

A képlet szerkesztésének befejezése után a bővítmény rekordonként kiértékeli a kifejezést és az eredményeket a memóriában tárolja. Ha a művelet minden rekordban sikeres, akkor az eredmények megjelennek a mezőben, melynek automatikus neve: Számított mező: <sorszám>. A számított mező létrehozása után a bővítmény a tábla jobb oldalán ismét megjeleníti az Oszlop hozzáadása feliratú oszlopot. A mezőnév menüjéből az automatikus név felhasználói névre cserélhető.

A számított mező másolásakor a bővítmény a képletek eredményét helyezi el az új táblába. A számított mezőt a helyi menü Oszlopok törlése utasításával vagy a Delete billentyűvel törölhetjük.

képletek újra számolása

A bővítmény a szokásos beállítások mellett automatikusan újra számolja a képleteket: [1] a táblák frissítése után, [2] objektumok átnevezése után, [3] kapcsolat létrehozása, módosítása és törlése után, [4] új számított objektum létrehozása vagy meglévő számított objektum képletének módosítása után, [5] a PowerPivot-kimutatás szűrőfeltételeinek változása után.

Amennyiben a tábla frissítése során, olyan új adat olvasódna be, amely hibát eredményezne egy képlet újra számolásakor, akkor a bővítmény megtagadja a tábla frissítését!

A felhasználó a Tervezés, Számítások, Számítási beállítások, Kézi számítási mód paranccsal letilthatja az automatikus újraszámítás szolgáltatást. Ebben az esetben a bővítmény csak a Tervezés, Számítások, Számítási beállítások, Azonnali kiszámítás utasításra frissíti a képletek eredményét.



68. ábra az automatikus újraszámítás tiltása

Az automatikus képlet-újraszámítás letiltása csak az aktív PowerPivot munkafüzet képleteinek kezelésére van hatással, a bővítmény általános működését nem befolyásolja. A manuális újraszámítási mód beállítás a munkafüzettel együtt mentődik és a fájl következő megnyitásakor is érvényesül.

automatikus összesítés

Amikor a kimutatás értékek területére mezőt helyezünk, akkor közvetve egy képletet hozunk létre: STATISZTIKAI FÜGGVÉNY(<mezőnév>). Ezt a képletet értékeli ki a program a sor és oszlopmezők tételei által meghatározott adatcsoportokon. A PowerPivot-kimutatás-modulja lehetőséget biztosít, olyan statisztikai objektum létrehozására is, amelynek képletét a felhasználó állítja össze. A két objektum funkciója azonos, de nem vehetjük őket egy kalap alá, mert néhány tulajdonságuk különböző. Ezért az előbbit nevezzük automatikus-, az utóbbit egyéni összesítésnek.

A bővítmény a két objektumot, hol megkülönböztetés nélkül mértéknek, hol megkülönböztetve implicit- és explicit mértéknek, vagy mértéknek és egyéni összesítésnek nevezi. Mi ragaszkodjunk a saját fogalmainkhoz, mert ez a megértés alapfeltétele!

Tehát automatikus összesítés az elemzendő mező az értékek kimutatás-területre helyezésével hozható létre. A megjelenített statisztikai érték [1] egész szám, tizedes tört és pénznem adattípusú mező esetén a bejegyzések összege, [2] szöveg, logikai és dátum adattípusú mező esetén a bejegyzések darabszáma.

Ettől eltérő statisztikai értéket az értékek kimutatás-területen álló mező menüjének Értékmező beállításai..., Értékösszegzési szempont, Értékmező összegzésének alapja listából választhatunk. A lista egyetlen újdonsága az Eltérők darabszáma, amely a mező bejegyzéseinek darabszáma, az ismétlődések nélkül. Vagy másként fogalmazva, az egyedi bejegyzések darabszáma.

Az automatikus összesítés nevét a bővítmény adja, amely a statisztikai érték- és a mező nevéből áll. A lista felett álló, Egyéni név vezérlővel a felhasználó egy második nevet is megadhat, de ez a név csak a szülő-kimutatásban és a kimutatás értékek területén lesz megjelenítve.

stat. érték neve	függvény	generált név
Összeg	SUM	Összeg - <mezőnév></mezőnév>
Darab	COUNTA	Elemszám - <mezőnév></mezőnév>
Átlag	AVERAGE	Átlag - <mezőnév></mezőnév>
Maximum	MAX	Maximum - <mezőnév></mezőnév>
Minimum	MIN	Minimum - <mezőnév></mezőnév>
Szórás	STDEV.S	Szórás - <mezőnév></mezőnév>
Szórásp	STDEV.P	Sokasági szórás - <mezőnév></mezőnév>
Var	VAR.S	Variancia - <mezőnév></mezőnév>
Varp	VAR.P	Sokasági variancia - <mezőnév></mezőnév>
Eltérők darabszáma	DISTINCTCOUNT	Eltérők darabszáma - <mezőnév></mezőnév>

^{69.} ábra automatikus összesítések

Mint, azt már az előző fejezetben említettem, a bővítmény adatnézetben a számítási területen is megjeleníti az összesítéseket. A szokásos beállítások mellett azonban csak az egyéni összesítéseket látjuk. Ezért először a PowerPivot ablak "Fájl" menüjéből a Váltás speciális módra utasítással jelenítsük meg a szalag Speciális lapját, majd adjuk ki róla az Implicit mértékek megjelenítése parancsot.

Adatnézetben az összesítések statisztikai értékei a teljes táblázatra, illetve szűrés esetén a szelektált rekord-halmazra vonatkoznak. Az automatikus összesítések jobb oldalán íves-nyilas szimbólum áll. A létrehozott automatikus összegzés szuverén objektum, az elemzett mező táblájában tárolódik és szülő-kimutatásának törlése után is elérhető marad.

[5	számla] 👻 🗙 🖌 $f_{\!x}$	='		
1	rendszám	'autók'		izből 💌
1	JAF-794	budapesti lakasok		2001. 07. 07,
2	WYA-469	Javitások' [Elemca	ám - rendszámi	2001. 07. 08
3	WYA-471	S 'javítások'[Eltérők	darabszáma – rendszár	2001. 07. 14.
4	JAF-796	szervizek'		2001. 07. 26.
5	WYA-467		2001. 07. 21.	2001. 07. 24
6	SBA-444		2001. 07. 21.	2001. 07. 31
	Elemszám - rendszán	n: 2652 🧳		
	Eltérők darabszáma -	- rendszám: 178 🛛 🛷		
		\sim		
		javítások irendszám szervizbe szervizből számla sajáthibás szerviz fźz Elemszám - rendszáma –	rendszám 🕹	

70. ábra

az automatikus összesítés a számítási területen, a névkiegészítőben és kapcsolatnézetben

A mező törlése természetesen együtt jár az őt elemző automatikus összesítések törlésével. A számítási területen álló automatikus összesítést a menüjéből kiadott utasítással vagy a *Delete* billentyűvel törölhetjük. A törlés a kimutatásból is eltávolítja az összesítést.

Gyakran elemzett, numerikus mezők szokásos statisztikai műveletét (összeadás) a felhasználó módosíthatja: a bővítmény ablakában jelöljük ki a mezőt, majd a Speciális, Összegzés szempontja listából válasszuk ki az értékek területre helyezéskor, automatikusan megjelenítendő statisztikai értéket!

egyéni összesítés

Az egyéni összesítés egy felhasználói képleten alapuló, statisztikai objektum, amely az értékek kimutatás-területen áll. Képletét a felhasználó hozza létre. A mezőlistán és a névkiegészítőben megjelenő univerzális név mellett lehetőségünk van egy lokális név használatára is, amely csak az adott kimutatásban és a kimutatás mezőlistájának értékek területén jelenik majd meg.

Az egyéni összesítés képletét a bővítmény a felhasználó által kiválasztott táblában tárolja és ennek a táblának a számítási területén is megjeleníti: <egyéni összesítés neve><kettőspont><szóköz> <érték a teljes táblára vonatkozóan>. Az egyéni összesítés képletét egy másik egyéni összesítés képletében az univerzális nevével szerepeltethetjük: '<táblanév>'[<egyéni összesítés neve>].

Egyéni összesítést létrehozhatunk a bővítmény ablakában, adatnézetben vagy az Excel ablakában, egy PowerPivot-kimutatás készítése közben.

Lássunk egy példát! A "budapesti lakások" táblában meg szeretném határozni a budapesti lakások átlagos négyzetméter-árát, tízezer forint pontossággal. Az egyéni összesítést adatnézetben hozom létre.

Először kiválasztom az objektum tárolására szánt táblát: "budapesti lakások", kattintok a számítási területen, majd egérrel vagy az F2 billentyűvel elhelyezem a kurzort a szerkesztőlécen. A deklaráció szintaktikája: <univerzális név><kettőspont><egyenlőségjel><képlet>. Tehát "átlag nm ár" :=ROUND(SUM ([ár]) / SUM([alapterület]) ; -4). A képletekben szereplő ROUND függvény funkciója és szintaktikája azonos a program KEREKÍTÉS függvényével. A DAX SUM függvénye azonban már különbözik az Excel SZUM függvényétől: csak egyetlen argumentumot határozhatunk meg és ez csak egy mezőnév lehet.



71. ábra az egyéni összesítés a számítási területen, a névkiegészítőben és kapcsolatnézetben

A szerkesztés befejezése után a cellában megjelenik az egyéni összesítés értéke a teljes táblára vonatkozóan. A statisztikai érték formátumát nem csak a szalagon, de az objektum menüjéből is megadhatjuk. Ez a beállítás természetesen a kimutatásban is érvényesül majd. Az egyéni összesítés menüje tartalmazza még a törlés, a leírás megadására szolgáló és az objektum megjelenítését tiltó parancsot. Az adatnézetben törölt egyéni összesítés a kimutatásból is törlődik.

A Kezdőlap, Számítások, AutoSzum lista utasításaival az aktuális mezőt elemezhetjük a leggyakoribb statisztikai függvényekkel. Az automatikusan létrehozott képlet, az automatikus nevet is beleértve, szerkeszthető és így egy bonyolultabb egyéni összesítés alapja lehet.

Foglaljuk össze, az utolsó két ábra alapján, a két összesítés-típus megjelenítési sajátosságait! [1] Az automatikus összesítést elsősorban a nevéről ismerhetjük fel. [2] Adatnézetben a két típust az automatikus összesítést után álló kétirányú íves nyilacska különbözteti meg. [3] Ezt a picike ábrát a bővítmény kapcsolatnézetben is megjeleníti az automatikus összesítés neve után. [4] A névkiegészítőben mindkét típust a görög szumma (Σ) jelöli. [5] A PowerPivot mezőlistán csak az egyéni összesítések jelennek meg. A felsorolás tanulsága: ne adjunk az automatikus összesítés nevéhez hasonlító nevet az egyéni összesítésnek, mert ez esetleg zavart okozhat a névkiegészítő használatakor.

Példánkat folytatva, kimutatással kell meghatározzuk az egyes kerületek átlagos négyzetméterárának eltérését a budapesti átlagtól, tízezer forint pontosággal. A bővítmény ablakában adjuk ki a *Kezdőlap, Kimutatás* utasítást, majd a megjelenő panel segítségével, határozzuk meg az objektum helyét. Deklaráljuk a két sormezőt a mezőlistán: "városrész" és "kerület", majd jelenítsük meg az egyéni összesítés szerkesztésére szolgáló parancstáblát a PowerPivot, Számítások, Mértékek, Új mérték... utasítással.

Először a tárolásra szánt táblát kell kiválasztanunk: "budapesti lakások". Felülről lefelé haladva a következő mező a név megadására szolgál: "eltérés a bp-i átlagtól". A Leírás mezőben megjegyzést fűzhetünk az objektumhoz. A Képlet mező az egyéni összesítés képletének összeállítására szolgál. Függvény és objektumnevek bevitelekor itt is rendelkezésünkre áll a névkiegészítő.

A mező fölött álló fx feliratú nyomógombbal a Függvény beszúrása panelt jeleníthetjük meg. A parancstáblán, felülről lefelé haladva, először a függvény-kategoriák felsorolását találjuk, majd ezt követi a kategoriába tartozó függvények listája. A bővítmény a lista alatt megjeleníti a kiválasztott függvény argumentumlistáját és leírását is. Utóbbi azonos a névkiegészítőben olvasható ismertetővel. A függvényt az OK gombbal helyezhetjük el a képletben.

Mérték				?	\times			
Táblázat neve:	budapesti lakáso	ok .			~			
Mérték neve:	eltérés a bp-i átla	agtól						
Leírás:	tízezer forint pon	tosággal						
Képlet: <i>f</i> _x	Képlet ellenőrzé	se						
='budapesti laká SUMX(ALL('bu	sok'[átlag nm ár] · dapesti lakások')	ROUND(SUMX(ALL(bu ;[alapterület]);-4	udapesti lakások') ;	[ár]) /				
A kifejezés A program	A kifejezés nem érvényes, vagy hiányosnak tűnik. Ellenőrizze és javítsa ki a kifejezést. A program elérte a bemenet végét.							
Formázási beáll	ítások							
Kategória:								
Általános Szám		Szimbólum:	HUF		~			
Pénznem		Tizedeshelyek:	0 🜲					
IGAZ/HAMIS		Ezres csoportosítás	()					
			ОК	Mégse	е .:			

72. ábra az egyéni összesítés szerkesztésére szolgáló parancstábla

Az "eltérés a bp-i átlagtól" értéke a vizsgált terület lakásainak átlagos négyzetméter-ára és a budapesti átlagos négyzetméter-ár különbsége: ='budapesti lakások'[átlag nm ár] - ROUND(SUMX (ALL ('budapesti lakások'); [ár]) / SUMX(ALL('budapesti lakások'); [alapterület]); -4).

A képletekben szereplő SUMX függvénynek nincs a programban megfelelője. Az első argumentumával meghatározott tábla minden rekordjában kiértékeli a második argumentumával meghatározott kifejezést, majd az eredményeket összeadja. Az ALL függvény szintén a DAX specialitása. Az argumentumával meghatározott teljes táblát, vagy egy mező összes bejegyzését adja eredményül, tekintet nélkül a kimutatás elrendezésére és szűrőire. Az "eltérés a bp-i átlagtól" képlete nem tartalmazza a lakások átlagos négyzetméter-árának kiszámítási módját, hanem szerepelteti az ezt az értéket eredményező "átlag nm ár"-t.

A szerkesztés befejeztével a Képlet ellenőrzése parancsgombbal kérhetjük a képlet ellenőrzését. Hiba esetén a panel egy újabb területtel egészül ki, amely az ellenőrzés eredményének leírását tartalmazza.

A panel Formázási beállítások területén álló vezérlőkkel a statisztikai érték adattípusát és megjelenítési formáját adhatjuk meg.

A létrehozást követően az egyéni összesítés megjelenik a mezőlistán, a gazda-tábla mezői után, valamint a kimutatás értékek területén.

Az egyéni összesítések módosítására és törlésére szolgáló parancstáblát a PowerPivot, Számítások, Mértékek, Mértékek kezelése... utasítással jeleníthetjük meg.

szűrők a DAX képletben

A DAX-ban nincs tartomány-hivatkozás, de a műveletek hatókörét a képletben deklarált feltételekkel határozhatjuk meg. Ismerkedjünk meg ezzel a lehetőséggel egy példán keresztül! Az "üzletkötők" tábla egy vállalat ügynökeinek adatait tartalmazza. A cégnek budapesti kirendeltségein kívül számos nagyvárosban is van üzlete. Az üzletkötők négy kategoriába vannak sorolva: A, B, C és D. A tábla további mezői az üzletkötők belépési dátumát, a 2017-es és a 2018-as kötéseik összegét valamint havi fizetésüket tartalmazza.

	üzletkötő 🗾 🔽	kirendelt	tség 🔽	kategória 💌	belépett 🔽	bevétel 2017 🔽	bevétel 2018 💌	fizetés 🔽
1	Abonyi Emőke	Debrecer	ı	D	2010. 07. 15.	3 270 000 HUF	5 646 000 HUF	290 000 HUF
2	Adorján Sebestény	Budapest	t IV. kerület	D	2011. 12. 28.	3 958 000 HUF	3 596 000 HUF	605 000 HUF
3	Agócs Aranka	Szombath	nely	D	2014. 07. 08.	4 378 000 HUF	3 532 000 HUF	250 000 HUF
4	Almási Jolán	Budapest	t XIV. kerület	D	2018. 12. 28.	5 367 000 HUF	5 192 000 HUF	395 000 HUF
5	Arató Pál	Budapest	t XVIII. kerület	D	2011. 07. 07.	3 849 000 HUF	6 477 000 HUF	390 000 HUF
6	Asolti Kriszta	Szombath	nely	D	2011. 04. 18.	5 572 000 HUF	5 166 000 HUF	290 000 HUF
7	Bagi Lídia	Budapest	t X. kerület	D	2012. 06. 06.	4 060 000 HUF	3 943 000 HUF	380 000 HUF
8	Bakonyi Aurél	Budapest	t XIV. kerület	D	2009. 05. 05.	4 748 000 HUF	4 495 000 HUF	385 000 HUF
9	Balla Vince	Budapest	t XIII. kerület	D	2011. 01. 16.	4 337 000 HUF	5 444 000 HUF	525 000 HUF
10	Beke havi bértömeg		kategória 🚽	T)18. 10. 06.	6 061 000 HUF	6 212 000 HUF	330-000 HUF
11	Benk kirendeltség	ज्ञ	A	в	18 01 30.	⊿ Ei üzletköt ő	ik	000 HUE
	Budapest II. ke	rület	645 000 HU	F 615 000 H	UF	🗌 üzletk	ötő	
	Budapest III. k	erület	590 000 HU	F 655 000 H	UF	✓ kirend	leltség	T
	Budapest IV. k	erület	1 650 000 HU	F 1 285 000 H	UF		ória .	-
	Budapest VIII.	kerület	565 000 HU	F 790 000 H	UF		ona	
	Budapest X. ke	rület	320 000 HU	F 550 000 H	UF	🔄 belépe	ett	
	Budapest XI. k	erület	680 000 HU	F 990 000 H	UF	bevéte	el 2017	
	Budapest XIII.	kerület	1 185 000 HU	F 320 000 H	UF		1 2019	
	Budapest XIV.	kerület	375 000 HU	F 1 005 000 H	UF		12010	
	Budapest XV. k	erület	1 005 000 HU	F 440 000 H	UF	fizetés		
	Budapest XVIII	. kerület		575 000 H	UF	✓ f _X har	vi bértömeg	
	Budapest XXI.	kerület	625 000 HU	F 1 170 000 H	UF			

73. ábra a tábla, a kimutatás és a mezőlista

A "havi bértömeg" egyéni összesítés a legalább három éve a vállalatnál dolgozó és emelkedő éves árbevételt produkáló üzletkötők havi bérét összesíti.



74. ábra a "havi bértömeg" egyéni összesítés képlete

A SUMX DAX függvényt már ismerjük. Az első argumentumában álló FILTER függvény egy tábla feltételekkel kiválasztott rekordjait adja eredményül. Első argumentuma a táblát, második a feltételeket határozza meg. A képletben szereplő QUOTIENT és a TODAY függvények működése és szintaktikája megegyezik a program KVÓCIENS és MA függvényeivel.

A képlet tehát a "fizetés" mező bejegyzéseit adja össze. A bejegyzések egy részét a képlet és a kimutatás szűrői kizárják a vizsgálatból, az összeadandó bejegyzés-csoportokat pedig a kimutatás felépítése határozza meg. Tehát a kimutatás azoknak az üzletkötőknek a bérét adja össze, akik már legalább három éve a vállalatnál dolgoznak [képlet szűrője], emelkedő árbevételt produkáltak [képlet szűrője], valamelyik budapesti kirendeltségen dolgoznak [kimutatás szűrője] és A vagy B kategoriásak [kimutatás szűrője]. A kimutatás a béreket kirendeltségek és azon belül, kategoriánként adja össze [kimutatás szerkezete].

mezőhívatkozások

A relatív-, az abszolút és a vegyes hivatkozás a cella- és tartomány azonosítók értelmezésének három lehetséges módja a programban. Most ismerkedjünk meg a bővítmény hivatkozásaival, azaz a mezőnevek értelmezési módjaival!

1	dátum 🗾	bevétel 🗾	HÁNYAD 🔽	GÖNGYÖLÍTVE 💽
1	2016. 01. 01.	10	1,7%	10
2	2016. 02. 01.	60	10,3%	70
3	2016. 03. 01.	90	15,5%	160
4	2016. 04. 01.	10	1,7%	170
5	2016. 05. 01.	80	13,8%	250
6	2016_06.01.	30	5.2%	280

^{75.} ábra a "bevételek" tábla

A fenti ábrán látható tábla egy szervezet havi bevételeit tárolja, milliós nagyságrendben. A dátumok a rekordok rögzítésének idejét mutatják. A tábla további két származtatott mezője az adott rekord bevétele és az összes bevétel arányát számolja ki, illetve a bevételeket göngyölíti.

A HÁNYAD számított mező képlete: ='bevételek'[bevétel] / SUM('bevételek'[bevétel]). Tehát az adott rekord bevétele és a táblában tárolt összes bevétel hányadosa. Egyszerű képlet, de azért elgondolkodtató: a benne szereplő "[bevétel]" karakterlánc első előfordulásában a vizsgált rekord bevételét, tehát egyetlen bejegyzést, második előfordulásában a "bevétel" mező összes bejegyzését jelenti. A mezőhivatkozás értelmezésének módosulását nyilvánvalóan a SUM függvény okozta.

A második számított mező, ahogy az ábrán látjuk, a bevételeket göngyölíti. Képlete kicsit bonyolultabb, mint az előbbi: = SUMX(FILTER('bevételek'; [dátum] <= EARLIER([dátum])); [bevétel]). A SUMX függvény tábla argumentumát a FILTER függvény szolgáltatja. Ennek a virtuális táblának a "bevétel" mezőjében álló bejegyzéseket adja össze. A FILTER függvény első argumentuma a "bevételek" tábla, amelynek rekordjai közül válogatja le azokat, amelyek megfelelnek a második argumentumában megfogalmazott feltételnek.



76. ábra a "bevételek" tábla számított mezőinek képlete: HÁNYAD, GÖNGYÖLÍTVE

Most már csak a feltétel, [dátum] <= EARLIER ([dátum]), két "[dátum]" hivatkozásának értelmezése vár megfejtésre. Ennek érdekében tegyünk egy kis kitérőt és nézzünk egy másik példát az EARLIER függvény működésére.

J.	betűk 💽	számok 💽	ÖSSZEG 🛛 💌
1	a	4	20
2	b	2	10
3	c	5	25
4	d	1	5
5	e	3	15



Az ÖSSZEG számított mező képlete: =SUMX (karakterek; EARLIER ([számok])). Az eredményeket vizsgálva, azt látjuk, hogy minden rekordban a "számok" bejegyzésének ötszörösét kaptuk. A tábla

öt rekordot tartalmaz. Az első rekord képletének kiértékelésekor a bővítmény elhelyezte a feldolgozás alatt álló rekord "számok" bejegyzését minden rekordban, majd az öt darab négyest összeadta. A második rekord képletének kiértékelésekor a bővítmény elhelyezte a feldolgozás alatt álló rekord "számok" bejegyzését minden rekordban, majd az öt darab kettest összeadta. És így tovább. Az EARLIER(<mezőnév>) kifejezés, tehát a kiértékelés alatt álló rekord mezőbejegyzését jelenti.

Visszatérve eredeti példánkhoz, azt látjuk, hogy az összeadandó bevételek folyamatosan növekvő halmazát a kronologikusan álló dátumok biztosítják: – Válogasd le azokat a rekordokat, amelyek dátuma azonos vagy korábbi az én dátumomnál! – utasítaná a bővítményt az éppen kiértékelés alatt álló rekord. Ez a tábla rekordszámával azonos számú összehasonlítás-műveletet jelent minden egyes rekord képletének feldolgozásakor. Összefoglalva: az EARLIER függvény argumentumaként álló "[dátum]" a képletet tartalmazó rekord "dátum" mezőjének bejegyzése, az első "[dátum]" hivatkozás pedig a tábla összes dátum bejegyzését jelenti, de nem úgy, mint az első képlet SUM([bevételek]) esetében az elemek összességét, hanem minden egyes bejegyzést külön-külön.

PowerPivot-kimutatás

a bővítmény kimutatás-modulja

A bővítmény kimutatás-modulja több szolgáltatásában eltér a hagyományos kimutatástól. Ezért célszerű az elnevezésben is megkülönböztetni őket. Tehát a bővítmény ablakának *Kezdőlap, Kimuta*tás parancsával létrehozott objektum a PowerPivot-kimutatás.

Mivel a PowerPivot-kimutatás mindig a teljes "adatbázist" vizsgálja, ezért a létrehozás-parancs kiadása után csak az új kimutatás helyét kell meghatároznunk a *Kimutatás létrehozása* panelon. A *Létező munkalapon* lehetőséget választva, először a *Hely* mező után álló ikonra kell kattintani, majd a *Tartomány* (Sic!) *kiválasztása* (Range Selection) panel megjelenítése után kattintással határozhatjuk meg a létrehozandó kimutatás bal felső celláját.

A kimutatás-terület és a mezőlista megjelenítése jelzi, hogy a bővítmény létrehozta a memóriában az objektum kezeléséhez szükséges adatszerkezetet. A bővítménynek a szalagon nincs külön kimutatás-lapja. A folyamatosan látható *PowerPivot-* és az aktív kimutatás mellet ideiglenesen megjelenített *Kimutatáseszközök, Elemzés és Tervezés* lapok tartalmazzák a PowerPivot-kimutatáshoz kapcsolódó utasításokat, amelyek néhány apróbb részletben különböznek csak a megszokott felülettől.

A bővítmény a táblák frissítését követően az elemzésükre létrehozott kimutatásokat automatikusan frissíti.

PowerPivot-kimutatás beszúrását a program-ablakból is kezdeményezhetjük a Beszúrás, Táblázatok, Kimutatás utasítással. A megjelenő Kimutatás létrehozása paranctáblán A munkafüzet adatmodelljének használata lehetőséget kell választanunk, majd a panel vezérlőivel meg kell határoznunk a kimutatás helyét.

a PowerPivot-kimutatás segédablaka

A Kimutatásmezők feliratú segédablakot csak aktív PowerPivot-kimutatás esetén látjuk. A panel megjelenítése illetve elrejtése a Kimutatáseszközök, Elemzés, Megjelenítés, Mezőlista paranccsal történik.

A segédablak neve alatt álló Aktív és Mind vezérlőkkel a kurzor által kijelölt kimutatás táblái (Aktív), illetve a PowerPivot adatbázis összes táblájának és a munkafüzet összes adatbázis-táblázatának (Mind) megjelenítése között választhatunk. A táblák nem a bővítmény ablakában elfoglalt helyük sorrendjében, hanem a nevük alapján, ABC sorrendben követik egymást.

A kapcsolatokkal egyesített táblák csoportot képeznek a mezőlistán. Az egy csoportot alkotó táblákat a bővítmény egymás alatt jeleníti meg a mezőlistán és vízszintes vonallal választja el más csoportoktól illetve önálló tábláktól.

Kimutatásmezők	- ×
Aktívak Mind	
Válassza ki a jelentésbe felvenni kívánt mezőket:	∯ ▼
Keresés	P
▷ 🗄 adatok	
▷ 🌐 egyik	
▷ 🏢 másik	
▷ 🌐 harmadik	
\sim \sim	

78. ábra táblák a segédablakban

A képen látható tábla-szimbólumok jelentése a következő. [1] Hengeres: importált vagy vágólappal létrehozott tábla, [2] henger nélkül: adatbázis-táblázatból létrehozott tábla, [3] felső csík nélkül: adatbázis-táblázat, amelyből nem készült tábla.

A mezőket a táblanevek előtt álló pici háromszöggel jeleníthetjük meg. Sorredjük azonos a táblában elfoglalt pozíciójukkal. Ezt a megjelenítést a segédablak, fogaskerekes szimbólummal jelölt, menüjének Adatforrás sorrendje szerinti rendezés beállítása biztosítja. A Rendezés A - Z utasítás ezt az elrendezést alakítja át ABC sorrendűre.

Az összes mező megjelenítését illetve elrejtését biztosító parancsot szintén a segédablak menüjében találjuk: Az összes kibontása/összecsukása.

Mikor a PowerPivot-kimutatás értékek területére olyan mezőt helyezünk, amely nem a kimutatás sor- vagy oszlopmezőinek tábláiból származik, akkor a bővítmény a mezőlistán jeleníti meg a kapcsolatok hiányára utaló információt. Talán még emlékszünk rá, hogy ilyen esetben elindíthatjuk a bővítmény automatikus-kapcsolat modulját.

A mezőket nem csak a táblanevek előtti álló, más alkalmazásokból már ismert, pici háromszöggel, hanem a táblanévre duplán kattintva is megjeleníthetjük/elrejthetjük.

z

79. ábra a mezőnév menüjének vezérlője piros nyíllal jelölve és az egér jobb billentyűvel megjeleníthető helyi menü

A mezőnevek menüje a rendezés és a szűrés eszközeit tartalmazza, míg helyi menüjük utasításaival a mező kimutatásban elfoglalt szerepét határozhatjuk meg. Az utolsó két parancs, *Hozzáadás szeletelőként* és *Hozzáadás idősorként*, külső szűrőt hoz létre a mező egyedi bejegyzéseivel. Utóbbi lehetőség természetesen csak dátum-mező esetén érhető el.

Külső szűrőt létrehozhatunk a program utasításával is: Kimutatáseszközök, Elemzés, Szűrés, Szeletelő beszúrása/Idősor beszúrása paranccsal is.

A mezőlista alján álló, *Elrendezésfrissítés elhalasztása* feliratú vezérlővel kikapcsolhatjuk a kimutatás automatikus frissítését. Ebben az üzemmódban a jelölőnégyzettel egy sorban álló *Frissítés* parancsgombbal kérhetjük a kimutatás aktualizálását.

kimutatás és kimutatásdiagram létrehozása

A bővítmény ablakának *Kezdőlap, Kimutatás* parancslistájának elemeivel tudunk kimutatást vagy kimutatásdiagramot létrehozni. Vegyünk sorra a lehetőségeinket.

[1] Kimutatás: PowerPivot-kimutatás létrehozása, a felhasználó által megadott helyen.

[2] *Kimutatásdiagram*: kimutatásdiagram létrehozása, a felhasználó által meghatározott helyen. A mezőlista alján álló objektum-területek funkciói és feliratai a következők. Csak szűrő-feladatot ellátó mezők területe: Szűrők. A függőleges, vagyis az értéktengely osztásait meghatározó mező területe: Jelmagyarázat (adatsor). A vízszintes, azaz a kategoria tengely osztásait meghatározó mező területe: Tengely (kategoriák). A statisztikai mező területe: Értékek.

[3, 4] Diagram és táblázat (vízszintesen), Diagram és táblázat (függőlegesen): létrehozza a nevében szereplő objektumokat, a nevében szereplő sorrendben és elrendezésben, a felhasználó által meghatározott helyen. A két objektum független egymástól. Az egyik objektum szelektálására létrehozott külső szűrő csatolható a másik objektumhoz. Szinkronizált működésű kimutatás-kimutatásdiagram párost két lépésben tudunk létrehozni: először elkészítjük a PowerPivot-kimutatást, majd a beszúrjuk a diagramot a Kimutatáseszközök, Elemzés, Eszközök, Kimutatásdiagram utasítással.

[5, 6] Két diagram (vízszintesen), Két diagram (függőlegesen): két kimutatásdiagram létrehozása a nevében szereplő elrendezésben, a felhasználó által meghatározott helyen. Az egyik objektum szűrője csatolható a másik objektumhoz.

[7] Négy diagram: négy kimutatásdiagram létrehozása, a felhasználó által meghatározott helyen. Az egyik objektum külső szűrője csatolható a többi objektumhoz.

[8] Egybesimított kimutatás: egyetlen PowerPivot-kimutatás létrehozása, meghatározott formai elemekkel, a felhasználó által meghatározott helyen.

Elemszám - név	Oszlopcímkék 👕						
Sorcímkék 👘	beosztott	felsővezető veze	ető Végi	összeg			
🗏 Baja	131	6	10	147			
pénzügy	26	2	2	30			
szállítás	59	2	4	65			
tervezés	46	2	4	52			
🗏 Ócsa	115	6	10	131			
pénzügy	23	2	2	27			
szállítás	43	Elemszám - név	<u></u>	státusz	. 💌		
tervezés	49	telephely 📑	osztály 🕋	beosztott	felsővezető	vezető	
🖃 Paks	130	Baja	pénzügy		26	2	2
pénzügy	26	Baja	szállítás		59	2	4
szállítás	47	Baja	tervezés		46	2	4
tervezés	57	Baja Összeg			131	6	10
🗆 Vác	105	Ócsa	pénzügy		23	2	2
pénzügy	21	Ócsa	szállítás		43	2	4
szállítás	45	Ócsa	tervezés		49	2	4
tervezés	39	Ócsa Összeg			115	6	10
Végösszeg	481	Paks	pénzügy		26	2	2
		Paks	szállítás		47	2	4
		Paks	tervezés		57	2	4
		Paks Összeg			130	6	10
		Vác	pénzügy		21	2	2
		Vác	szállítás		45	2	4
		Vác	tervezés		39	2	4
		Vác Összeg			105	6	10

80. ábra a "szokásos" és az "egybesimított" megjelenítésű kimutatás

Tehát a parancslista utolsó elemével egy speciális megjelenítésű kimutatást kapunk. Ha beszúrnánk egy kimutatást, akkor a következő beállításokkal alakíthatnánk át "egybesimított" kimutatássá.

Sorok és oszlopok összegzésének kikapcsolása: Kimutatáseszközök, Tervezés, Elrendezés, Végösszegek, Kikapcsolva a sorokban és oszlopokban. Táblázatos megjelenítés bekapcsolása: Kimutatáseszközök, Tervezés, Elrendezés, Kimutatás elrendezése, Megjelenítés táblázatos formában.

Külső sor- illetve oszlopmezői tételeinek kibontás-összecsukás kapcsolóinak elrejtése: Kimutatáseszközök, Elemzés, Megjelenítés, +/- gombok.

Külső sor- illetve oszlopmezők tételneveinek megjelenítése minden sorban: Mezőbeállítások, Elrendezés és nyomtatás, ☑ Tételcímkék ismétlése.

Minden második sor hátterének színezése: Kimutatáseszközök, Tervezés, Kimutatásstílusok beállítása, Sávos sorok. Az utolsó művelet a kimutatásstílus kiválasztása lenne, de az "Egybesimított kimutatás stílus", amellyel a bővítmény formázza az objektumot nem szerepel a választható formátumok között.

tételek egyedi sorrendje a kimutatásban

A sor- és oszlopmezők tételei a PowerPivot-kimutatásban is rendezetten, növekvő sorrendben jelennek meg, de az Excel egyéni listáit a bővítmény nem veszi figyelembe. Egyedi sorrendet a Power-Pivot "rendezés más mező alapján" nevű szolgáltatásával határozhatunk meg. A modul működésének lényege, hogy a sor illetve az oszlopmező tételeinek sorrendjét egy másik mező bejegyzéseinek sorrendjével határozzuk meg.



81. ábra rendezendő és rendező mezők

Az ábra két táblájának bal oldali mezőit tervezzük a kimutatás sor- vagy oszlopterületén szerepeltetni. A hét napjainak időrendi megjelenítését az "nap index" mező, a kerületek emelkedő sorszámú megjelenítését a "kerület index" mező bejegyzésein alapuló rendezés biztosítja majd.

A modul parancstábláját adatnézetben, a rendezendő mezőre kattintva, a *Kezdőlap, Rendezés* és szűrés, Rendezés más oszlop alapján utasítással jeleníthetjük meg. A rendezendő mezőt a bővítmény automatikusan kitölti. A rendező mezőt a panel jobb oldalán álló listából kell kiválasztanunk. A rendezendő mező azonos bejegyzéseit tartalmazó rekordokban a rendező mező bejegyzéseinek is azonosnak kell lennie!

PowerPivot bővítmény az Excel programhoz					
8	A(z) kerület nem rendezhető cím szerint, mert a(z) kerület mezőben levő legalább egy értékhez több különböző, cím mezőbeli érték tartozik. Például a [Város] mező rendezhető [Régió] szerint, mert mindegyik városhoz csak egy régió tartozik, de a [Régió] mező nem rendezhető [Város] szerint, mert mindegyik régióhoz több város tartozik.)			
Részletek >> OK					

82. ábra a fenti előírás megsértésekor megjelenő hibaüzenet

A "rendezés más mező alapján" deklaráció a táblában álló rekordok sorrendjét nem befolyásolja. Az egyedi sorrend szerint rendezendő mezők száma nincs korlátozva. A speciális megjelenítést, a mező kiválasztása után, a *Kezdőlap, Rendezés és szűrés, Rendezés más* oszlop alapján, Más oszlop szerinti rendezés kikapcsolása utasítással törölhetjük.

egyéni nézetek

fogalmak

Az egyéni nézet a felhasználó által meghatározott táblák meghatározott mezőinek, elnevezett csoportja. A program az egyéni nézetet a munkafüzetbe menti. Az egyéni nézet csak a bővítmény ablakában érvényesül, beolvasását követően a PowerPivot ablakban csak az egyéni nézet tábláit és mezőit látjuk, de a kimutatás mezőlistáján továbbra is a teljes adatbázis a rendelkezésünkre áll. Természetesen az "elrejtés az ügyféleszközök elől" művelettel eltiltott objektumok nélkül. A bővítmény az egyéni nézetet "perspektívának" nevezi. A teljes adatbázist meg "alapértelmezett" perspektívának.

egyéni nézetek kezelése

Az egyéni nézetek parancstábláját a Speciális, Perspektívák, Létrehozás és kezelés paranccsal jeleníthetjük meg. A Perspektívák panel bal oldalán, a Mezők felirat alatt a bővítmény hierarchikusan ábrázolja az adatbázis objektumait. A hierarchia csúcsán a teljes adatbázist jelentő, Táblázatok felirat áll. A második logikai szintet a táblanevek foglalják el, végül a harmadik szinten a mezőnevek állnak. A tartalmazott objektumok megjelenítésének szabályozására az első két logikai szint elemei előtt, plusz-mínusz jeles, pici vezérlők állnak, a Microsoft új, szuper minimalista dizájnjával összhangban, keret nélkül.

A Mezők felirat feletti Új perspektíva parancsgombbal kezdeményezhetjük egy új egyéni nézet létrehozását. A bővítmény az utasítás hatására egy jelölőnégyzetes oszlopot jelenít meg, "Új perspektíva: <sorszám>" automatikus névvel. A név természetesen lecserélhető.

rspektívák				?	
łasználjon perspektívákat az ragy üzleti helyzethez definiál Új perspektív <u>a</u>	adatok nézete ják, és a segí	inek definiálásál tségükkel egysze	noz. A perspektív erűsíthető a nagy	ákat általában egy adott felhasználói csoportho adathalmazokban való navigáció.	Z
Mezők	számlák	használók	típusok		
- <u>T</u> áblázatok					
+ autóhasználat					
+ autók					
+ javítások					
- munkatársak					
anyja neve					
belépett					
kilépett					
név		\checkmark			
osztály		\checkmark			
személyi szám		\checkmark			
születési dátum					
Nincsenek objektumok a kövel használatával kapcsolódik ehh	tkező perspekt ez a perspekt í	ívában: "típusok vához, a mezőlista	". Amikor egy felha a üres lesz.	asználó az Excel vagy egy másik ügyféleszköz	
				OK Méas	e

83. ábra az egyéni nézetek objektumlistái

A felhasználó a jelölőnégyzetek segítségével állítja össze az objektum-listát. Az adatbázis összes objektumát a Táblázatok sorban álló, az adott tábla összes objektumát a tábla nevének sorában álló négyzettel választhatjuk ki. Az első két logikai szinten álló objektumok jelölőnégyzetében álló négyzet a részleges, tehát a szelektív kiválasztást jelöli.

A bővítmény esetleges üzeneteit a listák alatt, egy erre a célra megjelenített színes téglalapban olvashatjuk. Az egyéni nézetek kezelése az adott lista nevére mutatva megjelenő eszköztár parancsaival történik. A parancsgombok funkciója balról jobbra haladva: [1] a nézet törlése, [2] a nézet átnevezése és [3] másolat készítés a nézetről egy új lista létrehozáshoz. A mentett a nézetek öszszetétele szabadon módosítható. A panel *OK* gombja menti az új listát és a módosításokat.

A nézetek listáját a Speciális lap Perspektívák csoportja tartalmazza. A lista <Alapértelmezett> bejegyzése a teljes adatbázist jeleníti meg.



84. ábra az egyéni nézetek listája az eszköztáron

Összefoglalva, az egyéni nézet és az "elrejtés az ügyféleszközök elől" műveletek funkciója azonos, a megjelenítendő objektumok meghatározása, de míg az előbbi csak a bővítmény ablakában érvényesül, addig az utóbbi a kimutatás mezőkészletét is szűkíti.

több a többhöz kapcsolat

fogalmak

Két tábla több a többhöz viszonyáról akkor beszélhetünk, ha a táblák tetszőleges rekordjához a másik tábla több rekordja tartozhat. Ez a logikai kapcsolat a mindennapi életben sem ritka. Gondoljunk csak a színészek és a színdarabok vagy a fordítók és a nyelvek viszonyára! Egy színházban egy színész több színdarabban is játszhat és egy színdarab előadása általában több színészt igényel. Vagy vegyünk egy fordító irodát, egy fordító akár több nyelvet is beszélhet és egy nyelvet több fordító is ismerhet. A több a többhöz viszonyban álló táblák csak egy közbenső tábla segítségével kapcsolhatók össze. Ez a harmadik tábla a kapcsoló tábla.



85. ábra két példa a több a többhöz kapcsolatra

Az ábrán láthatjuk, hogy a kapcsoló tábla nem csak a kapcsoló mezőket, de a logikai viszony egy vagy több tulajdonságát is tárolhatja. Például a felső tábla-csoportban a szerep nevét: egy kiválasztott színész az egyik színdarabban az "apa", a másikban a "szerető". Vagy a nyelvtudás táblában a szint mező: egy fordító az egyik nyelvet "anyanyelvi" szinten beszéli, a másikat inkább csak "érti". A fenti képen azt is megfigyelhetjük, hogy a több a többhöz kapcsolatban a kapcsoló tábla mindkét kapcsolatban a kapcsolat több oldalán áll.

számított mező több a többhöz kapcsolatban

A fogalmak tisztázásánál használt fordítók példánál maradva, ki szeretnénk számolni az egyes fordítók havi jövedelmét, amely a "fizetés" és a felsőfokon beszélt nyelvek után járó "pótlék" összege. A pótlék mértéke nyelvenként különböző. A számított mező a "fordítók" táblában álljon.



86. ábra a példa két számított mezője: a JÖVEDELEM és a PÓTLÉK

A legegyszerűbb megoldást választjuk: a nyelvtudás táblában létrehozunk egy számított mezőt, amely tartalmazza az adott nyelvtudás után járó esetleges összeget, majd a fordítók tábla számított mezője összeadja az adott fordítóhoz tartozó összegeket.

A nyelvtudás tábla PÓTLÉK mezőjének képlete: =IF([szint] = "felsőfok"; RELATED(nyelvek [pótlék])). A fordító csak akkor jogosult a nyelvpótlékra, ha az adott nyelvet felső szinten beszéli. Ezért az IF függvényt alkalmazzuk, amely csak annyiban tér el a program HA függvényétől, hogy ha a harmadik argumentumát nem adjuk meg, akkor az első argumentum HAMIS értékénél a függvény üres eredményt ad. A nyelvek tábla neve a képletben nincs aposztrófokkal keretezve, mert a név nem tartalmaz szóközt és ékezetes karaktert. Utóbbi kitétel független az operációs rendszer területi beállításaitól.

A fordítók tábla JÖVEDELEM mezőjének képlete: = [fizetés] + CALCULATE(SUM ('nyelvtudás' [PÓTLÉK]) ; 'nyelvtudás'[fordító] = EARLIER([fordító])). A CALCULATE függvény a DAX specialitása, az első argumentumával deklarált kifejezést értékeli ki, azokon a rekordokon, amelyeket a további argumentumaival megadott szűrő-feltételekkel kell kiválasztania. A képletben alkalmazott CALCULATE függvény összeadja a nyelvtudás tábla, azon rekordjainak pótlékát, amelyeknek fordítója azonos a vizsgált fordítóval.



87. ábra a PÓTLÉK és a JÖVEDELEM számított mezők képlete

összesítések több a többhöz kapcsolatban

Hogy megértsük a több a többhöz kapcsolat kezelését, először nézzünk egy feladatot, amelynek táblái egy a többhöz viszonyban állnak egymással.



88. ábra egy a többhöz viszonyban álló táblák

Hány gyerekük van az egyes férfiaknak? A megoldás egyszerű: a mezőlistán a sorterületen elhelyezzük a férfiak tábla férfi mezőjét, majd a gyerekek tábla gyerek mezőjét az értékek területre húzzuk. A létrehozott Elemszám - gyerek automatikus összesítés értékei adnak választ a kérdésünkre. A lista végén álló tizenkét gyerek apukája nem szerepel a férfiak táblában.



89. ábra a feladat megoldása automatikus összesítéssel

Tehát a bővítmény képes leválogatni az egy a többhöz viszonyban álló táblák összetartozó rekordjait, még akkor is, ha a táblák nincsenek közvetlen kapcsolatban.

Most térjünk vissza a fordítók adatbázis több a többhöz logikai viszonyban álló tábláihoz! A fordító jövedelme a fizetés és a felsőfokú nyelvtudásaiért kapott pótlékok összege. Kimutatással összegezzűk az egyes fordítók pótlékainak összegét.

Kipróbáljuk az előbb bevált módszert: a fordítók tábla fordító mezőjét a sorterületre helyezzük, majd a nyelv tábla pótlék mezőjét az értékek területére húzzuk... De most ez nem jön be. A mezőlista tetején megjelenik a kapcsolatok hiányára utaló figyelmeztetés és minden fordító mellett a nyelvek tábla összes pótlék bejegyzésének összege áll.

Szükség lehet a táblák köz	ötti kapcsolat	okra.	×				
AUTOMATIKUS FELISMERÉS LÉTREHOZÁS							
Keresés			0				
Nereses	fordító	-	Összeg -	pótlék			
	Béres Va	azul	105 000 HUF				
	Hajós Er	105 000	HUF				
✓ fordito	Hamar E	105 000	HUF				
fizetés	Jenei Ág	105 000 HUF					
	Nemes I	Magda	105 000	HUF			
	Nyéki Zs	olt	105 000	HUF			
(The such take	Ormai Já	inos	105 000	HUF			
	Pajor Re	Pajor Rezső		HUF			
	Sényi An	dor	105 000	HUF			
✓ pótlék	Varga Ki	nga	105 000	HUF			
_	Végössze	g	105 000	HUF			
Image: Provide transmission of the second		_					

90. ábra az automatikus összesítés eredménye

Tehát több a többhöz kapcsolatban álló táblákban a bővítmény, segítség nélkül, nem tudja meghatározni az összetartozó rekordokat. A megoldás természetesen az egyéni összesítés: "összpótlék". Képlete: =CALCULATE(SUM ('nyelvek'[pótlék]); FILTER('nyelvtudás'; [szint] = "felsőfok")).

🔺 🛅 fordítók	fordító	-	összpótlék
✓ fordító	Hajós Emőke	15 000 HUF	
fizetés	Jenei Ágnes		10 000 HUF
f ===============================	Nemes Magda	45 000 HUF	
$\checkmark f_X$ osszpotiek	Zpotlek Pajor Rezső		30 000 HUF
	Sényi Andor		35 000 HUF
E E nyelvek	Varga Kinga	20 000 HUF	
N III nuch tudác	Végösszeg	80 000 HUF	
v 🔤 nyelvtudas			
	<u> </u>		

91. ábra a feladat megoldása egyéni összesítéssel

Az egyéni összesítés képletében a CALCULATE függvény második argumentuma a FILTER függvényel képzett, a felsőfokú bejegyzésekre szűrt nyelvtudás tábla, amelynek kapcsolódó rekordjait már meg tudja keresni a bővítmény a nyelvek táblában.

A fentiekből következik, hogy a CALCULATE függvény második és további argumentumai táblák is lehetnek. Ezt a tapasztalatot felhasználhatjuk, ha a több a többhöz kapcsolatban álló táblák öszszes összetartozó rekordjára szükségünk van az összesítéshez. Páldául meg szeretnénk állapítani, hány színdaraban játszanak az egyes színészek.



a "színészek" és "színdarabok", több a többhöz logikai viszonyban álló táblái

Ha az előbb megismert sémát kívánjuk alkalmazni, akkor a "hány darabban" egyéni összesítés így alakul: =CALCULATE(COUNTA('színdarabok'[cím]); FILTER(szerepek ; [cím] = RELATED('színdarabok'[cím]))). Magyarul a FILTER függvénnyel a "szerepek" és a "színdarabok" táblák kapcsolatát deklaráltuk, amelyet az összetartozó rekordok keresésénél a CALCULATE függvénynek használnia kell. Ez a kvázi kapcsolat-meghatározás egyszerűbb szintaktikával is történhet: CALCULATE(COUNTA('színdarabok'[cím]); szerepek). Tehát elegendő csak a kapcsolótábla nevét megadnunk.

Ha a kapcsolótábla kapcsoló mezői ezt megengedik, más módszert is alkalmazhatunk több a többhöz kapcsolatban álló táblák elemzéséhez. Például: állapítsuk meg a fenti táblacsoportban, [1] hány szerepet formálnak meg és [2] hány színdarabban játszanak az egyes színészek! Tételezzük fel, hogy egy színész egy színdarabban több kisebb szerepet is játszhat.

Az első kérdés megválaszolása egyszerű: helyezzük a színészek tábla név mezőjét a mezőlista sorterületére, majd húzzuk a szerepek tábla szerep mezőjét az értékek területre. Az automatikus összesítés helyes értékeket ad. A második kérdést is megválaszolhatjuk automatikus összesítéssel: a szerepek tábla tartalmazza a színdarabok címét, de ha egy színész egy színdarabban több kisebb szerepet is játszik, akkor a színdarab címe ismétlődni fog. Az egyedi bejegyzéseket a DISTINCTCOUNT függvénnyel számoltathatjuk meg. Tehát a második kérdést is megválaszolhatjuk automatikus ösz-szesítéssel. Ha akarjuk, a megoldásból még az első táblát is kihagyhatjuk: a fordító nevét is a kap-csolótáblából vesszük.

		\sim	
	=	SOROK	Σ ÉRTÉKEK
	n	év 🔻	Elemszám - szerep 🔻
			Eltérők darabszáma – cím 🔻
név 🗠	Elemszám - szere	p Eltérők darabszáma – cím	
Csernus Máté	3	3	
Nógrádi Sára	5	3	EDIGO(TÉG
Országh Huba	2	2	FRISSITES
Reményi Ottó	5	3	
Solymári Ernő	3	3	
Szamosi Dóra	2	2	
Szendrő Vera 2		2	
Végösszeg	22	4	

93. ábra a feladat megoldása két automatikus összesítéssel

feltételes formázás a kimutatásban

fogalmak

A bővítmény KPI (Key Performance Indicator) szolgáltatása egy ikonkészletes feltételes formázás, amely egyéni összesítések célértékes vagy középértékes vizuális elemzését teszi lehetővé.

A szolgáltatás elnevezése, a magyar verzióban "fő teljesítmény mutató", többszörösen félrevezető. [1] A név azt sugallja, hogy egy független objektumról van szó, ami nem igaz. A rajzi objektum az egyéni összesítés grafikus megjelenítése. [2] A szolgáltatás az egyéni összesítés tárgyára vonatkozóan semmilyen korlátozást nem ír elő, tehát a "teljesítmény" szónak a névben nincs értelme, mert a rajzi megjelenítés kapcsolódhat "nyereséghez", "kihasználtsághoz", bármihez... [3] Egy kimutatásban több egyéni összesítéshez kérhetjük a szolgáltatást és a megjelenített grafikus rendszerek között semmilyen alárendeltségi viszony nincs, így a "fő" szónak sincs semmi értelme. Tehát a mutató nem fő- és nem teljesítmény! Akkor mi? Ikonkészletes feltételes formázás.

A grafikus megjelenítés az egyéni összesítés értéktartományának szakaszokra bontásán és a kiválasztott ikonkészlet elemeinek a szakaszokhoz rendelésén alapszik. A szolgáltatás legfontosabb tulajdonságait a felhasználó határozza meg: [1] ikonkészletet választ, [2] megadja az elemzés típusát, célértékes vagy középértékes vizsgálatot szeretne, [3] megadja a szakaszhatárokat, [4] a célértéket illetve a középértéket, valamint [5] meghatározza az ábrácskák sorrendjét.

A feltételes formázás beállítását követően önállóan elrejthetjük, illetve megjeleníthetjük az egyéni összesítés [1] kiszámolt értékeit, [2] a feltételes formázás ábrácskáit, [3] a feltételes formázás cél- illetve középértékét. A felhasználó által megadott cél- vagy középérték nemcsak konstans lehet, hanem egy másik egyéni összesítés is.

a feltételes formázás kezelése

A bővítmény ablakában, adatnézetben, az egyéni összesítést kijelölve, a Kezdőlap, Számítások, KPI létrehozása, vagy az egyéni összesítés helyi menüjében a KPI létrehozása... paranccsal állíthatunk be feltételes formázást. Az egyéni összesítés helyi menüje tartalmazza a feltételes formázás módosításának és törlésének utasításait is. A program-ablak PowerPivot, Számítások, KPI-k listája tartalmazza a feltételes formázás parancsait: Új KPI... és a KPI-k kezelése... parancsokat.

célértékes elemzés létrehozása

Az ikonkészletes feltételes formázással általában a vizsgált adathalmaz elemeinek nagyságát kívánjuk szemléltetni. A határértékek és a kategoriákhoz rendelt ábrácskák ismeretében, már egyetlen futó pillantással megállapíthatjuk egy elem hozzávetőleges nagyságát. Minél jobban megközelítik az elemek, az általunk ideálisnak vélt értéket, amit nevezzünk célértéknek, annál elégedettebbek vagyunk. Akkor vagyunk a legboldogabbak, ha némely elem még nagyobb is ennél a határértéknél. Gondoljunk csak árbevételeink nyereség tartalmára.

De van úgy, hogy akkor vagyunk elégedettek, ha minél kisebbek egy adathalmaz elemei. Ilyen lehet például az árbevételeink költség része. Ebben az esetben a határértékek a célértéknél nagyobbak lesznek és az ikonok sorrendje is változhat. A célértékes vizuális elemzés, az ideálisnak vélt érték értelmezésének módja szerint tehát lehet [1] alulról közelítő, "minél nagyobb annál jobb", vagy [2] felülről közelítő, "minél kisebb annál jobb" típusú. Először ismerkedjünk meg az alulról közelítő vizuális elemzés tulajdonságaival egy kéttáblás adatbázis segítségével: diákok - vizsgák. A feladat szempontjából lényeges körülmények: [1] egy diák több tantárgyat hallgat, [2] egy tantárgyból több vizsgát kell abszolválnia, [3] megfelelt a tantárgyi követelménynek az a hallgató, akinek a vizsgákon elért pontszámainak átlaga elérte az abból a tantárgyból elnyerhető maximális pontszámok átlagának hatvan százalékát, [4] kitűnően megfelelt az a hallgató, aki kilencven százalékot ért el. Értékeljük a hallgatók féléves munkáját, tantárgyanként, a megszerzett pontszámaik alapján.



94. ábra a példa két táblája

Az egy oldali tábla, vizsgák, tartalmazza az összes tantárgy összes vizsgáját és az egyes vizsgákon elérhető maximális pontszámot. A két táblát a vizsga azonosítóját tartalmazó mező kapcsolja öszsze. Az adott vizsgán elért pontszámot a diákok táblában találjuk. Az értékelést kimutatás segítségével fogjuk elvégezni. Egyéni összesítéssel megállapítjuk az egyes hallgatók, az adott tantárgy vizsgáin megszerzett pontszámainak átlagát, majd egy három-elemű ikonkészletes-formázással megjelenítjük, hány százaléka ez az érték, a maximális pontszámok átlagának, amelyet szintén egyéni összesítéssel számolunk ki.

A kimutatás sormezői a 'diákok'[név] és a 'vizsgák'[tantárgy] lesznek. A statisztikai mező, a "teljesítmény" nevű egyéni összesítés, amely a hallgatók átlagos pontszámát számolja ki. Képlete: =ROUND(AVERAGE('diákok'[pontszám]); o). Ezután hozzuk létre feltételes formázás célértékét szolgáltató egyéni összesítést! Neve legyen "maximális". Képlete =CALCULATE(ROUND(AVERAGE('vizsgák'[maximális pontszám]); o); 'diákok'). A második összesítés létrehozása után adjuk meg a feltételes formázás tulajdonságait.

A Fő teljesítmény mutató feliratú parancstáblán először a KPI alapmezője (érték) feliratú listából ki kell választanunk a formázni kívánt egyéni összesítést. Ezt követően a célértéket kell deklarálnunk: [1] ha ez konstans, akkor beírjuk a Rögzített érték mezőbe, [2] ha egy másik egyéni összesítés, akkor a Mérték legördülő listából kell kiválasztanunk a nevét. A mi esetünkben ez a "maximális" nevű egyéni összesítés! A szakaszhatárok megadása előtt ikonkészletet kell választanunk a parancstábla aljáról. A vizuális elemzéshez a bővítmény négy darab három elemű és egy darab öt elemű ikonkészletet kínál. Válasszuk az "x, !, ✓" készletet!

Az egyéni összesítés értéktartományának szakaszhatárait az Állapotküszöbök definiálása felirat alatt álló színes sávon kell megadnunk. A modul három elemű ikonkészlethez két darab, öt elemű ikonkészlethez négy darab szakaszhatárolót jelenít meg. A kis, tölcsérre emlékeztető ábrácskát lenyomott egérbillentyűvel jobbra-balra huzigálhatjuk. Állítsuk be az első csúszkát hatvan, a másodikat kilencven százalékra!



95. ábra alulról közelítő célértékes vizuális elemzés tulajdonságai

Ha az elemzés célértéke egyéni összesítés, akkor a határértékeket az egyéni összesítés eredményének százalékában adhatjuk meg. Konstans célérték esetén a határértékek is konstansok lesznek. A sávon megjelenített Cél feliratú szaggatott vonal a célértéket jelöli. A mi példánkban ez száz százalékot jelent.

célérték	3 elemű ikonkészlet	5 elemű ikonkészlet		
egyéni összesítés	40%, 80%	20%, 40%, 60%, 80%		
konstans	a célérték 40%, 80%-a	a célérték 20%, 40%, 60%, 80%-a		

96. ábra a célértékes elemzéshez felajánlott határértékek

A feltételes formázás tulajdonságainak megadását követően az egyéni összesítés vezérlője átalakul a mezőlistán. Az fx jelzés helyett egy közlekedési lámpát, az egy jelölőnégyzet helyett hármat kapunk, amellyel külön-külön szabályozhatjuk a statisztikai értékeknek (Érték <egyéni összesítés>), a feltételes formázás cél- vagy középértékének (Cél) illetve a feltételes formázásnak (Állapot) a megjelenítését.

	név		I ak	ok név			
	vizsga kód						
	pontszám			nontszám			
	f _y teliesítmény			fr maximális			
	f., na svina ális			n in in			
ш.	JX maximalis		4 € 1	eijesitmeny	oliocítmónu		
🗉 vizs	aák				eijesititietty/		
V 1	tantárgy		L	_ Cél			
	vizsaa kód		[Állapot			
_		-					
1	név 💽	vizsga kó	id 👘 🖬	pontszám			
1	Rideg Emilia	E1-2			95		
2	Bacsó Zoltán	E2-2			280		
3	Lakos Mónika	MI2-2			97		
4	Kövér Aladár	H1-1			138		
- E	teljesítmény: 180 🔳						
	maximum: 266						

97. ábra a feltételesen formázott egyéni összesítés megjelenítése a mezőlistán és adatnézetben a számítási területen

A feltételes formázás létrehozás után az ábrácskák helyett számok (-1, 0, 1) jelennek meg a kimutatásban. Ez program-hiba. Ha az Állapot jelölőnégyzet pipáját töröljük, majd újra megadjuk, akkor láthatóvá válnak az ikonok.

Sorcímkék	 teljesítmény 	Sorci	ímkék	-	teljesítmény	teljesítmény állapot
🗏 Aradi Róbert		⊟ Ar	radi Róbert			
művészettörténet I.	264		művészettörténe	et I.	264	1
művészettörténet II	. 256		művészettörténe	et II.	256	1
művészettörténet II	I. 211		művészettörténe	et III.	211	0
művészettörténet IV	1. 127		művészettörténe	et IV.	127	-1
világirodalom II.	216		világirodalom II.		216	0
🗏 Arató Csanád	Sorcímkék		* teljesítmény	teljesi	ítmény Állapot	
esztétika III.	🗆 Aradi Róbert					-1
humánetológia II.	művészettö	rténet l	. 264	0		1
statisztika III.	művészettö	rténet l	I. 256	õ		0
statisztika IV.	művészettö	rténet l	II. 211	õ		0
	művészettö	rténet l	V. 127	×		
	világirodalo	om II.	216	Ō		
	🗏 Arató Csanád					
	esztétika II	I.	134	8		1
	humánetol	ógia II.	257	0		
	statisztika	III.	228	•		
	statisztika	IV.	126	•		
					~]

^{98.} ábra a kimutatás a feltételes formázás beállítása előtt, a beállítás után, majd a hiba kijavítását követően

Konstans célérték megadása szükségessé teheti a színes sávval szimbolizált értéktartomány tágítását illetve szűkítését. Erre a célra szolgál a sáv két szélén álló kétirányú nyíl. Ha lenyomott egérbillentyűvel a kis ábrácskát "kihúzzuk" a panelről, akkor tágítjuk, ha "behúzzuk" a panel közepe felé, akkor szűkítjük a szimbolizált értéktartományt.



99. ábra az intervallum tágítása (a kettős nyilat kifelé húzni) és szűkítése (... befelé húzni)

A felülről közelítő célértékes vizuális elemzés létrehozásához át kell húznunk a kis tölcséreket a célérték jobb oldalára és ha szükséges a négy színes sáv bal alsó elemével meg kell fordítanunk az ikonok sorrendjét.

középértékes elemzés létrehozása

Vannak olyan adathalmazok, amelynek elemeitől azt várjuk, hogy minél jobban közelítsék meg az általunk ideálisnak vélt értéket, a középértéket. Másként fogalmazva az egyes elemek ne legyenek sokkal nagyobbak, de ne legyenek sokkal kisebbek sem a középértéknél. Akkor vagyunk elégedettek, ha az értékek egy szűk sávon belül szóródnak. Példaként gondoljunk autóbusz járatok követési idejére vagy egy valutára, amelyet az ország nemzeti bankja tranzakciókkal próbál egy árfolyam-sávban tartani. Ezeket a halmazokat középértékes elemzéssel szemléltethetjük, amely egy alulról közelítő és egy felülről közelítő célértékes ábrázolás egyesítése. Beállítása az intervallum-sáv alatti négy színes téglalap jobb felső elemével történik. A jobb alsó vezérlővel az ikonok sorrendjét fordíthatjuk meg. A megjelenítés tulajdonságainak megadási módja azonos a célértékes elemzésnél megismerttel.



100. ábra középértékes vizuális elemzés szakaszhatárai

A Fő teljesítmény mutató KPI panel egy összetett parancstábla, két lappal. A lapok közötti váltás vezérlője a felfelé mutató kettős nyíl a panel bal alsó sarkában. A parancstábla második lapján megjegyzéseket fűzhetünk a feltételes formázás objektumaihoz.

Fő teljesítménymutató (KPI)	?	\times
KPI alapmezője (érték): átlag	~	
KPI állapot <u>a</u>		
Határozza meg a célértéket:		
◯ <u>M</u> érték:	~	
<u>R</u> ögzített érték: 1300		
	\geq	4
<u>I</u> konstílus kiválasztása:		
)	
)	
* Leírások		
ОК	Mégs	e:

101. ábra az összetett parancstábla második lapját megjelenítő vezérlő
függvények

a DAX-függvény jellemzői

A DAX-függvények néhány tulajdonsága eltér a programban megszokottól. Ezek a következők.

[1] Angol nevek. A függvények nevét nem fordították le magyar nyelvre.

[2] Nincsenek "kötetlen argumentum-számú" függvények. Természetesen az Excelben sincsenek ilyenek, de azért a kétszázötvenhatos argumentum-határt nehéz megközelíteni a SZUM, az ÁT-LAG vagy mondjuk a SZORZAT függvénnyel. A DAX hasonló függvényei mindig csak egyetlen mezőt vizsgálnak.

[3] Nincs tartomány-hivatkozást. Másként fogalmazva, nincs rekord-csoport hivatkozást. Ez a sajátosság a mezők tartományára is vonatkozik. Ha tartomány-hivatkozással nem is, de függvényekkel szűkíthetjük egy DAX kifejezés hatókörét.

[4] Szűrőkezelés a képletben. Függvények segítségével meghatározhatjuk a képlet kiértékelésekor számításba veendő szűrők körét.

[5] Táblát eredményező függvények. A DAX függvények egy része virtuális táblát eredményez, egy vagy több nevesített mezővel, egy vagy több rekorddal. A táblát eredményező függvények csak egy másik függvény argumentumaként szerepelhetnek a képletben.

[6] Korlátlan egymásba ágyazási lehetőség. A logikai szinteknek számának csak a számítási kapacitás szab határt.

dátumok kezelése

A bővítmény TODAY függvénye, azonos a program MA függvényével: az operációs rendszer által szolgáltatott, aktuális dátumot adja eredményül, nulla óra nulla perc nulla másodperc időponttal kiegészítve.

Egy dátum éveinek, hónapjainak és napjainak számát a YEAR, a MONTH és a DAY függvényekkel határozhatjuk meg. A függvények egyetlen argumentuma a dátumot eredményező kifejezés, vagy a dátumot tartalmazó mező, beleértve a számított mezőket is, amelyekkel dátumot hoztunk létre. A három függvény működése azonos a program ÉV, HÓNAP és NAP nevű függvényeivel.

Egy dátum és az őt tartalmazó hét tulajdonságait a WEEKDAY és WEEKNUM függvényekkel határozhatjuk meg. A két függvény működése hasonló a program HÉT.NAPJA és HÉT.SZÁMA függvényeivel. A WEEKDAY függvénnyel megállapíthatjuk, hogy a dátum a hét hányadik napjára esik. A dátumot a függvény első-, a sorszámozás módját a függvény második argumentumával kell deklarálnunk. Ha a második argumentum kettő (2), akkor a hétfő az egyes (1), a vasárnap a hetes (7) sorszámot kapja. A WEEKNUM függvénnyel megállapíthatjuk, hogy a vizsgált dátumot az év hányadik hete tartalmazza. A függvény első argumentumával a dátumot, második argumentumával a hét kezdőnapját határozhatjuk meg. Ha a második argumentum kettő (2) akkor a hét első napja a hétfő. A függvény nem az EU-s szabvány szerint működik, mert az év első hetének a január elsejét tartalmazó hetet tekinti és nem az év első csütörtökét tartalmazót.

A DATE, az EDATE és az EOMONTH függvényekkel, a program DÁTUM, KALK.DÁTUM és HÓ-NAP.UTOLSÓ.NAP függvényeihez hasonlóan, naptár-helyes dátumot képezhetünk. A DATE függvény argumentumaival egy dátum éveinek számát, hónapjainak számát és napjainak számát deklarálhatjuk. Az argumentumok lehetnek szám-konstansok, számot tartalmazó mezők vagy számot eredményező kifejezések. A dátum összeállítása előtt a függvény a tizedes tört argumentumait egészre kerekíti.

Az EDATE függvény az első argumentumával meghatározott dátum, második argumentumával deklarált számú hónappal időben eltolt megfelelőjét adja eredményül. A függvény második argumentuma tehát egy előjeles szám-konstans, előjeles számot tartalmazó mező vagy egy előjeles számot eredményező kifejezés. A dátum képzése előtt a függvény a tört számot egészre kerekíti. A

második argumentum negatív előjele az első argumentum dátumánál korábbi, pozitív előjele, későbbi időpontot eredményez. Az EOMONT függvény működése és szintaktikája azonos az EDATE függvénnyel, csak az eltolással képzett dátum hónapjának utolsó naptári napját adja eredményül.

A YEARFRAC függvény egy időszak hosszát számolja ki években. Eredménye tizedes tört. Első két argumentuma, tetszőleges sorrendben, az időszak első és utolsó napját definiálja, harmadik argumentumával a számítás módját írhatjuk elő: a lehetséges öt metódust egész számok szimbolizálják, nullától négyig. A tényleges értéket, azaz az ABS([dátum₁] - [dátum₂]) / 365,25 képlettel nyert törtszámot, az egyes számú (1) változat közelíti meg legjobban.

A három argumentumos DATEDIFF függvénnyel két dátum közötti dátumegységek számát határozhatjuk meg. Első argumentumával a vizsgált időszak első napját, második argumentumával az utolsó napját, harmadik argumentumával a megszámlálandó dátumegységet deklaráljuk. A dátumegység nevét idézőjelek nélkül kell megadnunk: year, quarter, month, week, day, hour, minute, second. A felsorolás utolsó három eleme időegység, tehát az első és a második argumentummal időpontokat is meghatározhatunk: kezdés, befejezés sorrendben.

A DATEVALUE függvénnyel dátumot hozhatunk létre karakterláncból. A függvény egyetlen argumentuma az átalakítandó karakterlánc. Például: =DATEVALUE ([év] & ".01.01.")

TODAY	aktuális dátum képzése
YEAR, MONTH, DAY	dátumegységek számának megállapítása
WEEKDAY, WEEKNUM	nap sorszáma a héten, hét sorszáma az évben
DATE, EDATE, EOMONTH	dátum képzése
YEARFRAC	két dátum különbsége években, tört számként
DATEDIFF	két dátum közötti dátumegységek megszámlálása
DATEVALUE	szöveges dátum konvertálása dátummá

102. ábra dátum-kezelő függvények funkció szerint csoportosítva

A DATEVALUE függvénnyel kapcsolatban tudnunk kell: a szöveg adattípusú dátumokat a DAX automatikusan dátummá konvertálja, ha azt a feldolgozás megköveteli.

tételek csoportosítása a kimutatásban

A kimutatásos adatelemzés az adathalmazok csoportosított statisztikai vizsgálatát teszi lehetővé. A csoportosítás a sor és az oszlopmezők tételei alapján történik. A csoportosító mezők tételei tovább kategorizálhatók a program eszközeivel, de ezek a szolgáltatások, karakteres és numerikus mezők esetén, a PowerPivot kimutatásban elérhetetlenek. Az utasítások a szalagon aktívak, de a parancsok hibaüzenetet generálnak: A kijelölés nem fogható csoportba (sic!).

A tételek csoportosíthatóságának feltételeit a felhasználónak kell megteremtenie. A gyakorlatban ez azt jelenti, hogy létre kell hoznia a táblában egy számított mezőt, amely tartalmazza a csoportosítást biztosító azonosítókat.

karakteres tételek csoportosítása

Szöveg adattípusú mezők tételeinek csoportosításához létre kell hoznunk a csoportneveket tartalmazó számított mezőt. Képletében minden tételnek deklarálnunk kell a csoportját.

Lássunk egy példát a szöveg-csoportosító képletre! Egy vállat személyautóinak javításait kívánjuk PowerPivot-kimutatással elemezni. Kíváncsiak vagyunk a javítások számára, gyártók szerinti bontásban. Ehhez a "javítások" tábla "típus" mezőjének tételeit kell csoportosítanunk. A személyautók három konszerntől származnak. [1] Renault: Dacia és Renault típusok. [2] PSA: Citroen és a Peugeot modellek. [3] VW: Audi, Skoda, Seat, Volkswagen autógyárak termékei. Létre kell hoznunk egy számított mezőt, amelyben megadjuk minden típus gyártóját. A feladatot két IF függvényekkel oldjuk meg: =IF(SEARCH ("dacia"; [típus];; o) > o || SEARCH("renault"; [típus];; o); "Renault"; IF(SEARCH("citroen"; [típus];; o) > o || SEARCH("peugeot" ; [típus];; o) > o; "PSA"; "VW")). A számított mező neve legyen GYÁRTÓ!

A SEARCH függvénnyel egy, az első argumentumával meghatározott, karakterláncot keresünk a függvény második argumentumával meghatározott szövegben. A keresett karakterlánc helyettesítő karaktereket is tartalmazhat. A függvény eredménye egy sorszám: hányadik karaktertől kezdődik a szövegben a keresett karakterlánc. A függvény további argumentumai nem kötelezőek. A függvény harmadik argumentuma egy sorszám, a szöveg hányadik karakterétől kezdődjön a keresés, ha nem a teljes szöveget szeretnénk átvizsgálni. A negyedik argumentum szintén szám; az eredménytelen kereséshez rendelt érték. A mi képletünkben ez nulla. Ha a negyedik argumentum hiányzik, akkor a sikertelen keresés üres mezőt eredményez. A SEARCH függvény a kis és nagybetűk között nem tesz különbséget.

A képlet felépítése a következő. Minden gyártót egy IF függvényben deklarálunk, kivétel a legbelső IF, amelyben kettőt. A függvény feltétel argumentumában VAGY logikai kapcsolókat használtunk. Az autógyár nevét a SEARCH függvénnyel keressük a "típus" mezőben. Ha a feltétel teljesül, azaz a SEARCH függvény nullánál nagyobb számot ad eredményül, akkor a típus neve tartalmazza az autógyár nevét és a GYÁRTÓ mező bejegyzése az IF függvény második argumentumában álló tröszt-név lesz. Ellenkező esetben a függvény harmadik argumentumában álló IF függvény hajtódik végre, azaz folytatódik a keresés.

Fogalmazzuk meg általánosan is képletet! A nevek a következők. [1] "mező": a csoportosítandó mező neve. [2] "tétel₁", "tétel₂", "tétel₃"...: a csoportosítandó mező egyedi értékei. [3] "név₁", "név₂", "név₃"...: a csoportnevek. Ha a csoportok száma "n", akkor a képletben álló IF függvények száma "n-1". A legbelső IF függvény második és harmadik argumentuma: "név_{n-1}" és "név_n".

```
=IF(
	SEARCH("tétel<sub>1</sub>";[mező];; 0) > 0 ||
	SEARCH("tétel<sub>2</sub>";[mező];; 0) > 0 ||
	SEARCH("tétel<sub>3</sub>";[mező];; 0) > 0;"név<sub>1</sub>";
	IF(
	SEARCH("tétel<sub>4</sub>";[mező];; 0) > 0 ||
	SEARCH("tétel<sub>5</sub>";[mező];; 0) > 0 ||
	SEARCH("tétel<sub>6</sub>";[mező];; 0) > 0;"név<sub>2</sub>";
	IF(...
```

103. ábra a csoportneveket adó számított mező képlete

numerikus tételek csoportosítása

A numerikus mező tételeit a mező értéktartományának szakaszokra bontásával csoportosíthatjuk. Kialakíthatunk azonos vagy eltérő méretű szakaszokat. Az így kialakított csoportokat sorszámmal vagy névvel azonosíthatjuk.



104. ábra a numerikus mező értéktartományának felosztása azonos- (felül) és különböző méretű (alul) szakaszokra, csoportok azonosítása sorszámmal (felül) és névvel (alul)

Először vizsgáljuk meg az azonos nagyságú szakaszok kialakításának részleteit. Először indexet, azután neveket rendelünk a csoportokhoz. Csoportosítsuk a "számla" mező tételeit százezres szakaszokkal! A 300 000 forint alatti, valamint az 599 999 forint feletti számlák egy-egy csoportot alkossanak! A számított mezőt nevezzük CSINDEX-nek. Képlete: =IF([számla] < 300000 ; 0 ; IF([számla] < 600000 ; ROUNDUP(([számla] - 300000) / 100000 ; 0) ; 4)).

A 300 000 forint alatti számlák csoport-indexét a külső IF függvény IGAZ argumentuma adja. A függvény HAMIS argumentumában álló beágyazott IF függvény IGAZ argumentuma a 300 000 és 599 999 közötti, HAMIS argumentuma az 599 999 feletti számlák indexét állítja be. A ROUNDUP függvény azonos a program KEREK.FEL függvényével.

Fogalmazzuk meg általánosan a képletet. A nevek a következők. [1] "Értékek": a csoportosítandó mező. [2] "Alsó határ" és "felső határ": az értéktartomány skálázásának első és utolsó száma. [3] "lépésköz": a szakaszok nagysága.

> =IF([értékek] < alsó határ ; o ; IF([számla] < felső határ ; ROUNDUP(([értékek] - alsó határ) / lépésköz ; o) ; csoportok száma - 1))

105. ábra a csoportindexet adó számított mező képlete, azonos nagyságú értéktartományok esetén

Az indexek segítségével már csoportosítani tudjuk a numerikus mező tételeit, de a sorszámok a kimutatásban nem túl informatívak, ezért az indexekhez "karakteres" azonosítót, vagy másként fogalmazva, címkét kell rendelnünk. Ezt a SWITCH függvénnyel tehetjük meg legegyszerűbben: =SWITCH([CSINDEX];0;"<300 000";1;"300 000-399 999";2;"400 000-499 999";3;"500 000 - 599 999";4;">=600 000").

A SWITCH függvény első argumentuma egy mezőnév vagy egy kifejezés. A mi esetünkben ez a CSINDEX. A függvény további argumentumai párban állnak. A pár első tagja a függvény első argumentumának egy lehetséges értéke, második tagja pedig az ehhez az értékhez rendelt konstans vagy kifejezés. Az argumentumlistát egy nem kötelező, önálló argumentum zárja. A függvény először kiértékeli az első argumentumát, majd a kapott értéket keresi az argumentum párosok első tagjaiban. Találat esetén kiértékeli a páros második argumentumát. Ha az érték az argumentum párosokban nem található, akkor végrehajtja az utolsó argumentumában deklarált műveletet. Amenynyiben az utolsó argumentum hiányzik, üres eredményt kapunk. Az argumentum-párosok második tagjainak és az utolsó, egyedülálló argumentumnak azonos adattípusú vagy azonos adattípusra konvertálható eredményt kell adniuk.

A tételek különböző nagyságú értéktartományokba sorolása egymásba ágyazott IF függvényekkel történik. A 104. ábra alsó vonala a "pontszámok" mező tételeinek értéktartományát szimbolizálja. Az "EREDMÉNY" csoportosító mező képlete a következő. EREDMÉNY: =IF([pontszámok] < 120 ; "nem felelt meg" ; IF([pontszámok] < 180 ; "megfelelt" ; "dicséretesen megfelelt")).

> =IF([mező] < határ1; név1; IF([mező] < határ2; név2; IF([mező] < határ1; névn; névn+1...

> > 106. ábra

numerikus mező értéktartományának különböző nagyságú szakaszokra bontása

A fenti ábra a példa képletének általánosítása: ahány határérték, annyi IF függvény áll a képletben. A legbelső IF második és harmadik argumentuma az utolsó előtti- és az utolsó csoport neve.

dátum-tételek csoportosítása

A dátum adattípusú mezők tételeit dátumegységek szerint csoportosíthatjuk a kimutatásban. A dátumegységek lehetnek: évek, félévek, negyedévek, hónapok, hetek, az év napjai, a hónap napjai és a hét napjai. A dátumegység azonosítása történhet sorszám vagy név alapján.

dátumegység	azonosító
évek	sorszám
félévek	sorszám, név
negyedévek	sorszám, név
hónapok	sorszám, név, rövidített név
hetek	sorszám
év napjai	sorszám
hónap napjai	sorszám
hét napjai	sorszám, név, rövidített név

107. ábra dátumegységek azonosítása

Ha egy kimutatásban biztosítani szeretnénk a fenti táblázatban felsorolt összes csoportosítási lehetőséget, akkor tizennégy számított mezőt kellene létrehoznunk a dátum-mezőt tartalmazó táblában. A bővítmény ezt részben meg is teszi, amikor dátum adattípusú mezőt helyezünk a kimutatás sor vagy oszlopterületére.

dátumok automatikus csoportosítása

A PowerPivot-kimutatás sor- vagy oszlopterületére helyezett dátum adattípusú mező tételeit a bővítmény év-, negyedév- és hónapok szerinti csoportosításban jeleníti meg. A csoportosítás automatikusan létrehozott számított mezőkön alapul: <mezőnév> (év), <mezőnév> (negyedév), <mezőnév> (hónap). A bővítmény létrehoz egy negyedik, rejtett mezőt is, a hónapok sorszámával, amely a hónapnevek időrendi megjelenítését szolgálja: <mezőnév> (hónap indexe).

teljesítés 💽	teljesítés (é	v) 💌	teljesítés (negyedé	v) 💌	teljesítés (hói	nap index	e) 💌	teljesítés (hónap) 💌	Oszlo
2018.02.12.	2018		Qtr1				2	febr	
2018.02.12.	2018		Qtr1				2	febr	
2018.02.13.	2018		Qtr1				2	febr	/
2018.02.14.	2018		Qtr1				2	febr	
2018.02.14.	2018			_		~	2	febr	
2018.02.15.	2018	Y Szű	rők		szlopok		2	febr	
		Sor	ok	ΣΕ	rtékek				
		teljesít	és (év) 🔻						
		teljesít	és (negyedév) 🔻						
		teljesít	és (hónap) 🛛 🔻			Sorcímké	k	*	
		teljesít	és 🔻			= 2018	1	_	
						€fe	ebr		
		Elrer	ndezésfrissítés elhalas	ztása	Frie	±m	nárc		
			raceconnosices enhands	20050		∃ ⊕ Qtr2	2		
						⊕ Qtr3 ⊕ Otr4	5 1		
						± 2019			
						Végössze	g		

108. ábra automatikusan létrehozott mezők a táblában, a pivot tábla segédablakában és a kimutatásban

A kimutatás sor- vagy oszlopterületén álló dátumegység-mezők tételeinek helyi menüjében a Csoportosítás... utasítására kattintva a program dátum-csoportosító parancstáblája jelenik meg. A tételek között ott találjuk a Napokat is, de a bővítmény kimutatásában nem tudunk beállítani több napot tartalmazó dátumegységet, például heteket, mert a panel Napok száma vezérlője inaktív. A PowerPivot-kimutatás sor- vagy oszlopterületére helyezett, csak időpontokat tartalmazó dátum adattípusú mező csupa "(üres)" címkéjű tételt eredményez.



109. ábra csak időpontokat tartalmazó mező a táblában, a PowerPivot-mezőlistán és a kimutatásban

Tetszőleges "(üres)" feliratra kattintva, jelenítsük mega a csoportosító parancstáblát és válaszszunk időegységeket! Az utasítás kiadása után a kimutatásban megjelenek a csoportosított tételek. Természetesen ezeknek a csoportosító időegységeknek is számított mező az alapja. Adatnézetben a származtatott mezők eredményeit vizsgálva látjuk, hogy a percek szerinti csoportosítást biztosító mező képlete hibás: =FORMAT([mezőnév]; "mm"). Javítsuk ki a képletet: =FORMAT([mezőnév]; "nn")! A függvényt később részletesen bemutatom.

A létrehozott segéd-mezők, függetlenül attól, hogy szerepelnek-e a kimutatásban vagy sem, mindaddig a rendelkezésünkre állnak, amíg nem adjuk ki a Csoportbontás utasítást. Ezt megtehetjük a kimutatásban álló mező egy tételének helyi menüjéből vagy a Kimutatáseszközök, Elemzés, Csoportosítás listájából.

a naptár-tábla

A dátumok egységek szerinti, teljes körű csoportosítását illetve elemzését naptár-táblával biztosíthatjuk. Ez a tábla tartalmazza az elemzendő időszak összes dátumát és dátumegységeinek azonosítóit. A naptár-tábla funkciója kettős: [1] lehetővé teszi a dátum-adattípusú mezők tételeinek csoportosítását a kimutatásban és [2] biztosítja az időszak-kezelő függvények működéséhez szükséges összefüggő dátum-tartományt.

A bővítmény naptár-táblája a lehetséges tizennégyből öt dátumegység-azonosítót, plusz egy vegyes mezőt tartalmaz. amelyben név és index is szerepel. Ha nem akarunk lemondani a többi azonosító alkalmazásáról, akkor ezt a táblát át kell alakítanunk. A változtatásokat a PowerPivot moduljával el tudjuk menteni és ezt követően már a saját naptár-táblánkat használhatjuk.

A bővítmény a naptár-táblát hol dátumtáblázatnak, hol naptárnak nevezi.

Date 💽	Év 💌	Hónap száma 💌	Hónap 💌	HHH-ÉÉÉÉ 💌	Hét napjának száma 💌	Hét napja 💌
2017. 01. 01. 0:00:00	2017	1	január	jan-2017	1	vasárnap
2017. 01. 02. 0:00:00	2017	1	január	jan-2017	2	hétfő
2017. 01. 03. 0:00:00	2017	1	január	jan-2017	3	kedd
2017. 01. 04. 0:00:00	2017	1	január	jan-2017	4	szerda
2017_01.05.0:00:00	2017	1	január	jan-2017	5	csütörtök

110. ábra a bővítmény naptár-táblája

a naptár-tábla létrehozása

A bővítmény naptár-moduljának parancsait a *Tervezés, Naptárak, Dátumtáblázat* listában találjuk. Ha az adatbázisban létezik dátum adattípusú mező, akkor az Új utasítással kérhetjük a naptár-tábla beszúrását. A megjelenő Naptár nevű tábla első mezője a dátumokat tartalmazó "Date" mező. A további mezőket töröljük le és szerkesszük meg, a mind a tizennégy dátumegység-azonosítóit tartalmazó, táblát. A mezők elnevezését és DAX képletét az alábbi táblázat tartalmazza.

mezőnév	DAX képlet
év	= YEAR([Date])
félév index	= ROUNDUP(MONTH([Date])/6; 0)
félév név	= IF([félév index] = 1 ; "I. félév" ; "II. félév")
negyedév index	= ROUNDUP(MONTH([Date])/3;0)
negyedév név	= SWITCH([negyedév index] ; 1 ; "I. negyedév" ; 2 ; "II. negyedév" ; 3 ; "III. negyedév" ; 4 ; "IV. negyedév")
hónap index	= MONTH([Date])
hónap név	= FORMAT([Date]; "mmmm")
hónap név rövidítve	= CONCATENATE(FORMAT([Date]; "mmm"); ".")
hét index*	<pre>= IF(IF(WEEKDAY(DATE(YEAR([Date]);1;1);2) < 5; WEEKNUM([Date];2); WEEKNUM([Date];2)-1) = 0; IF(WEEKDAY(DATE(YEAR([Date])-1;1;1);2) < 5; WEEKNUM(DATE(YEAR([Date])-1;12;31);2); WEEKNUM(DATE(YEAR([Date])-1;12;31);2)-1); IF(WEEKDAY(DATE(YEAR([Date]);1;1);2) < 5; WEEKNUM([Date];2); WEEKNUM([Date];2)-1))</pre>
év napjai index	= ([Date] - DATE(YEAR([Date]) - 1 ; 12 ; 31)) * 1
hónap napjai index	= DAY([Date])
hét napjai index**	= WEEKDAY([Date]; 2)
nap név	= FORMAT([Date];"dddd")
nap név rövidítve	= CONCATENATE(FORMAT([Date]; "ddd"); ".")

* az év első hete az év első csütörtökét tartalmazó hét ** a hétfő az egyes (1) a vasárnap a hetes (7)

111. ábra a naptár-tábla dátumegység-azonosító mezői

Az egyes függvények működését itt külön-külön nem ismertetem csak a hét sorszámát kiszámoló képlet felépítését magyarázom el. A hetek számát a WEEKNUM([Date]) képlettel számolhatjuk ki, amelyet nevezzünk így <hét száma>. A DAX WEEKNUM függvénye nem az EU-s szabvány szerint működik, mert az év első hetének a január elsejét tartalmazó hetet tekinti és nem az év első csütörtökét tartalmazót. Tehát azokban az években, amelyekben a január elseje péntek, szombat vagy vasárnapra esik, a WEEKNUM függvény eredményéből le kell vonnunk egyet: <hét száma> - 1. Először a vizsgált dátum évének első napját kell képeznünk. <újév>: DATE(YEAR([Date]);1;1). Ezután meg kell állapítanunk, az <újév> sorszámát a héten, a hétfőt egyesnek tekintve. <újév index>: WEEKDAY(<újév>; 2).

A következő lépésben logikai művelettel meghatározzuk, hogy az <újév index> a hétfő - csütörtök időszakra esik-e: <újév index> < 5. A művelet eredménye határozza meg a <működés>-t. Képlete: IF(<újév index> < 5 ; <hét száma> ; <hét száma> - 1). A DAX IF függvénye egy kis eltéréstől eltekintve, azonos a program HA függvényével.

Ha az év első napja péntek, szombat vagy vasárnapra esik, akkor a <működés> nullát eredményez. Ezt az állapotot egy logikai művelettel ellenőrizhetjük: <működés> = 0. Ha az eredmény IGAZ, akkor az év első napjai az előző év utolsó hetébe esnek, tehát meg kell határoznunk az előző év utolsó hetének számát. A számítás lépéseit már ismerjük.

<előző újév>: DATE(YEAR([Date]) - 1; 1; 1).

<előző újév index>: WEEKDAY(<előző újév> ; 2)

<szilveszter>: DATE(YEAR([Date]) - 1; 12; 31)

<szilveszter hét száma>: WEEKNUM(<szilveszter>; 2)

A végső képlet a fentieknek megfelelően: IF(<működés> = 0; IF(<előző újév index> < 5; <szilveszter hét száma>; <szilveszter hét száma> - 1); <működés>).

Az "év napjai index" mező képletében álló eggyel való szorzás a mező adattípusát állítja át dátumról számmá.

Végül az elkészült táblát optimalizáljuk a használatra! Határozzuk meg a név-mezők tételeinek egyedi rendezését a rendezés más mező alapján művelettel.

rendezendő mező	rendező mező
félév név	félév index
negyedév név	negyedév index
hónap név, hónap név rövidítve	hónap index
nap név, nap név rövidítve	hét napjai index

112. ábra egyedi rendezések a naptár-táblában

Rejtsük el azokat a sorszám-mezőket, amelyeket valószínűleg nem fogunk a kimutatásban szerepeltetni: félév index, negyedév index, hónap index, hét napjai index.

Kapcsolatnézetre váltva, a táblában egy hierarchikus mezőcsoport-kezdeményt találunk. Ezt töröljük a *Dátumhierarchia* név helyi menüjében álló paranccsal, majd ha szükséges, hozzunk létre saját rangsor szerinti mezőcsoportot.

Az elkészült táblát a Konfiguráció mentése paranccsal rögzíthetjük. Ezt követően a naptár-tábla, a bővítmény részeként, tetszőleges adatbázisba beszúrható. Próbaként töröljük a táblát, majd az Új utasítással hozassuk létre újra.

A fenti menüparancs csak akkor aktív, ha az adatbázisban található dátum adattípusú mező. A beszúrt naptár-tábla első napja az adatbázis dátum-mezőiben álló legkorábbi dátum évének első napja, utolsó napja pedig, az adatbázis dátum-mezőiben álló legkésőbbi dátum évének szilvesztere. A két határnapot a *Tartomány frissítése* utasítás után is megjeleníti a program, ha a naptár-tábla aktív.

A naptár-táblával lefedett időszak az adatbázis-táblák frissítését követően módosulhat. A megnövekedett dátum-tartomány határnapjait a felhasználóknak kell megadnia a *Tartomány frissítése* utasítással megjelenített parancstáblán. De hiába próbálkozunk, mert ez a funkció a magyar verzióban nem működik. Marad az amatőr módszer: a naptár törlése, majd ismételt beszúrása.

A naptár-táblán végrehajtott módosításainkat is a Konfiguráció mentése paranccsal kell rögzítenünk. A bővítmény eredeti naptár tábláját az Alapértelmezett paranccsal állíthatjuk vissza.

Természetesen a felhasználó önállóan is létrehozhat naptár-táblát, de akkor minden adatbázisban neki kell gondoskodnia a tábla importálásos vagy vágólapos létrehozásáról. A létrehozás után a menüszalag Naptárak, Megjelölés dátumtáblázatként, Megjelölés dátumtáblázatként parancsával deklarálnia kell a naptár-táblát és a dátumokat tartalmazó mezőjét. Ha a bővítmény naptár-modulját használjuk, akkor a deklaráció automatikus.

A PowerPivot része a CALENDAR és a CALENDARAUTO függvények, amelyekkel virtuális naptártáblát generálhatunk. A tábla egyetlen mezője a dátumokat tartalmazó Date mező. A CALENDAR függvény dátumtartományát két argumentumával kell meghatároznunk: első dátum, utolsó dátum. A CALENDARAUTO függvény dátumtartományának első napja, az adat-bázis legkorábbi dátuma évének január elsejéje, utolsó napja az adatbázis légkésőbbi dátuma évének szilvesztere. A függvény egyetlen argumentuma opcionális: a pénzügyi év utolsó hónapjának száma. Ha a bővítmény argumentumot észlel, akkor a visszaadott virtuális naptár első napja az adatbázis legkorábbi dátuma pénzügyi évének első napja és utolsó napja az adatbázis legkésőbbi dátuma pénzügyi évének utolsó dátuma lesz.

a naptár-tábla kapcsolatai

A naptárt össze kell kapcsolni az összes elemzendő táblával és az összes elemzendő dátum-mezővel. Tehát előfordulhat, hogy két tábla között több kapcsolatot is létre kell hoznunk. Természetesen a bővítmény több kapcsolatot esetén is mindig csak egyet fog figyelembe venni: az aktív kapcsolatot. Az aktív kapcsolat az elsőnek létrehozott, vagy a felhasználó által kiválasztott kapcsolat.





A táblák közötti kapcsolat váltása két lépésben történik: először törölnünk kell a jelenlegi kapcsolat aktív-attribútumát és csak ezután választhatunk másik kapcsolatot. Adatnézetben a művelet végrehajtásának lépései a következők. [1] A *Tervezés, Kapcsolatok, Kapcsolatok kezelése* paranccsal nyissuk meg a kapcsolat-kezelő panelt! [2] A szerkesztés funkcióval jelenítsük meg az aktív kapcsolat részleteit! [3] A bal alsó sarokban töröljük az *Aktív* feliratú jelölőnégyzet pipáját majd az OK gombbal térjünk vissza a kapcsolatkezelő ablakába! [4] Válasszuk ki a használni kívánt kapcsolatot, majd a *Szerkesztés* nyomógombbal jelenítsük meg...

Egyszerűbb a dolgunk kapcsolatnézetben. Az aktív kapcsolatot folytonos-, az inaktívat szaggatott vonal szimbolizálja. A vonal helyi menüje tartalmazza a kapcsolat-váltás parancsait: *Megjelölés inaktívként*, *Megjelölés aktívként*. Természetesen a művelet ebben a nézeteben is két lépésből áll, először inaktiváljuk a jelenlegi-, majd aktíváljuk a használni kívánt kapcsolatot.

naptár-tábla a kimutatásban

Miután aktiváltuk a szükséges naptár-tábla kapcsolatot, hozzáfoghatunk a kimutatás felépítéséhez. A naptár-tábla mezői a sor vagy oszlop területre kerülnek, nagyság szerint, csökkenő sorrendben. Például, ha a statisztikai mező tételeit három dátumegység alapján szeretném összesíteni, akkor a naptártábla mezői ebben a sorrendben állnak a mezőlista sorok területén: "év", "hónap név", "dátum".

A bővítmény és a program dátumszűrő listája azonos. Néhány félreérthető relációja pontosításra szorul. Két érték között...: a határértékeket is beleértve. Következő-, E-, Előző hét: a hét vasárnappal kezdődik. Utolsó negyedév: előző negyedév. Folyó év: az aktuális év összes napja. Évkezdettől: az aktuális év eltelt napjai, az aktuális napot is beleértve.

időszak-kezelő függvények

A bővítmény dátum-kezelő függvényeinek egy része meghatározott időszak dátumait vizsgálja. Ez az időszak lehet egy dátumegység, például év, negyedév, hónap, de lehet a felhasználó által, határnapokkal deklarált időintervallum is. Ezeket a függvényeket a bővítmény "időbeliintelligencia-függvényeknek" nevezi.

Az időszak-kezelő függvények ismertetésekor "az elemzett dátumegység" kifejezés a kimutatás sor-vagy oszlopterületen álló mezőnek, azt a tételét jelenti, amelyhez a tárgyalt függvényt tartalmazó, egyéni összesítés egy konkrét előfordulása tartozik. Másként fogalmazva, az a dátumegységtétel, amelyet az egyéni összesítés, egy konkrét előfordulása, a kimutatásban elfoglalt pozíciójával a sor-vagy oszlopterületen meghatároz, az "az elemzett dátumegység".

év	٠	félév 🚽	¥	FD	LD
2013		I. félé	v	2013.01.26	2013.01.26
		II. félé	v	2013.08.15	2013.12.22
2014		I. félé	v	2014.03.17	2014.06.18
		II. félé	v	2014.10.27	2014.12.27
2015		I. félé	v	2015.04.06	2015.04.06
		II. félé	v	2015.07.25	2015.12.08
2016		I. félé	v	2016.05.01	2016.05.09
		II. félé	v	2016.09.19	2016.09.19
FD		=FIR	ST	DATE(A[dátu	mok])
LD		=LAS	STD	ATE(A[dátur	nok])

114. ábra az elemzett dátumegység (2015 II. félév) és az őt meghatározó LD nevű egyéni összesítés egy előfordulása (2015.12.08)

Minden időszak-kezelő függvény egy vagy több dátumot keres egy meghatározott időintervallumban. Erre a dátumtartományra a függvény leírásában a "vizsgált időszak" vagy a "vizsgált dátumegység" szavakkal fogok hivatkozni. Tehát az elemzett dátumegység és a vizsgált dátumegység nem szinonimák, hanem önálló fogalmak!

Az időszak-kezelő függvények bemutatásakor meg kell különböztetnünk egy dátumegység naptári és a táblában tárolt dátumait. Vegyük például a 2017-es év januári dátumait. A naptárban ennek a dátumegységnek a legkorábbi dátuma 2017.01.01., de a táblában tárolt dátumok között a legkorábbi csak 2017.01.25. A táblában álló dátumokat a "tárolt" jelzővel különböztessük meg a "naptárban" állóktól!

Az időszak-kezelő függvények dátum-argumentumában a mezőneveket mindig a táblájuk nevével együtt kell szerepeltetnünk.

meghatározott napok adatainak feldolgozása

Az időszak-kezelő függvények funkciójuk szerint tovább csoportosíthatók. Az első csoport függvényei egy kifejezést értékelnek ki egy dátumegység meghatározott napjának illetve napjainak adataival. Másként fogalmazva, ezekkel a függvényekkel a hónap, a negyedév, vagy az év meghatározott napjának rekordjait dolgozhatjuk fel. Vegyük sorra a függvényeket és az általuk kezelt napokat!

függvény	kezelt nap
OPENINGBALANCEMONTH/-QUARTER/-YEAR	az elemzett napot tartalmazó hónapot/negyedévet/évet megelőző, tárolt hónap/negyedév/év tárolt utolsó napja
CLOSINGBALANCEMONTH/-QUARTER/-YEAR	az elemzett napot tartalmazó hónap/negyedév/év tárolt utolsó napja
TOTALMTD/-QTD/-YTD	az elemzett napot tartalmazó hónap/negyedév/év napjai az elemzett nappal bezárólag

115. ábra

a képlet-kiértékelő időszak-kezelő függvényekkel kezelt napok

A csoportot tehát három függvény, az OPENINGBALANCE, a CLOSINGBALANCE és a TOTAL függvények, hónapokat, negyedéveket és éveket vizsgáló változatai alkotják.

A kilenc függvény szintaktikája azonos. Első argumentumukkal a kiértékelendő kifejezést, második argumentumukkal a dátumokat tartalmazó mezőt kell deklarálnunk, a harmadik, nem kötelező argumentumukkal pedig logikai operátorokkal összekapcsolt szűrőfeltételeket határozhatunk meg a tábla egyik mezőjéhez.



116. ábra egy példa a képlet-kiértékelő időszak-kezelő függvények szintaktikájára

A képen látható egyéni összesítés képlete az elemzett napot tartalmazó hónap utolsó tárolt napjának, "A" és "B" kategoriás munkáinak árbevételét összesíti. A bővítmény RIGHT függvényének funkciója és szintaktikája azonos a program JOBB függvényével.

Ezek a függvények tehát meghatározott, tárolt dátumok rekordjainak feldolgozását teszik lehetővé, ezért naptár-táblát nem igényelnek.

dátumo	k 💌	számol	k	-				16						
2016.0	1.02	1	1					datumo		szamok	¥			
2016.0	1.05	1	1					2016.0	1.20	1	dátumok 😁	OBQ	CBQ	TQ
2016.0	1.11	1	1	datumok 😁	OBM	CBM	IM	2016.0	1.25	1	2016.01.20		2	1
2016.0	1.24	1	1	2016.01.02		2	1	2016.0	2.21	1	2016.01.25		2	2
2016.0	1.24	1	1	2016.01.05		2	2	2016.0	3.12	1	2016.02.21		2	3
2016.0	2.03	1	1	2016.01.11		2	3	2016.0	3.12	1	2016.02.22		2	4
2016.0	2.28	1	1	2016.01.24		2	5	2016.0	4.05	1	2016 03 12		2	6
2016.0	2.28	1	1	2016.02.03	2	4	1	2016.0	4.10	1	2016 04 05	2	4	1
2016.0	2 28	1	1	2016.02.28	2	4	5	2016.0	5.03	1	2016.04.10	2	4	2
2016.0	2.28	1	1	2016.03.05	4	3	1	2016.0	5.03	1	2016.05.03	2	4	6
2016.0	3.05		1	2016.03.11	4	3	2	2016.0	5.03	1	2016.07.26	4	1	2
2010.0	3.05	1	1	2016.03.26	4	3	5	2016.0	7.26	1	2016.09.26	7	1	2
2010.0	2.26			2016.04.02	3	1	1	2016.0	7.26	1	2010.09.20	7	1	1
2010.0	2.20							2016.0	9.26	1	2016.09.29	1	1	1
2010.0	3.20							2016.0	9.29	1	2010.10.15	<u> </u>		
2016.0	5.20		L					2016.1	0 15	1				
016.0	411	-	1											
	OB	4 =0	PEN		MONTH(SUM/ [szái	mokl) · Ald	látumokl)	OB			FOLIARTER	RI SLIMI	
	CBN	/ =C	105	INGBALANCE	MONTH	SUM([szán	nok]) · A[d	átumokl)	CBC	0 =010	SINGBALANCE	OUARTER	(SUM(

117. ábra a képlet-kiértékelő függvények működésének szemléltetése forrás-kimutatás párosokkal

TM

=TOTALMTD(SUM([számok]) ; A[dátumok])

=TOTALQTD(SUM(...

TQ

A függvény-családok éveket vizsgáló tagjainak (OPENINGBALANCEYEAR, CLOSINGBALANCEYEAR, TOTALQTD) van egy negyedik, elhagyható argumentumuk is, amellyel az év utolsó napját deklarálhatjuk, a hónap és a nap megadásával. Az argumentum értékének idézőjelek között kell állnia! A hónapot megadhatjuk névvel ("szeptember 6"), rövidített névvel ("szept-6") és sorszámmal ("9/6"). Tagoló karakter a szóköz, a kötőjel vagy a perjel lehet. Az "év vége" argumentumot nem használhatjuk üres szűrő-argumentummal. Magyarul ez a formula hibát generál: ...; ; "jan 3"). Ha egyedi év végét deklarálunk szűrők nélkül, akkor a harmadik argumentumban ismételjük meg a dátummező deklarációját. Például: OPENINGBALANCEYEAR(SUM([számok]) ; adatok[dátumok] ; adatok[dátumok] ; "szept-6").

A névkiegészítőben az OPENINGBALLANCE függvény leírása értelmetlen: "A megadott szűrők alkalmazása után kiértékeli a megadott kifejezést az előző hónap/negyedév/év végének megfelelő dátumra". Helyesen: a megadott szűrők alkalmazása után kiértékeli a megadott kifejezést az előző hónap/negyedév/év tárolt utolsó dátumára.

dátumot adó időszak-kezelő függvények

Az időszak-kezelő függvények többsége dátumot ad eredményül. Ez lehet egyetlen dátum vagy dátumok halmaza. Ezek az időszak-kezelő függvények már csak naptár-táblával működnek! Először vegyük sorra az egyetlen dátumot eredményező függvényeket.

A STARTOFMONTH/-QUARTER/-YEAR és ENDOFMONTH/-QUARTER/-YEAR függvények a nevükben szereplő dátumegységek elemzésére alkalmasak. Az argumentumukkal meghatározott dátumok közül az elemzett dátumegység tárolt, legkorábbi/legkésőbbi dátumát adják eredményül. A negyedéveket és a hónapokat vizsgáló változatok egyetlen argumentuma a dátumokat tartalmazó mező. Az éveket vizsgáló változatok második, nem kötelező argumentumával az év utolsó napját adhatjuk meg, az előző fejezetben ismertetett módon.

dátumok 🔽	számok	Ŧ						dátumok 🔽	számok	•			
2016.01.05	1	bónan n	óv 🔻	1	SOM	FOM		2013.01.26	1				
2016.01.05	1	ianuár		2	016.01.05	2016 01 22		2013.08.15	1				
2016.01.07	1	február		2	016.02.06	2016.02.06		2013.08.25	1				
2016.01.22	1	március		2	016 02 02	2016.02.00		2013.12.08	1				
2016.02.06	1	áprilie	•	2	016 04 07	2016.03.00		2013.12.22	1				
2016.03.03	1	máius		2	016.05.16	2016.04.07		2014.03.17	1				
2016.03.06	1	iútius		2	016.03.10	2016.03.31		2014.04.20	1	4.4		COV	FOY
2016.04.07	1	Junus		2	016.07.03	2016.07.29		2014.06.18	1	ev		3012 01 20	2012 12 22
2016.05.16	1	auguszt	us	2	016 11 12	2016.08.08		2014.10.27	1	20.	15	2013.01.20	2015.12.22
2016.05.31	1	docomb		2	016 12 02	2016.11.24		2014.12.27	1	20.	14	2013.12.08	2014.12.27
2016.07.03	1	decenic	e	2	010.12.05	2010.12.05		2015.04.06	1	20.	15	2014.12.27	2015.12.08
2016.07.16	1							2015.07.25	1	20.	10	2015.12.08	2016.09.19
2016.07.20	1				600	500		2015.09.21	1				
2016.07.22	1	negyede	v nev		SUQ	EUQ		2015.09.23	1				
2016.07.29	1	I. negye	dev		2016.01.05	2016.03.06		2015.12.08	1				
2016.08.08	1	II. negy	eaev		2016.04.07	2016.05.31		2016.05.01	1				
2016.11.12	1	III. negy	/edev		2016.07.03	2016.08.08		2016.05.04	1				
2016.11.24	1	IV. negy	edev		2016.11.12	2016.12.05		2016.05.09	1				
2016.12.03	1							2016.09.19	1				
		SOM	=STA	RTO	FMONTH(A[o	dátumok])	EOM	=ENDOFMO	ONTH(A[dá	itumo	ok])		
		SOQ	=STA	RTO	FQUARTER(A	[dátumok])	EOQ	=ENDOFQU	ARTER(A[dátur	nok])	
		SOY	=STA	RTO	FYEAR(B[dát	umok])	EOY	=ENDOFYE	AR(B[dátu	mok])		
							201						

118. ábra

az egyetlen dátumot eredményező függvények működésének szemléltetése forrás-kimutatás párosokkal

Az egyetlen dátumot eredményező függvények csoportjába tartoznak még a FIRSTDATE és a LAST-DATE függvények is, amelyek az elemzett dátumegység tárolt, legkorábbi/legkésőbbi dátumát adják eredményül. Ezzel a függvénypárossal tehát tetszőleges dátumegységet elemezhetünk, beleértve a féléveket és a heteket is. A két függvény egyetlen argumentuma a dátumokat tartalmazó mező.

dátumok 💌	számok	-					
2013.01.26	1						
2013.08.15	1	óv	- 5	áláv náv	-	50	ID
2013.08.25	1	CV			-	2012 01 26	2012 01 26
2013.12.08	1	02013		1. IEIEV		2013.01.20	2013.01.20
2013.12.22	1			II. Telev		2015.08.15	2015.12.22
2014.03.17	1	■ 2014		I. felev		2014.03.17	2014.06.18
2014 04 20	1			II. félév		2014.10.27	2014.12.27
2014 06 18	1	2015		I. félév		2015.04.06	2015.04.06
2014.00.10	1			II. félév		2015.07.25	2015.12.08
2014.10.27	1	2016		I. félév		2016.05.01	2016.05.09
2014.12.27	1			II. félév		2016.09.19	2016.09.19
2015.04.06	1						
2015.07.25	1						
2015.09.21	1						
2015.09.23	1			FD		=FIRSTDATE(A	A[dátumok])
2015.12.08	1			LD		=LASTDATE(A	(dátumok))
2016.05.01	1						
2016.05.04	1						
2016.05.09	1						
2016.09.19	1						

119. ábra a FIRSTDATE és a LASTDATE függvények működésének szemléltetése forrás-kimutatás párossal

Mint a bevezetőben említettem a dátumokat tartalmazó mező nevét a táblanévvel együtt kell megadnunk. Ha elfelejtkezünk erről az előírásról, akkor a következő hibaüzenetet kapjuk: A következő elem értéke nem állapítható meg: "<mezőnév>". Vagy nem létezik "<mezőnév>" elem, vagy nincs "<mezőnév>" nevű oszlophoz tartozó aktuális sor.

A NEXTDAY és a PREVIOUSDAY függvények az elemzett dátumot követő illetve megelőző naptári (!) nap dátumát keresik az egyetlen argumentumukkal meghatározott mezőben. Ha a tárolt dátumok között megtalálható a keresett nap, akkor a dátumát kapjuk eredményül, különben nem kapunk eredményt.

AZ		•	dátumok	Ŧ	számok	Ŧ				
1			2016.08.0	2	1					
2			2016.08.02	2	1	dat	tumok		NEXIDAY	
3			2016.08.0	2	1	20:	16.08.02	2.	2016.08.03	
4			2016.08.0	2	1	20:	16.08.05	j.	2016.08.06	
5			2016.08.0	3	2	20:	16.08.09).	2016.08.10	
6			2016.08.0	3	2					
7			2016.08.04	4	1					
8			2016.08.0	5	2	dát	tumok	•	PRE_DAY	
9			2016.08.0	6	1	20:	16.08.03	3.	2016.08.02	
10	0		2016.08.0	6	1	20:	16.08.06	j.	2016.08.05	
11	1		2016.08.0	9	2	20:	16.08.10).	2016.08.09	
12	2		2016.08.1	0	1					
13	3		2016.08.1	0	1	1.1				
TDAY :	=CALC	U	LATE(COUN	T([dátumok])	; N	EXTDA	(A	(dátumok]))	
DAY :	=CALC	U	LATE(COUN	T([dátumok])	; P	REVIOU	SD	AY(A[dátumo	

120. ábra

a NEXTDAY és a PREVIOUSDAY függvények működésének szemléltetése forrás-kimutatás párosokkal

dátumokat adó időszak-kezelő függvények

Az egyetlen dátumot eredményül adó függvények ismertetése után tekintsük át a dátumok halmazát eredményező időszak-kezelő függvényeket.

A NEXT és a PREVIOUS függvénycsaládok tagjai az elemzett dátumegységet követő/megelőző naptári (!) dátumegység tárolt dátumait adják eredményül, ha találnak ilyen dátumokat, az egyetlen argumentumukkal megadott mezőben. Ha nincsenek ilyen dátumok, akkor nem kapunk ered-

ményt! Mindkét függvénycsalád rendelkezik hónapokat, negyedéveket és éveket vizsgáló változattal: NEXTMONTH, NEXTQUARTER, NEXTYEAR és PREVIOUSMONTH, PREVIOUSQUARTER, PREVIOUSYEAR. A függvények egyetlen argumentuma a dátumokat tartalmazó mező.



121. ábra a NEXT és a PREVIOUS függvénycsaládok működésének szemléltetése a hónapokat vizsgáló változataikkal

A DATESMTD, a DATESQTD és a DATESYTD függvények az elemzett dátum hónapjának/negyedévének/évének tárolt dátumait adják eredményül az elemzett dátumig, az elemzett dátumot is beleértve. Egyetlen argumentumuk a dátumokat tartalmazó mező.



122. ábra a DATES függvénycsalád működésének szemléltetése

A kimutatást a kép bal oldalán álló adatok elemzésére hoztuk létre. Az első három egyéni összesítés a bejegyzéssel bíró napok számát göngyölíti, az összegzést havonta (D_MTD), negyedévente (D_QTD) és évente (D_YTD) újrakezdve. A negyedik összesítés a bejegyzésekhez tartozó számokat adja össze és ezt göngyölítve (ÖSSZ) jeleníti meg, az összegzést havonta újrakezdve.

A DATEADD függvény az elemzett dátumegység, egy időben eltolt előfordulásának, tárolt dátumait adja eredményül. A találati intervallum első napja az elemzett dátumegység tárolt, legkorábbi, utolsó napja pedig az elemzett dátumegység tárolt, legkésőbbi dátuma lesz. Argumentumai a dátumokat tartalmazó mező, az eltolás mértékét meghatározó előjeles, egész szám valamint az eltolás dátumegysége. Természetesen az elemzett dátumegység és az eltolás dátumegysége azonos. Választhatunk éveket, negyedéveket, hónapokat és napokat. A dátumegységet idézőjel nélkül kell a képletbe beírni: year, quartier, month és day.

A PARALLELPERIOD függvény szintén egy eltolással képzett dátumegység tárolt dátumait adja eredményül, de találati időintervalluma a dátumegység minden naptári napja. A PARALLELPERIOD függvényt használva az eltolás dátumegységének napokat nem választhatunk.



123. ábra a DATEADD és a PARALLELPERIOD függvények találati tartománya éveket vizsgálva

A képen látható példa képletei azonosak, DATEADD/PARALLELPERIOD (A[dátumok]; 1; year). Mindkét függvény az "A" tábla "dátumok" mezőjének 2015-ös dátumait vizsgálja, de amíg a PARALLEL-PERIOD az összes tárolt, 2015-ös dátumot eredményül adja, addig a DATEADD csak az április elseje és az október harmincegyedike által kijelölt időszakba esőket, a határértékeket is beleértve.

d	látumok 🔄	< 5	zámok	\mathbf{v}					
	2015.03.22		1						
	2015.03.22		1						
	2015.10.16		1	4.1	-1-	_		D	
	2015.11.01		1	ev	ек 1.С	-	DA_YEA	ĸ	PP_TEAK
	2015.11.02		1	20	16		1		6
	2015.11.18		1	20	1/		2		4
	2015.12.09		1						
	2016.01.08	Τ	1			1			
	2016.05.04	Τ	1			1	DA_YEAK	=0	OUNTROWS
	2016.07.17		1				PP_YEAK	=0	OUNTROWS
	2016.08.29	T	1						
	2017.06.02	Τ	1						
	2017.10.03		1						
	2017.11.29		1						



Az ábrán látható példában a 2016-os év legkorábbi illetve legkésőbbi dátuma január nyolc és augusztus huszonkilenc. Ebbe az időintervallumba egyetlen 2015-ös dátum esik: a március huszonkettő. Tehát a DATEADD függvény egyetlen dátumot eredményezett, a PARALLELPERIOD viszont az öszszes előforduló 2015-ös dátumot, azaz hat dátumot adott eredményül.

Negyedéveket elemezve a DATEADD függvény már csak hónap pontossággal dolgozik. A következő ábra ezt a működést mutatja be. A kimutatás "EH" és "UH", azaz az első hónap és az utolsó hónap nevű egyéni összesítése, a vizsgált negyedév tárolt, legkorábbi és legkésőbbi dátuma hónapjának sorszámát mutatja a negyedéven belül.

AZ	dátumok	számo	k							
001	2017.01.12	1		n évek 🕝	EH	UH	DA Q			
002	2017.02.01	1		I. negyedév	1	3				
003	2017.03.19	1		II. negvedév	1	2	2			
004	2017.03.26	1		III. negyedév	2	3	3			
005	2017.04.28	1		IV. negvedév	3	3	1			
006	2017.04.29	1								
007	2017.05.11	1		=MONTH(FI	RSTDATE(D	Oľdátumok	1)) - (ROU			
008	2017.05.19	1	EH	(MONTH(FI	(MONTH(EIRSTDATE(D[datumok]))/3.0)-1)*3					
009	2017.05.20	1			STDATE/ DI	(dátumok)				
010	2017.08.09	1	UH							
011	2017.08.29	1		(MONTH(LA	STDATE(D	[datumok]))/3;0)-			
012	2017.09.12	1	DA_Q	=COUNTROW	VS(DATEAD	D(D[dátu	mok] ;-1;			
013	2017.12.19	1								
014	2017 12 27	1								

125. ábra a DATEADD függvény működése negyedéveket vizsgálva

Az "EH" és "UH" egyéni összesítések képletében szereplő ROUNDUP függvény működése és szintaktikája azonos a program KEREK.FEL (ROUNDUP) függvényével.

A kimutatás forrását vizsgálva látjuk, hogy a negyedik negyedévben csak a negyedév utolsó hónapjába eső dátumok állnak. A DATEADD függvény tehát a harmadik negyedév utolsó hónapjába eső dátumokat adja eredményül, azaz egyetlen dátumot. A forrás harmadik negyedévében a második és a harmadik hónapra esnek a dátumok, tehát a DATEADD a második negyedév második és harmadik hónapjának dátumait eredményezi, azaz a májusi és a júniusi dátumokat. Ebben a két hónapban csak három májusi dátumot találunk. És így tovább.

Ahogy láttuk, a DATEADD függvény működése az elemzett dátumegység legkorábbi és legkésőbbi, tárolt dátumán alapszik. Amennyiben a vizsgált időszakban csak egyetlen dátum fordul elő, akkor ez a nap kerül eltolásra és csak az így képzett dátummal azonos, tárolt dátum lesz a függvény eredménye. Ha nincs a vizsgált mezőben ilyen dátum, akkor a függvény nem ad eredményt. Hónapokat elemezve és legalább két, tárolt dátumot feltételezve a két függvény már azonos eredményt ad.

Meghatározott kimutatás-elrendezésben problémát okoz a DATEADD függvény eredménytelensége. Erre a problémára a bővítmény "A következő függvény csak egybefüggő dátumtartományokkal használható: DATEADD" hibaüzenettel figyelmeztet vagy statisztikai értékek nélküli kimutatást jelenít meg.

dátumok 💌	számok 🛛 💌			
2016.01.01	1		\sim	
2016.01.20	1			
2016.01.22	d.egységek	DA_MONTH		
2016.02.09	= 2016			
2016.02.11	jan.	2		
2016.03.05	febr	. 3		
2016.03.07	máj.	. 2	SOROK Σ ÉRTÉKEK	
2016.03.08	= 2017			
2016.05.07	ápr.	1	év 🔻 DA_MONTH	•
2016.05.16	szep	ot. 2	hónap név rövidítve 🔻	
2016.06.06	okt.	3		
2016.06.30	1			
2017.02.08	1		Elrendezésfrissítés elhalasztása FRISS	ITES
2017.04.05	1	1		

DA_MONTH =COUNTROWS(DATEADD(A[datumok] ; 1 ; month))

126. ábra

hibát eredményező elrendezés a DATEADD függvény hónapos változatát használva

Az ábrán látható forrás-kimutatás páros a DATEADD függvény működését szemlélteti hónapokat elemezve. A kimutatás két sormezője a naptár-tábla "év" és "hónap név rövidítve" mezője. Ha a

hónapokat oszlopokként szeretnénk megjeleníteni és a mezőt áthelyezzük az oszlopterületre, akkor az eredménytelen cellák a fent említett hibát okozzák. A problémát az egyéni összesítés képletének módosításával orvosolhatjuk.

=IF(HASONEVALUE('naptár'[év]); COUNTROWS(DATEADD(A[dátumok]; 1; month)))

127. ábra az egyéni összesítés módosított képlete

A képlet HASONEVALUE függvényének egyetlen argumentuma a vizsgált dátumegységet tartalmazó tábla és mezőnév, eredménye logikai érték. A függvény megállapítja, hogy az egyéni összesítés az adott kimutatás-pozícióban egyetlen értéket eredményez-e vagy sem. Ha igen, eredménye IGAZ, ha nem, HAMIS eredményt kapunk. A függvény egyenértékű a COUNTROWS(VALUES ([mezőnév])) = 1 kifejezéssel. A DAX VALUES függvénye egy mező egyedi értékeit adja eredményül. Mint már láttuk az IF függvény harmadik argumentuma elhagyható. Ha hiányzik és a függvény feltétel-argumentumának kiértékelése HAMIS logikai értéket eredményez, akkor az IF üres eredményt ad. Összefoglalva: a képlet módosításával elértük, hogy eredménytelen kiértékelések helyett üres bejegyzéseket kapjunk, így a statisztikai értékek már összefüggő tartományt képeznek a kimutatásban.

A SAMEPERIODLASTYEAR függvény az elemzett év tárolt legkorábbi és tárolt legkésőbbi dátuma által meghatározott időszak tárolt dátumait adja eredményül, a határnapokat is beleértve a vizsgált évet megelőző évből. Tehát azonos a DATEADD (<dátumok> ; -1 ; year) kifejezéssel.

A DATESINPERIOD függvény egy dátum és időben eltolt megfelelője által meghatározott időszak dátumait keresi az első argumentumával meghatározott dátumok között és a megtalált dátumokat adja eredményül a határnapokat is beleértve. A bázis-dátum deklarálására a függvény második argumentuma szolgál. Az eltolás mértékét jelentő előjeles egész számot és az eltolás dátumegységét a függvény harmadik és negyedik argumentumával adhatjuk meg. Utóbbi év, negyedév, hónap és nap lehet, angolul deklarálva: year, quarter, month és day.

AZ	dátumok	számok	hónapok 🔻	DIP_DAY	
0001	2016.02.01	1	= 2016		
0002	2016.02.15	1	márc.	5	
0003	2016.02.16	1	ápr.	2	
0004	2016.02.26	1	máj.	4	
0005	2016.02.27	1	jún.	1	
0006	2016.02.29	1			
0007	2016.03.02	1	DIP_DAY	=CALCULATE(SUM(A[számok]) ;
8000	2016.03.03	1		DATESINPER	lIOD(A[dátumok] ; "2016.03.02." ; 100 ; day))
0009	2016.03.06	1	utolsó nap	2016.06.10. (="16/3/2" + 100)
0010	2016.03.14	1			
0011	2016.03.24	1			
0012	2016.04.18	1	évek 🔻	DIP_YEAR	
0013	2016.04.30	1	2016	35	
0014	2016.05.08	1	2017	55	
0015	2016.05.15	1	2018	55	
0016	2016.05.20	1	2019	12	
0017	2016.05.25	1			
0018	2016.06.01	1	DIP_YEAR	=CALCULATE(SUM(A[szàmok]) ;
0019	2016.06.15	1		DATESINPER	NOD(A[dátumok] ; "2016.03.02." ; 3 ; year))
0020	2016.06.22	1	utolsó nap	2019.03.02. (=DATUM(2016 + 3 ; 3 ; 2)

128. ábra

a DATESINPERIOD függvény működésének szemléltetése nap és év dátumegység-argumentumokkal

Az ábrán az utolsó nap dátumát kiszámító képletek természetesen a program képletei! Az egyéni összesítés képletének bevitele után megjelenő "A DatesBetween és a DatesInPeriod függvény csak dátumoszlop hivatkozását fogadja el első argumentumként" hibaüzenet a dátumok-argumentumban hiányzó táblanévre figyelmeztet. A DATESINPERIOD függvény második argumentuma tehát a keresett dátumukat tartalmazó időszak első napja, amelyet szabadon határozhatunk meg, az utolsó napot már a bővítmény számolja ki a függvény harmadik és negyedik argumentuma alapján. A DATESBETWEEN függvény szintén egy időszak, tárolt dátumait adja eredményül, de ennél a függvénynél már az időszak utolsó napját is szabadon határozhatjuk meg. Három argumentuma sorrendben, a dátumokat tartalmazó mező, az időszak első napjának és az időszak utolsó napjának deklarálására szolgál.

az időszak-kezelő függvények rendszerezése

A könnyebb megjegyezhetőség érdekében, osztályozzuk a megismert függvényeket közös jellemzőik alapján. Egy tényt az összes időszak-kezelő függvényről elmondhatunk: mind meghatározott dátumot vagy dátumokat keres egy mezőben, de némelyik nem a dátumot adja eredményül, hanem a dátumokat tartalmazó rekordokban elvégzi a felhasználó által előírt műveleteket.

Utóbbi csoportba tartozik OPENINGBALANCE, a CLOSINGBALANCE és a TOTAL függvénycsalád. Az OPENINGBALANCE függvény az elemzett dátumot megelőző, a CLOSINGBALANCE és a TOTAL az elemzett dátum dátumegységét vizsgálják. Az OPENINGBALANCE és a CLOSINGBALANCE család tagjai a dátumegység legkésőbbi dátumát, a TOTAL függvénycsalád tagjai a dátumegység összes tárolt dátumát az elemzett dátumig bezárólag, tartalmazó rekordokat dolgozzák fel. Mindhárom függvénycsaládnak van hónapos, negyedéves és éves változata. Mivel alapvetően a napok vizsgálatára szolgáló függvényekről van szó, naptár-tábla nélkül is használhatók.

függvé neve	ny	elemzett dátumegység	v. dátumegység pozíciója	vizsgált dátumegység	keresett dátum	
	-MONTH		elemzett	hónap		
OPENINGBALANCE	-QUARTER		napot tartalmazót	negyedév	logkácőbbi	
	-YEAR		megelőző	év		
	-MONTH			hónap	legkesobbi	
CLOSINGBALANCE	-QUARTER	nap		negyedév	-	
	-YEAR		elemzett	év		
	-MONTH		tartalmazót	hónap		
TOTAL	-QUARTER			negyedév	előfordulók elemzettel bezárólag	
	-YEAR			év	elemzettel bezal olag	

129. ábra a kifejezés-kiértékelő időszak-kezelő függvények

Amit fontos megjegyeznünk: az OPENINGBALANCE függvénycsalád az elemzett napot megelőző hónap/negyedév/év tárolt, legkésőbbi dátumát tartalmazó rekordokat dolgozza fel.

A további időszak-kezelő függvények dátumot vagy dátumokat adnak eredményül. Először tekintsük át az egyetlen dátumot eredményező függvényeket.

függ ne	vény ve	elemzett és vizsgált dátumegység	keresett dátum		
	-MONTH	hónap			
STARTOF	-QUARTER	negyedév	legkorábbi		
	-YEAR	év			
	-MONTH	hónap			
ENDOF	-QUARTER	negyedév	legkésőbbi		
	-YEAR	év			
FIRSTDATE		hét/hónap/negyedév/	legkorábbi		
LASTDATE		félév/év	legkésőbbi		
NEXTDAY			elemzettet követő		
PREVIOUSDAY		пар	elemzettet megelőző		

130. ábra egyetlen dátumot eredményező időszak-kezelő függvények

Az ábra tanulsága szerint ezek a függvények az elemzett dátumegységet vizsgálják és a NEXTDAY és a PREVIOUSDAY függvényeket leszámítva, az elemzett dátumegység tárolt legkorábbi illetve legkésőbbi dátumát adják eredményül.

A dátumok halmazát eredményező időszak-kezelő függvények a vizsgált időintervallumban előforduló dátumokat adják eredményül. Kivétel a napok elemzésére szolgáló DATES függvénycsalád, amely az elemzett napot tartalmazó dátumegység tárolt dátumait adja eredményül az elemzett nappal bezárólag.

A dátumok-halmazát eredményező függvények között is találunk hetek és félévek elemzésére is alkalmas függvényeket: DATESINPERIOD, DATESBETWEEN.

A DATEADD és a PARALLELPERIOD függvényekkel az elemzett dátumegység egy időben eltolt megfelelőjét vizsgálhatjuk. Ez lehet az elemzett dátumegységet megelőző/követő, tetszőleges sorszámú dátumegység.

A DATESINPERIOD függvény a vizsgált időszak első napjának, a DATESBETWEEN függvény az első és utolsó napjának, dátumegységektől független megadását teszik lehetővé.

függv nev	/ény /e	elemzett dátumegység	v. dátumegység pozíciója	vizsgált dátumegység	keresett dátum	
	-MONTH	hónap		hónap		
NEXT	-QUARTER	negyedév	elemzettet	negyedév		
	-YEAR	év	KOVELO	év	előfordulók	
	-MONTH	hónap		hónap		
PREVIOUS	-QUARTER	negyedév	elemzettet	negyedév		
	-YEAR	év	megelozo	év		
	-MTD		elemzett	hónap	1// 1.1/1	
DATES	-QTD	nap	napot	negyedév	elofordulok	
	-YTD		tartalmazó	év	elemzettel Dezalolag	
DATEADD		nap/hónap/negyedév/év	- 4 - 4	elemzettel megegyező, szűkített		
PARALLELPERIC	D	hónap/negyedév/év	eitoit	elemzettel megegyező		
SAMPERIODLASTYEAR		év	előző	év, szűkített	előfordulók	
DATESINPERIOD		hét/hónap/negyedév/	határnapokkal	elemzettel	1	
DATESBETWEEN	I	félév/év	megadott	megegyező		

131. ábra dátumok halmazát eredményező időszak-kezelő függvények

Miután a függvényeket eredményük szerinti csoportosításban rendszereztük, keressünk más, az osztályozásra alkalmas közös tulajdonságokat!

A TOTAL és a DATES függvénycsalád rokonsága szembeötlő, nemcsak tagjaik elnevezése, de az elemzendő és a vizsgált dátumegységeik is azonosak, sőt a keresett dátumok is megegyeznek.

függvény neve		elemzett dátumegység	v. dátumegység pozíciója	v. dátumegység vizsgált pozíciója dátumegység		függvény eredménye	
	-MTD			hónap		kifejezés	
TOTAL	-QTD		_	negyedév			
	-YTD	222	elemzett	év	előfordulók	creamenye	
	-MTD	пар	tartalmazó	hónap	bezárólag	dátumok	
DATES	-QTD			negyedév	bezarolag		
	-YTD			év			

132. ábra

a TOTAL és a DATES függvénycsalád összehasonlítása

Tehát mindkét függvénycsalád tagjai az elemzett napot tartalmazó dátumegység dátumait gyűjtik össze az elemzett nappal bezárólag. A különbség csupán annyi, hogy a TOTAL függvények a megtalált dátumokat tartalmazó rekordokban kiértékelik az argumentumukkal meghatározott kifejezést, a DATES függvények pedig a megtalált dátumokat adják eredményül.

idő-kezelő függvények

A DAX, akárcsak az Excel nem ismeri a negatív idő (-3:04) fogalmát. A táblák számított mezőiben a bővítmény az időpontok/időtartamok tizedes tört megfelelőjével számol és az eredményt előjel nélkül jeleníti meg (3:04). A megjelenítést szó szerint kell érteni, mert egy másik számított mezővel le tudjuk kérdezni az eredmény előjelét: SIGN(<eredmény>). A bővítmény SIGN függvénye funkciójában és szintaktikájában azonos a program ELŐJEL (SIGN) függvényével.

	[ELŐJEL]	•	$f_{\mathcal{K}} = \text{SIGN}(\text{ELT})$		
	dátum 💽	érkezés 💽	távozás 💌	ELTÉRÉS 🛛 💌	ELŐJEL 🛛 🔽
1	2016. 02. 18.	8:02	17:51	1:19	1
2	2016. 02. 19.	9:06	17:36	0:00	0
3	2016_02_22.	9:52	16:39	1:43	-1



A tábla ELTÉRÉS számított mezőjének képlete: =[távozás] - [érkezés] - "8:30". A bővítmény Dátum adattípust állít be az eredményeknek. Az ELŐJEL számított mező eredménye 1, ha az ELTÉRÉS mező tizedes tört "eredménye" pozitív, o ha nulla és -1, ha negatív.

Ha az ELTÉRÉS-t kimutatással összegezni szeretnénk, akkor a mező adattípusát Tizedes törtre kell átállítanunk, mert dátum adattípusú mező összesítéséhez SUM függvényt nem használhatunk.

Nézzünk egy példát az időtartalmak összesítésére! Egy vállalat munkatársai kötetlen munkaidőben dolgoznak. A munkaidő nyolc és fél óra. A táblában az "ELTÉRÉS" számított mezővel képeztük a munkahelyen ténylegesen eltöltött és az elvárt időtartam különbségét. Ezt fogjuk havi bontásban összesíteni a kimutatásban.

	[ELTÉRÉS]		= ROUND([tá	vozás] - [érkezé)			
	dátum 🔽	érkezés 🔽	távozás 🛛 💌	ELTÉRÉS 🔽	ELŐJEL			
1	2016. 02. 18.	8:02	17:51	0,055	hónapok	-	ÖSSZ	hiány
2	2016. 02. 19.	9:06	17:36	0	febr.		00:30	hiány
3	2016. 02. 22.	9:52	16:39	-0,072	márc.		07:22	
4	2016. 02. 23.	9:32	18:26	0,017	ápr.		00:25	hiány
5	2016 02 24	9:22	18:55	0.044	maj.		00:30	
-	2016 02 25	0.50	16-12	-0.09	vegossz	eg	06:56	
0	20,18-112, 25.	9:52	10:15	-0,09				
			-					

ossz =ABS(SUM([ELTERES])) hiány =IF(SIGN(SUM([ELTÉRÉS])) = -1 ; "hiány")

134. ábra a "negatív" időtartam kezelésének bemutatása, tábla-kimutatás párossal

A feladatot két egyéni összesítéssel oldjuk meg. Az elsővel képezzük az időtartamok "előjel" nélküli összegét, a másodikkal pedig az "előjelet". Az "össz"-ben szereplő ABS függvény működése azonos a program ABS függvényével, funkciója a negatív időpontartamok kialakulásának megakadályozása.

A DAX hat időkezelő függvénnyel rendelkezik. Az operációs rendszer által szolgáltatott, aktuális időpontot a NOW függvénnyel vihetjük be a képletbe. Az időpont a tábla frissítésekor aktualizálódik. Az egyéni összesítés képletében álló NOW aktualizálásához a kimutatást is frissíteni kell!

A HOUR, a MINUTE és a SECOND függvények egy időpont/időtartam óráinak, perceinek és másodperceinek számát adják eredményül. Működésük megegyezik a program ÓRA, PERC és MPERC függvényeinek működésével. Egyetlen argumentumuk az időpont/időtartam meghatározására szolgál. Ezek a függvények az adatok dátum részét, illetve az őket megtestesítő tizedes törtek egész részét nem veszik figyelembe.

Ez utóbbi körülmény elgondolkodtató: hogyan tudunk akkor időtartamokat összegezni, ha az összeg meghaladja az egyet? A megoldás a szám egész részének és tört részének önálló feldolgozása.

Ha például az "idő" mező bejegyzéseit szeretnénk összegezni és az eredményt "órák:percek" formátumban megjeleníteni, akkor az egyéni összesítés képlete a következő. ÓP: =TRUNC(SUM ([idő]) * 24) & ":" & MINUTE (SUM([idő])). Ezt a képletet kell egy elágazásba ágyaznunk, hogy az adatokkal nem rendelkező dátumegységekben ne jelenjenek meg a kettőspontok: =IF(TRUNC(SUM([idő])) > 0 ; ÓP). A bővítmény TRUNC függvénye működésében és szintaktikájában azonos a program CSONK függvényével.

A TIME függvénnyel időpontokat/időtartamokat képezhetünk az időegységek számának megadásával. Három argumentuma sorrendben az órák, a percek és a másodpercek számát meghatározó konstans, mezőnév vagy kifejezés.

A TIMEVALUE függvény az egyetlen argumentumával meghatározott karakterláncból időpontot képez. Például: =TIMEVALUE(A[óra] & ":01:01"). A TIMEVALUE függvénnyel kapcsolatban tudnunk kell: a szöveg adattípusú időpontokat a DAX automatikusan dátum adattípussá konvertálja, ha azt a feldolgozás megköveteli.

információs függvények

Az első négy információs függvénnyel adattípus-ellenőrzést végezhetünk. A függvények egyetlen argumentuma egy mező vagy egy kifejezés, eredményük logikai érték.

	szöveg	logikai	pénznem	szám	dátum	üres
ISTEXT	IGAZ					
ISNONTEXT		IGAZ	IGAZ	IGAZ	IGAZ	IGAZ
ISLOGICAL		IGAZ				
ISNUMBER			IGAZ	IGAZ	IGAZ	

135. ábra az adattípus-ellenőrzés függvényei és eredményük, a vizsgált elem adattípusa szerint

Az adattípus ellenőrzésén felül a számok páros vagy páratlan voltára is rákédezhetünk az ISEVEN illetve az ISODD függvényekkel.

A táblázat üres cellái HAMIS eredményt jelentenek. Ahogy látjuk az ISNONTEXT függvény üres bejegyzést tartalmazó mező vagy üres eredményt adó kifejezés vizsgálatakor is IGAZ eredményt ad!

Az ISERROR függvény hibák detektálását teszi lehetővé. Egyetlen argumentuma a vizsgálandó mező vagy az ellenőrizendő kifejezés. Eredménye logikai érték.

		🖌 A szám		B szám 🛛 💌		hányados 占	•	APERB 🗾	HIBAE 🗾
		4		1		n			HAMIS
A szám	B szám	hányados				•			
	1	0		0) #	#DIV/0!			HAMIS
	0	#ZÉRÓOSZTÓ!			ŧ	#DIV/0!			HAMIS
		#ZÉRÓOSZTÓ!	0	1	LO	D		0	HAMIS
0	1	0	•		.			N - N	1047
0	0	#ZÉRÓOSZTÓ!	U	U	, +	#010/01	_	INAIN	IGAZ
0		#ZÉRÓOSZTÓ!	0		ŧ	#DIV/0!		NaN	IGAZ
1	1	1	1	1	L 1	1		1	HAMIS
1	0	#ZÉRÓOSZTÓ!	1	0) #	#DIV/0!			IGAZ
1		#ZÉRÓOSZTÓ!	1		#	#DIV/0!		00	IGAZ



136. ábra

az ISERROR függvényt bemutató tábla, forrásával és számított mezőinek képleteivel

A táblában megjelenő NaN (Not a Number) bejegyzés jelentése: nem szám. Annak ellenére, hogy a bővítmény a m^{∞} és a m^{NaN} bejegyzéseket nem hibaként jeleníti meg, az ISERROR függvény

hibaként detektálja őket. A hibaértékeket a PowerPivot az Excel-ben megszokott módon jelöli: #SZÁM, #ÉRTÉK.

Az ISBLANK és az ISEMPTY függvények az "üres-e" kérdésre adnak választ. Az ISBLANK függvény egy üres bejegyzés, üres statisztikai érték vagy eredményt nem adó kifejezés detektálására szolgál: ha az egyetlen argumentumával meghatározott objektum üres, akkor eredménye IGAZ, különben HAMIS. Az ISEMPTY függvénnyel táblát vagy táblát eredményező kifejezést vizsgálhatunk. A rekordok nélküli tábla IGAZ, a rekordokkal rendelkező HAMIS eredményt ad. A függvény eredménye egyenértékű a COUNTROWS(<tábla>) = o logikai kifejezés eredményével.



137. ábra az ISBLANK és az ISEMPTY függvények bemutatása

A kép bal oldalán álló táblázat a "C" nevű tábla forrása, amelynek elemzésére két kimutatást hoztam létre. Először egy automatikus összesítéssel, "Elemszám - AZ", megállapítottam a havi tételszámot. A bejegyzések nélküli, tehát a nulla tételszámú hónapokat, csak egyéni összesítéssel tudtam megjeleníteni. Ez a TSZÁM, amelyben az ISBLANK függvény szolgáltatja a logikai értéket az IF függvény feltétel-argumentumának.

A VNAGY egyéni összesítéssel azt vizsgáltam volt-e az adott hónapban ötnél nagyobb szám. A képletében álló ISEMPTY függvény ellenőrzi üres-e a FILTER függvény által eredményül adott tábla, majd a kapott logikai értéket a NOT függvény cseréli le az ellentétjére. A DAX és a program NEM függvénye azonos funkciójú és szintaktikájú.

A CONTAINS függvény a "létezik-e" kérdésre ad választ. Van-e a vizsgált táblának, a függvény argumentumaival meghatározott rekordja? A függvény első argumentuma a vizsgált táblát deklarálja. További argumentumai párban állnak: mezőnév-bejegyzés. Tehát a kérdés pontosan: létezik-e olyan rekord, amelynek felsorolt mezőiben a felsorolt bejegyzések állnak?

		2	érték	osztály	kategória	dátum	AZ
			35	1	В	2017.01.08	01
			88	2	A	2017.01.11	02
			16	5	с	2017.01.19	03
			22	3	A	2017.02.05	04
c3	-	hónapok	19	3	с	2017.02.18	05
HAMIS		jan.	69	3	A	2017.02.28	06
IGAZ		febr.	86	4	С	2017.03.24	07
HAMIS		márc.	26	3	A	2017.04.17	08
IGAZ		ápr.	53	3	с	2017.04.21	09
HAMIS		mái.	27	3	A	2017.05.01	10
		1000	83	2	A	2017.05.15	11

c3 =IF(COUNTROWS(D) > 0; CONTAINS(D; [kategória]; "C"; [osztály]; 3))

138. ábra a CONTAINS függvény bemutatása forrás-kimutatás párossal

A képen látható táblázat a "D" tábla forrása. A tábla elemzésére létrehozott kimutatás "c3" egyéni összesítése, azt vizsgálja, létezik-e C3-as tétel az adott hónapban. A CONTAINS függvény argumentumaiban álló konstansok relációja egyenlő (=), de ezt nem kell jeleznünk. A függvény "szűrőfeltételei", azaz a mezőnév-bejegyzés párosai logikai ÉS viszonyban állnak egymással. Az IF függvény nélkül az egyéni összesítés a tételeket nem tartalmazó hónapok HAMIS értékét is megjelenítené.

A három-argumentumos ISONORAFTER függvény az első két argumentumát hasonlítja össze a harmadik argumentumával deklarált relációt alkalmazva. Két lehetőséggel élhetünk: [1] az első argumentum "nagyobb vagy egyenlő", mint a második argumentum, [2] az első argumentum "kisebb vagy egyenlő", mint a második argumentum.

reláció	jelzése a névkiegészítőben	alternatív jelölés
argumentum₁ >= argumentum₂	ASC	1, TRUE
argumentum₁ <= argumentum₂	DESC	o,FALSE

139. ábra az ISONORAFTER függvény harmadik argumentuma

A névkiegészítő által felajánlott ASC és DESC karakterláncok félrevezetőek, mert ezek a rendezés parancsszavai a számítástechnikában. Használjuk inkább a nulla és az egy jelöléseket. A harmadik argumentum elhagyható, ebben az esetben a nagyobb egyenlő reláció kerül alkalmazásra. A fentiekből következik, hogy a ISONORAFTER([szám1]; [szám2]; 0) képlet eredménye azonos az ISON-ORAFTER([szám1]) képlet eredményével.

	[VIZSGÁLAT	LAT] 🔻 f_{sc}^{*} =ISONORAFTER([szám1] ; [szám2])				
4	AZ 🔽	szám1 💽	szám2 💽	VIZSGÁLAT	Oszlop hozzáadása	
1	01	1	0	IGAZ		
2	02	0	1	HAMIS	(
3	03	1	1	IGAZ		
			<	~		

140. ábra példa az ISONORAFTER függvény alkalmazására

Az üres bejegyzések az összehasonlításban a nullánál kisebb, de a negatív számoknál nagyobb "számnak" minősülnek. A függvény szövegeket vizsgálva furcsa eredményt ad. Lássunk egy példát!

	[VIZSGÁLAT]	 	🖌 =ISONORAFTER([gyümölcs] ; "Banán")			
1	AZ 🔽	gyümölcs 💽	VIZSGÁLAT 🗾 🔽	Oszlop hozzáadása		
1	01	körte	IGAZ			
2	02	szilva	IGAZ			
3	03	barack	IGAZ			
4	04	alma	HAMIS			
		~ ~				

141. ábra az ISONORAFTER függvény működése szöveget vizsgálva

A VIZSGÁLAT mező képletében alkalmazott reláció a "nagyobb egyenlő". Mi nagyobb vagy egyenlő a körtében, a szilvában és a barackban a banánnál?! Szabad a gazda? A kezdőbetű. A függvény tehát csak a szövegek első karakterét vizsgálta: a kezdőbetű azonos-e a viszonyítási alap első betűjével vagy az ABC-ben utána áll-e. A példából az is leszűrhető, hogy a függvény a kis- és nagybetűket nem különbözteti meg. A függvény az üres bejegyzést az ABC-ben az A betű előtt álló "karakternek" tekinti.

A két-argumentumos KEYWORDMATCH függvény argumentumainak azonosságát vizsgálja. A vizsgálat eredménye logikai érték: ha a két argumentum azonos IGAZ, különben HAMIS eredményt kapunk. Az előbb ismertetett ISONORAFTER és a KEYWORDMATCH függvények az összehasonlítandó adatokat meghatározó argumentumai lehetnek konstansok, mezőnevek vagy kifejezések.

	[ELLENŐRZÉS	5] 🔻 [f =KEYWORDMAT LEFT([besorolá	CH(IF([szám] < 5 ; s] ; 1))	"C" ; IF([szám] < 10 ; "B" ; "A")) ;
1	AZ 🔽	szám 💽	besorolás 🛛 💌	ELLENŐRZÉS 💽	Oszlop hozzáadása
1	01	9	BB	IGAZ	
2	02	12	В	HAMIS	
3	03	3	CC	IGAZ	
4	04	15	AA	IGAZ	

142. ábra egy példa a KEYWORDMATCH függvény alkalmazására

A képen látható tábla rekordjainak besorolását ellenőriztem egy számított mezővel. Azonos-e a "besorolás" mező bejegyzésének első karaktere a "szám" nagyságától függő betűjelzéssel. A besorolás második karaktere szubjektív adat. A bővítmény LEFT függvénye azonos a program BAL függvényével.

Az információs függvények tehát logikai értéket adnak eredményül. Ha a leírásukban szereplő kérdésre igen a válasz, akkor IGAZ, ha nem, akkor HAMIS logikai értéket kapunk.

függvény neve	függvény funkciója	mit vizsgál	mire válaszol
ISTEXT			szöveg?
ISNONTEXT	adattípus		nem szöveg?
ISLOGICAL	ellenőrzés		logikai?
ISNUMBER		konstans.	numerikus?
ISEVEN	páros-páratlan ellenőrzés	bejegyzés,	páros?
ISODD		statisztikai érték,	páratlan?
ISERROR	hiba-ellenőrzés	kifejezés	hibás?
ISONORAFTER	roláciá ollopőrzác		ez a reláció közöttük?
KEYWORDMATCH	Telacio-elleriorzes		azonosak?
ISBLANK	17. 7		üras)
ISEMPTY	letezes	tábla	ures:
CONTAINS	elienolizes	laDia	létezik?

143. ábra az információs függvények rendszerezése

Összefoglalva, a tárgyalt függvények az adat típusáról, feldolgozási hibáról, meghatározott reláció megvalósulásáról valamint adat illetve rekord létezéséről adnak felvilágosítást.

logikai függvények

Az IGAZ és a HAMIS logikai értékek képzése a DAX azonos nevű, argumentum nélküli függvényeivel történik: TRUE, FALSE. A logikai értékek konstansként is elhelyezhetők a képletben és mint a programban, a bővítményben sem kell őket idézőjelezni.

A NOT, az OR és AND függvények a logikai NEM, a logikai VAGY és a logikai ÉS műveletek elvégzésére szolgálnak. A NOT egyetlen, az OR és az AND két-két argumentuma logikai adattípusú bejegyzés vagy logikai adattípust eredményező kifejezés lehet. Tehát a DAX OR és AND függvénye csak két tényezőt vizsgál, ellentétben a program "korlátlan argumentum-számú" VAGY illetve ÉS függvényével. Bonyolult feltételrendszert a függvények egymásba ágyazásával vagy logikai kapcsolókkal fogalmazhatunk meg.

Nézzünk egy példát! A következő táblában keresem azokat az "A" és "B" betűs rekordokat, amelyek 30 és 70 közötti számot tartalmaznak.

1	AZ 🔽	betű 🔽	szám 🔽	VIZSGÁLAT1 🔽	VIZSGÁLAT2 🔽
1	01	Α	33	ОК	ОК
2	02	С	54		
3	03	В	15		
4	04	С	32		
5	05	В	44	ОК	ОК
6	06	Α	42	ОК	ОК
7	07	Α	29		
8	08	С	50		
9	09	В	94		
10	10	С	66		

144. ábra feladat a logikai operátorok és a logikai függvények alkalmazására

Két számított mezőt hoztam létre: az első képletében logikai kapcsolókat a másodikban logikai függvényeket alkalmaztam! Az első mező képletetének összeállításánal figyelembe kellett vennem, hogy a DAX-ban a logikai operátorok nem egyenrangúak, először az ÉS (&&) majd a VAGY (||) operátor kerül végrehajtásra.

145. ábra a számított mezők képlete

elágazások kezelése

Az alternatívák közötti választás automatizálására két függvény szolgál a DAX-ban: az IF és a SWITCH. Az IF függvény, a program HA függvényével azonos működésű, de szintaktikája egy picit eltér tőle: a DAX-ban a második argumentum megadása kötelező. Ha a harmadik argumentum hiányzik és a feltétel-argumentum HAMIS, akkor a függvény nem ad eredményt.

A SWITCH függvény furcsa keveréke a program VÁLASZT és az Access azonos nevű függvényének. Első argumentuma egy mezőnév, vagy egy kifejezés, további argumentumai párban állnak. A párosok első eleme az első argumentum kiértékelésének egy lehetséges értéke, a páros második eleme az ehhez eredményhez rendelt konstans vagy kifejezés. Másként fogalmazva, a párosok második tagja, azt rögzíti mi történjen akkor, ha a függvény első argumentumának kiértékelése a páros első elemének értékét adja. Az eredmény-művelet párosok után megadhatunk egy önálló argumentumot is, amelyet a függvény, akkor hajt végre, ha az argumentum-párosokkal nem deklarált eredményt kell feldolgoznia.

Lássunk egy példát! A következő tábla üzletkötőit kell kategorizálnunk a bevételeik alapján. Egy millió forint alatti árbevétel "D", egy és két millió közötti "C", kettő és három millió közötti "B" és három millió forint feletti bevétel "A" kategoriát eredményez.

	üzletkötő AZ 🛛 🔽	üzletkötő 🗾 🔽	bevétel 💽	KATEGÓRIA 💽	JUTALÉK 🔽
1	001	Almási Krisztina	1 600 000 Ft	С	50 000 HUF
2	002	Berényi Adél	2 730 000 Ft	В	165 000 HUF
3	003	Dallos Krisztina	3 800 000 Ft	A	495 000 HUF
4	004	Dobai Ágota	1 410 000 Ft	С	45 000 HUF
5	005	Dobos Valéria	5 930.000 Ft	A	775 000 HUF



146. ábra a SWITCH függvény bemutatása

A SWITCH függvény első argumentumában álló kifejezés eredménye négymillió forint árbevétel felett öt, öt millió forint árbevétel esetén hat. Ezek az eredmények már nincsenek szerepeltetve különkülön eredmény-művelet argumentum-párosokkal. Tehát ők a nem deklarált eredmények. Rájuk a függvény utolsó argumentuma vonatkozik, azaz a második "A" betű.

A tábla "JUTALÉK" számított mezőjének képlete a kategoriától függő jutalékot számolja ki, ötezer forint pontossággal, felfelé kerekítve. Ezt az értéket is a SWITCH függvénnyel képezzük. A DAX CEI-LING függvénye azonos a program PLAFON függvényével: az első argumentumával meghatározott számot a második argumentumával meghatározott szám többszörösére, a számegyenesen jobbra mozdulva, kerekíti.

A SWITCH függvény utolsó argumentuma elhagyható. Ebben az esetben a nem deklarált eredmény üres bejegyzést generál. A függvény művelet-argumentumainak, beleértve az utolsó argumentumot is, azonos adattípusúaknak kell lenniük.

Az IFERROR függvény a hiba-kezelés eszköze. Szintaktikája és működése azonos a program HAHIBA függvényével. A függvény az első argumentumát értékeli ki. Ha a feldolgozás hibát eredményez, akkor végrehajtja a második argumentumával meghatározott műveletet.

	AZ		összeg 🛛 💌	évek 🗾 🗾	ÁTLAG1 0 💌	ÁTLAG2 💽	ÁTLAG3 💽
1	01		210 000 Ft	3	#HIBA	70 000 HUF	70 000 HUF
2	02		70 000 Ft		#HIBA	70 000 HUF	70 000 HUF
3	03		210 000 Ft	2	#HIBA	105 000 HUF	105 000 HUF
4	04		190 000 Ft	1	#HIBA	190 000 HUF	190 000 HUF
5	05		150 000 Ft		#HIBA	150 000 HUF	150 000 HUF
		ÁTLAG:	1 = [összeg] /	/ [évek]			
		ÁTLAG:	2 = IF(ISERRO	DR([összeg]/[é	vek]) ; [összeg]	; [összeg] / [éve	:k])
		ÁTLAG	3 = IFERROR([összeg] / [évek	(];[összeg])		



Az IFERROR függvény tehát az IF és az ISERROR függvényekkel felírható műveletsor egyszerűsítése: IF(ISERROR(művelet1); művelet2; művelet1) = IFERROR (művelet1; művelet2).

Az üres bejegyzést eredményező, argumentum nélküli BLANK függvényt nem tudjuk egyik függvény-kategoriába se besorolni, de talán az elágazásokat kezelő képletekben fordul elő leggyakrabban.

előjeles számok kezelése

A DAX SIGN és ABS függvénye azonos a program ABS és ELŐJEL függvényeivel. A SIGN függvény megállapítja az egyetlen argumentumával meghatározott szám előjelét. Eredménye mínusz egy (-1), ha a vizsgált szám negatív, nulla (0) ha a vizsgált szám nulla és egy (1), ha a vizsgált szám pozitív. Az ABS függvény az egyetlen argumentumával deklarált szám abszolút értékét adja eredményül. Ez a két függvény tehát az előjeles számok megkülönböztetését és egyszerű kezelését teszik lehetővé.

aritmetikai műveletek függvényei

A QUOTIENT és a MOD függvények az első argumentumukkal meghatározott számot elosztják a második argumentumukkal deklarált számmal, majd a hányados egész részét (QUOTIENT), illetve az osztási maradékot (MOD) adják eredményül. A két függvény működése és szintaktikája azonos a program KVÓCIENS és a MARADÉK függvényeivel. Ezt a két műveletet szokták még "egész-osztásnak" és "maradék-képzésnek" is nevezni.

Mindkét függvény hibát eredményez, ha második argumentumuk nulla vagy üres bejegyzés. A hiba elkerülésének egyik lehetséges módja a DIVIDE függvény használata. Első két argumentuma neki is az osztandó és az osztó, de rendelkezik egy harmadik argumentummal is, amellyel a függvény eredményét határozhatjuk meg nullával vagy üres bejegyzéssel való osztás esetére. Ez az argumentum csak szám-konstans vagy a BLANK() függvény lehet. A hiba javítása általánosan megfogalmazva a következő. EGÉSZOSZTÁS: INT(DIVIDE(osztandó ; osztó ; BLANK())). MARADÉKKÉP-ZÉS: osztandó - EGÉSZOSZTÁS * osztó.

A hatványozást a DAX-ban a kalap (^) műveleti jellel vagy a POWER függvénnyel írhatjuk elő. A függvény két argumentuma az alap és a kitevő. Az SQRT függvény az egyetlen argumentumával meghatározott szám négyzetgyökét adja eredményül.

Két pozitív egész szám legnagyobb közös osztóját a GCD, legkisebb közös többszörösét az LCM függvénnyel képezhetjük. A művelet végrehajtása előtt, ha ez szükséges, a függvények az argumentumaikban álló tört számokat egészre kerekítik.

A FACT függvénnyel egyetlen argumentumának faktoriálisát számíthatjuk ki. A tizedes tört argumentumot a függvény egészre csonkolja.

A PI argumentum nélküli függvénnyel a matematikai állandó értékét írathatjuk be a képletbe, tizennégy számjegy pontossággal (3,14159265358979).

csonkoló és kerekítő függvények

A csonkoló és a kerekítő függvények segítségével a számolás pontosságát határozhatjuk meg. A DAXban két csonkoló és kilenc kerekítő függvényt találunk.

A számítás pontosságát meghatározó függvények működését a számegyenes segítségével lehet a legegyszerűbben elmagyarázni: a függvény az átalakítandó szám bal/jobb oldalán, legközelebb álló, a pontosságnak megfelelő, számot adja eredményül.

A kérdés tehát az: a képzett szám az átalakítandó szám melyik oldalán áll a számegyenesen. Másként fogalmazva: a függvény az új szám képzéséhez a számegyenesen milyen irányban "mozdul el"? Négy válasz lehetséges. [1] Az irányt a kerekítés általános szabályai határozzák meg. [2] Az elmozdulás a számegyenes minden pontján azonos irányú, balra vagy jobbra. Jelölése: balra, illetve jobbra mutató nyíl. [3] Az elmozdulás a nulla irányába történik. Jelölése: a nullára mutató nyilak. [4] A képzett számot a nullától távolodva kapjuk. Jelölése: a nullától balra és jobbra mutató nyilak.

A számítás pontosságát meghatározó függvények két argumentumúak. Az elsővel a számot a másodikkal a pontosságot határozzuk meg. A csonkoló függvények ebből a szempontból kivételek. Az INT függvénynek nincs második argumentuma, mert pontossága egész. A szám tört részében álló számjegyeket nullára cseréli. A TRUNC függvény két argumentumú, de második argumentuma elhagyható. Ha a második argumentumot nem adjuk meg, akkor ez a függvény is egészre csonkol. A TRUNC függvény második argumentuma előjeles szám. Ha a pontosság pozitív szám, akkor az átalakítandó szám tört részében a pontosságban megadott számú számjegy fog állni, a többit a függvény "lenullázza". Ha a pontosság negatív, akkor a függvény az átalakítandó szám egész részében, a tizedes elválasztótól balra haladva, a pontosságban megadott számú számjegyeket nullára cseréli.

A "kerekítő" függvények pontosság argumentumuk értelmezése alapján két csoportba sorolhatók. Az első csoportba tartozók a ROUND, a ROUNDDOWN és a ROUNDUP a második argumentumuk által meghatározott számú számjegy pontosságra alakítják át az első argumentumukkal meghatározott számot. A másik csoport tagjai, az MROUND, a FLOOR, a CEILING és az ISO.CEILING függvények az első argumentumukkal meghatározott számot a második argumentumukkal meghatározott szám többszörösére alakítják át.

A TRUNC, a ROUND, a ROUNDDOWN és a ROUNDUP függvények második argumentuma negatív szám is lehet. Tehát ezekkel a függvényekkel nem csak egy tizedesjegy, két tizedesjegy, három tizedesjegy... (1, 2, 3...), hanem tízes, százas, ezres... (-1, -2, -3...) pontossággal is számolhatunk.

Az egy szám többszörösét előállító függvények működésüket argumentumaik előjelének vizsgálatával kezdik: az eltérő előjelek hibát eredményeznek!

1	szám		alap 💽	MROUND	🕨 🔽 JAVÍTOTT 💽
1		22	-4	#HIBA	24
2		27	3	#HIBA	27
3		-48	9	#HIBA	-45
4		-43	4	#HIBA	-44
	MROUND	=M	ROUND([szám] ; [a	alap])	
	MROUND J	=M	ROUND(ABS([szár	n]);ABS([ala	p])) * SIGN([szám])

148. ábra az argumentumok eltérő előjeléből adódó hiba és javítása

Az egyargumentumos EVEN és ODD függvények az argumentumukkal meghatározott számhoz, a nullától távolodva, legközelebb álló páros illetve páratlan számot adják eredményül.

Az ebben a fejezetben ismertetett függvények tehát működésük alapján három csoportba sorolhatók: csonkolók, adott számjegyre kerekítők és egy szám többszörösére kerekítők.

függvény neve	működés jellege	elmozdulás iránya	második argumentum	
INT	a a mir a lá a	<	nincs	
TRUNC	CSONKOIAS	> 0 <	elhagyható, negatív is lehet	
ROUND	kerekítés	szabály szerint	1	
ROUNDDOWN	adott számú	> 0 <	KOTElezo pegatív is lebet	
ROUNDUP	számjegyre	< 0 >	negativ is lefter	
MROUND		szabály szerint		
FLOOR	kerekítés	> 0 <	előjele azonos	
CEILING	többszörösére	< 0 >	az első argumentuméval	
ISO.CEILING	100000000000000000000000000000000000000	>		
EVEN	páros-páratlan	())	nince	
ODD	átalakítás	< 0 >	nincs	

149. ábra kerekítő és csonkoló függvények rendszerezése

A ROUND és az MROUND függvények az átalakítandó szám előjelétől függetlenül a kerekítés szabályai szerint működnek.

teszt-értékek generálása

A csoport két függvényét bonyolult képletek teszteléséhez használhatjuk. A számított objektumok frissítésekor a függvények mindig új eredményt adnak: az argumentum nélküli RAND nulla és egy közé eső törtszámot, míg a kétargumentumos RANDBETWEEN az argumentumaival meghatározott intervallumból egész számot. A RANDBETWEEN függvény első argumentuma az intervallum kezdőértékét, második argumentuma az intervallum utolsó értékét határozza meg. A függvény hibát eredményez ha nem teljesül a "kezdőérték <= utolsó érték" reláció. Az argumentumként megadott törtszámot a bővítmény egészre kerekíti, az első argumentumot a számegyenesen jobbra, a másodikat balra mozdulva el a számegyenesen.

további matematikai függvények

A DAX többi matematikai függvénye már csak a felhasználók egy nagyon szűk csoportjának készült. Ebbe a csoportba tartozik a kombinatorika két függvénye (COMBIN, COMBINA), a logaritmus-függvények (EXP, LN, LOG, LOG10), a szögfüggvények (ACOS, ACOSH, ACOT, ACOTH, ASIN, ASINH, ATAN, ATANH, COS, COSH, COT, COTH, DEGREES, RADIANS, SIN, SINH, TAN, TANH) és a valószínűségszámítás függvényei (BETA.DIST, BETA.INV, CHISQ.DIST, CHISQ.DIST.RT, CHISQ.INV, CHISQ.INV.RT, CONFIDENCE.NORM, CONFIDENCE.T, EXPON.DIST, PERMUT, POISSON.DIST).

szám és szöveg konvertálása

A DAX függvénytára hat adattípus-konvertáló függvényt tartalmaz. Ebből kettőt már ismerünk, a DATEVALUE és a TIMEVALUE függvényeket. Előbbivel szövegként megadott dátumot, utóbbival szövegként megadott időpontot konvertálhatunk dátum adattípussá.

A CURRENCY függvény az egyetlen argumentumával deklarált kifejezés eredményét, négy tizedesjegyre kerekítve pénznem adattípusra alakítja át. A logikai értékek 0,0000 Ft (HAMIS) és 1,0000 Ft (IGAZ) értéket eredményeznek.

Az azonosítóként használt számok kezelése, számtalan esetben, egyszerűbb, ha a számot szöveggé alakítjuk át. A jó olvashatóság érdekében a számok formátumát a karakterláncban is ajánlott megtartani. Ezt a lehetőséget biztosítja a bővítmény FIXED és FORMAT függvénye.

A FIXED egy számot ezres csoportosításban, két tizedesjegyre kerekítve, szöveggé alakít át. A függvény első argumentuma az átalakítandó szám deklarációja. A második, nem kötelező argumentummal a szám törtrészében álló számjegyek számát határozhatjuk meg. Az ezres csoportosítást a harmadik, elhagyható argumentumban álló, nullától eltérő számmal tilthatjuk meg. Ha nem kérjük a csoportosítását, akkor a tört rész számjegyeinek számát meg kell adnunk.

1	AZ 🗾	szám 💌	1ARG 🗾	2ARG 💌	3ARG 💽
1	01	-55278,3	-55 278,30	-55 278,300	-55278
2	02	53064,619	53 064,62	53 064,619	53065
3	03	273478,2117	273 478,21	273 478,212	273478
G	=FIXED([szám]) 2ARG	=FIXED([szám]	;3) 3ARG	=FIXED([szám] ;

150. ábra a FIXED függvény működésének szemléltetése három számított mezővel

Ha a FIXED függvény második argumentuma több tizedesjegyet ír elő a ténylegesnél, akkor a bővítmény nullával tölti fel a szám törtrészét a kívánt számjegyig. Ezt az eljárást figyelhetjük meg a képen látható tábla, 2ARG nevű számított mezőjében.

A FORMAT függvénnyel is számot alakítunk át szöveggé, de az eredményként kapott karakterlánc nem csak különböző formátumú szám, de logikai érték, dátum és időpont is lehet. Választhatunk a függvény, névvel azonosított, formátumai közül vagy saját megjelenítést írhatunk elő.

Először tekintsük át a függvény formátumait. A használni kívánt megjelenítés nevét, idézőjelek között a függvény második argumentumával kell megadnunk. A névben a kis- és nagybetűk nincsenek megkülönböztetve: FORMAT(szám deklarációja ; "formátum-név").

átalakítandó szám	"general number"	"fixed"	"standard"	"currency"	"percent"	"scientific"
0,00055	0,00055	0,00	0,00	0,00 Ft	0,06%	5,50E-04
-0,00555	-0,00555	-0,01	-0,01	-0,01 Ft	-0,56%	-5,55E-03
5,55555	5,55555	5,56	5,56	5,56 Ft	555,56%	5,56E+00
-55,5555	-55,5555	-55,56	-55,56	-55,56 Ft	-5555,55%	-5,56E+01
-5555,55	-5555,55	-5555,55	-5 555,55	-5 555,55 Ft	-555555,00%	-5,56E+03
55555,5	55555,5	55555,50	55 555,50	55 555,50 Ft	5555550,00%	5,56E+04

151. ábra a FORMAT függvény "számformátumainak" szemléltetése

Ahogy a képen megfigyelhetjük, a FORMAT függvény a szöveggé alakítás előtt, a formátumnak megfelelő számú számjegyre kerekíti a konvertálandó számot. A felsorolt hat lehetőséggel tehát "számként" formázva jeleníthetjük meg a szöveggé alakított számot. A függvény további három saját formátumával logikai adattípusnak látszó szöveget hozhatunk létre a kiindulásként megadott számból.

átalakítandó szám	"yes/no"	"true/false"	"on/off"
0	Nem	Hamis	Ki
<> 0	lgen	lgaz	Be

152. ábra

szám átalakítása logikai adattípusnak látszó szöveggé a FORMAT függvénnyel

A táblázat tanúsága szerint a FORMAT függvény a nullát HAMIS, és minden a nullától eltérő számot IGAZ logikai értéknek tekint.

A FORMAT függvény első argumentumával megadott számból dátumot és időpontot is képezhetünk. Ezek formázására a függvény hét formátumot kínál. Az alábbi táblázat első oszlopa a formátumok nevét, első sora az átalakítandó számokat tartalmazza.

		_		
formátum neve	-5,555	0	5,555	
"general date"	1899.12.25. 13:19:12	0:00:00	1900.01.04. 13:19:12	
"long date"	1899. december 25.	1899. december 30.	1900. január 4.	
"medium date"	25-dec-99	30-dec-99	04-jan-oo	
"short date"	1899.12.25.	1899.12.30.	1900.01.04.	
"long time"	13:19:12	0:00:00	13:19:12	
"medium time"	01:19 du.	12:00 de.	01:19 du.	
"short time"	13:19	00:00	13:19	

153. ábra

szám átalakítása dátumot és időpontot tartalmazó szöveggé a FORMAT függvénnyel

A FORMAT függvény választható formátumai tehát a következők. Számok: general number, fixed, standard, currency, percent és scientific. Logikai értékek: yes/no, true/false, on/off. Dátumok: generate date, long date, medium date, short date. Időpontok: long time, medium time, short time.

Egyéni megjelenítést a FORMAT függvény formátum-leíró karaktereivel határozhatunk meg. A formátum-leíró karakterekből álló kód a függvény második, idézőjelek között álló argumentuma.

Először vegyük számba azokat a karaktereket, amelyekkel az átalakítandó számot számként formázhatjuk. A kettős kereszt (#) és a nulla (0) a szám jelölője a kódban. Ezekkel a karakterekkel írjuk elő a tizedes elválasztótól balra (egész rész) és a tizedes elválasztótól jobbra (tört rész) álló számjegyek számát. A formátum-kódban a tizedes elválasztót ponttal, az ezres elválasztót vesszővel jelöljük. Vigyázzunk ne keverjük össze a kódban és a karakterláncban álló tizedes- és ezres elválasztót! A mi területi és nyelvi beállításaink mellett a tizedes elválasztó a vessző, az ezres elválasztó pedig a szóköz karakter. Lássunk néhány példát a formátum-leíró karakterek használatára! A táblázat első oszlopa az átalakítandó számot, első sora az alkalmazott kódot tartalmazza.

	átalakítandó szám	,,,,	"#.##"	"#,###"	"0.00"	"0,000"	"#,##0.00"
	55555,555	55555,555	55555,56	55 556	55555,56	55 556	55 555,56
ľ	-5,5	-5,5	-5,5	-6	-5,50	-0 006	- 5,50
ľ	0,5	0,5	,5	1	0,50	0 001	0,50
ľ	0	0	,		0,00	0 000	0,00

154. ábra

formátum-leíró karakterek a parancskódban és hatásuk a megjelenítésben

Vizsgáljuk meg az egyes parancskódok hatását a megjelenítésre! [""] Az üres formátumkód formázás nélküli megjelenítést eredményez. Ez a kód tehát a függvény "general number" megjelenítésével azonos. [#.##] A törtrész számjegyeinek száma legfeljebb kettő, ennél több számjegy esetén, a számot a függvény két tizedesjegyre kerekíti. Az egész rész számjegyeinek száma nincs korlátozva. Az önállóan álló nulla sem a törtrészben, sem az egész részben nem lesz megjelenítve. ["#,###"] A szám ezres csoportosításban, törtrész nélkül, egészre kerekítve fog megjelenni. Ha a szám nulla, akkor a függvénynek nincs eredménye. ["0.00"] A törtrész számjegyeinek száma legfeljebb kettő, ennél több számjegy esetén, a számot a függvény két tizedesjegyre kerekíti. Az egész rész számjegyeinek száma nincs korlátozva. Ha a kódban álló nulla pozíciójában az átalakítandó szám nem tartalmaz számjegyet, akkor abban a pozícióban a függvény nullát fog megjeleníteni a karakterláncban. ["0,000"] A szám ezres csoportosításban, törtrész nélkül, egészre kerekítve fog megjelenni. Ha a szám nulla, akkor a függvénynek eredménye is nulla, a kódban megadott formázással. ["#,##0.000"] Formátum-kód az eddig ismertetett elemekből. Ezres csoportosítás az egész részben, három számjegy a törtrészben. Ha a tört rész háromnál több számjegyből áll, akkor a függvény a számot három tizedesjegyre kerekíti, ha a tört rész háromnál kevesebb számjegyből áll, akkor a hiányzó számjegyeket a függvény nullákkal pótolja. Ha nincs egész rész a függvény nullát jelenít meg a tizedes elválasztótól balra.

A következő táblázatban bemutatom, hogyan rendelhetünk egyéni formai jegyeket a normál alakban illetve a százalékként ábrázolt szám megjelenítéséhez, illetve hogyan fűzhetünk pénznem jelölőt illetve mértékegységet a karakterláncban álló számhoz.

átalakítandó szám	"0.0e+0"	"0.0 %"	"#,##0.00 €"	"#,##0.0 km"
55555,555	5 , 6e+4	5555555,5 %	55 555,56 €	55 555,6 km
-5,5	-5,5e+0	-550,0 %	-5,50 €	-5,5 km
0,5	5,0e-1	50,0 %	0,50€	0,5 km
-0,055	-5,5e-2	-5,5 %	-0,06€	-0,1 km

155. ábra

szöveges normál alak, a százalék, a pénznem és a mértékegység egyéni formázása

Vegyük ismét számba az egyes parancskódok hatását a megjelenítésre! ["o.oe+o"] A normál alak kódja tehát négy elemből áll: a mantissza formátum-deklarációja, kis "e" betű vagy nagy "e" betű, "plusz jel" és a karakterisztika formátum-kódja. ["o.o %"] Százalékos megjelenítést írhatunk elő a szám formátum-deklarációja követő százalékjellel. A százalékjel előtt álló szóközt, a kódban is szó-közzel kell megadnunk. ["#,##0.00 €" és "#,##0.0 km"] A kódban a szám formátum-deklarációját megelőző és az azt követő karakterek, a szóközt is beleértve, a karakterláncban is meg lesznek jelenítve.

A FORMAT függvény lehetőséget kínál a pozitív és a negatív számok, valamint a nulla eltérő megjelenítésére is. A függvény második argumentuma pontosvesszőkkel szakaszokra osztható fel. Legfeljebb három szakaszt hozhatunk létre. Az első szakasz a pozitív számok, a második szakasz a negatív számok, a harmadik szakasz pedig a nulla formátum-deklarációját tartalmazza.

átalakítandó szám	"többlet #,##0.00 €"	"többlet #,##0.00 €;hiány #,##0.00 €"	"többlet #,##0.00 €;hiány -#,##0.00 €;0"
55555,5	többlet 55 555,50 €	többlet 55 555,50 €	többlet 55 555,50 €
55,5555	többlet 55,56 €	többlet 55,56 €	többlet 55,56 €
0	többlet 0,00 €	többlet 0,00 €	0
-55,5555	-többlet 55,56 €	hiány 55,56 €	hiány -55,56 €
-55555,5	-többlet 55 555,50 €	hiány 55 555,50 €	hiány -55 555,50 €

156. ábra a FORMAT függvény második argumentumának szakaszai

["többlet #,##0.00 ϵ "] Szakaszok nélkül a pozitív és a negatív számok, valamint a nulla megjelenítése azonos. ["többlet #,##0.00 ϵ ;hiány #,##0.00 ϵ "] Amennyiben a bővítmény a függvény második argumentumában két szakaszt talál, akkor az elsőt a pozitív számok és a nullák, a másodikat pedig a negatív számok megjelenítési előírásának tekinti. A táblázat harmadik oszlopát nézegetve, látjuk, hogy a negatív számok előtt hiányzik a mínusz jel. Ez a mi esetünkben nem hiba, hiszen a tájékoztató "hiány" felirat egyértelműen mutatja a mennyiség "előjelét". Ha a hagyományos formátumot szeretnénk, akkor a formátumkódban is szerepeltetnünk kell a mínusz jelet, ahogy azt a következő kódban látjuk. ["többlet #,##0.00 ϵ ;hiány -#,##0.00 ϵ ;0"] Háromszakaszos formátum-deklarációval nem csak a pozitív és a negatív számok, de a nulla megjelenítését is beállíthatjuk.

Miután áttekintettük a számként történő megjelenítés szabályozását, ismerkedjünk meg a dátumdeklarációk formátum-leíró karaktereivel illetve karakter csoportjaival.

megjelenítendő	kód	-50 000	50 000
rövid dátum	"с"	1763.02.06	2036.11.21.
évszám két számjeggyel	"уу"	63	36
évszám	"уууу"	1763	2036
hónapszám	"m"	2	11
hónapszám vezető nullával	"mm"	02	11
hónapnév rövidítve	"mmm"	febr	nov
hónapnév	"mmmm"	február	november
napszám	"d"	6	21
napszám vezető nullával	"dd"	06	21
napnév rövidítve	"ddd"	V	Р
napnév	"dddd"	vasárnap	péntek
negyedév száma	"q"	1	4
hét sorszáma az évben*	"ww"	6	47
nap sorszáma az évben	"у"	37	326
nap sorszáma a héten**	"w"	7	5

157. ábra

formátum-leíró karakterek és karakter csoportok szöveges dátumok formázásához

A függvény az év első hetének ("ww") a január elsejét tartalmazó hetet tekinti. A napok sorszámozása a héten ("w") hétfővel kezdődik, amelynek sorszáma az egyes. A táblázatban felsorolt formátum-leíró karakterek és karakter csoportok tetszőleges sorrendben és összetételben variálhatók. A dátumegységeket a mindennapi életben megszokott karakterekkel (pont, szóköz, kötőjel) választhatjuk el.

kód	44444
"yy. mmmm d."	21. szeptember 5.
"yyyy-mm-dd dddd"	2021-09-05 vasárnap
"mmm. d. ddd."	szept. 9. V.

158. ábra három példa a szöveges dátumok megjelenítésének szabályozására

Az időpontok megjelenítésének szabályozása a "h" az "n" és az "s" (óra, perc, másodperc) karakterekkel történik. Az egyszámjegyű időegységek vezető nulláját a karakterek megkettőzésével (hh, nn, ss) írhatjuk elő.

megjelenítendő	kód	-13,131313	91,919191
rövid dátum időponttal	"с"	1899.12.17. 3:09:05	1900.03.31. 22:03:38
óraszám	"h"	3	22
óraszám vezető nullával	"hh"	03	22
percszám	"n"	9	3
percszám vezető nullával	"nn"	09	03
másodperc-szám	"s"	5	38
másodperc-szám vezető nullával	"ss"	05	38

^{159.} ábra

szöveges időpont megjelenítését előíró karakterek és karakter csoportok

Az időegységek elválasztása a kettősponttal történik, amit a kódban kell deklarálnunk. Például a fenti táblázat két számát a "hh:nn" kód előírásai szerint a bővítmény így jeleníti meg: 03:09 és 22:03.

A formátum-leíró karakterek és karakter csoportokon kívül a FORMAT függvény második argumentumában tetszőleges szöveg megjelenítését is előírhatjuk, de a megjelenítendő szövegben álló formátum-leíró karaktereket fordított perjellel (\) meg kell jelölnünk. A backslash után álló betűt a bővítmény nem megjelenítési utasításnak, hanem betűnek tekinti. A következő táblázatban erre mutatok néhány példát.

kód	44444
"q. \neg\ye\dév"	3. negyedév
"ww.\hét"	36. hét
"y. \nap yy-mm-dd"	248. nap 21-09-05
"dd. (a \hét w. \napja)"	05. (a hét 7. napja)

160. ábra formátum-leíró karakterek a megjelenítendő szövegben

A DAX hatodik konvertáló függvénye, a VALUE, egy szövegként megadott számot tizedes törtté alakít át. A függvény egyetlen argumentuma az átalakítandó szöveg. Lássunk egy példát a függvény alkalmazására! Egy vállalat adatbázisaiban, az alkalmazottakat nevük kezdőbetűivel és belepésük évszámának utolsó két számjegyével azonosítanak. Ki szeretnénk számolni, hány éve dolgoznak már a vállalatnál az egyes kollégák. Az alábbi képen látható tábla "kód" nevű mezőjének utolsó két számjegyét fogjuk felhasználni a képletben.

1	AZ 🔽	kód 🔽	ÉVEK 🔽
1	01	BM07	10
2	02	KL06	11
3	03	FK04	13
4	04	SI10	7

ÉVEK =YEAR(TODAY()) - VALUE("20" & RIGHT([kód] ; 2))

161. ábra a példa táblája az ÉVEK számított mezővel

A bővítmény RIGHT függvényének funkciója és szintaktikája azonos a program JOBB függvényével: a második argumentumával meghatározott számú karaktert ad eredményül az első argumentumával meghatározott karakterlánc végéről.

A VALUE függvényt csak összefűzéssel létrehozott számok képzésénél és nagyon különleges esetekben kell alkalmaznunk, mert aritmetikai műveletek esetén a szövegként tárolt szám számmá konvertálását a bővítmény automatikusan elvégzi.

hierarchia kezelése

Az adatbázis-kezelés alapkövetelménye a rekordok egyértelmű azonosíthatósága. A modern számítástechnikában ezt általában egyetlen, egyedi bejegyzésekből álló, mezővel oldjuk meg. Ez a mező a tábla kulcs mezője, amelynek neve hagyományosan, "<táblanév egyes számban> AZ". Például a "kollégák" tábla kulcs mezőjének neve "kolléga AZ". Az "AZ" karakterlánc az "azonosító" szó rövidítése. A kulcs mező egy kiválasztott bejegyzése az őt tartalmazó rekord kulcsa.

A tábla kulcs mezőjét hierarchia tárolására is felhasználhatjuk. Ez úgy lehetséges, hogy a táblában létrehozunk egy mezőt a függőség tulajdonság számára és ebbe a mezőbe mindig a rekord közvetlen "felettesének" kulcsát helyezzük el. Nézzünk egy példát! Egy vállalt dolgozóinak függőségi struktúrája háromszintes.



162. ábra a vállalat függőségi struktúrája, a dolgozók rekordjainak kulcsával (kolléga AZ)

A munka a mentorok irányításával folyik. A mentor az a kolléga, aki a struktúrában közvetlenül a munkatárs felette áll. A munkák jóváírása elszámolási egységek alapján történik.



163. ábra a példa táblái

A "kollégák" tábla "mentor AZ" mezője az adott munkatárs irányítójának "kolléga AZ"-ját tartalmazza. Másként fogalmazva a "mentor AZ" egyenlő az irányító munkatárs rekordjának kulcsával. A rekordok kulcsával tárolt struktúrát a DAX hierarchia-kezelő függvényeivel dolgozhatjuk fel.

	kolléga AZ	név	mentor AZ	FÜGGŐSÉG	SZINT A	SZINT B	SZINT C	SZINTEK	MENTOR	TAR E
1	B1	Ács Ábel	A1	A1 B1	A1	B1		2	Pék Áron	HAMIS
2	C4	Bán Ernő	B5	A2 B5 C4	A2	B5	C4	3	Sós Márk	HAMIS
3	C1	Eke Adél	B2	A1 B2 C1	A1	B2	C1	3	Pap Hugó	IGAZ
4	C2	Gál Ádám	B2	A1 B2 C2	A1	B2	C2	3	Pap Hugó	IGAZ
5	A2	Kis Géza		A2	A2			1		HAMIS
6	C3	Kun Iván	B2	A1 B21C2	A1	B2	C3	3	Pap Hugó	IGAZ
	FÜGGŐSÉG	= PATH([kolléga AZ] ; [mentor AZ])			SZ	INT C	= PATHIT	em([füg	GŐSÉG] ; 3)	
	SZINT A	= PATHITEM([FÜGGŐSÉG] ; 1)			SZ	INTEK	= PATHLE	ENGTH([F	ÜGGŐSÉG])	
	SZINT B	= PATHITEM([FÜGGŐSÉG] ; 2)			Т	ARE	= PATHC	ONTAINS([FÜGGŐSÉG]	; "b2")
	MENTOR	= LOOKUP	VALUE([név]	; [kolléga AZ] ; [mento	r AZ])				

164. ábra a munkák tábla számított mezői

A teljes függőségi sort, végén a vizsgált elemmel, a PATH függvénnyel képezhetjük. Két argumentuma a kulcs mező és a "felettes" rekordok kulcsát tartalmazó mező. Az argumentumok csak mezők lehetnek. A függvény szintaktikáját és eredményét a példa "FÜGGŐSÉG" számított mezőjének képlete mutatja be. A különböző szinteken álló elemeket függőleges vonás karakter (|) választja el. A függőségi sor egyes elemeit a PATHITEM függvénnyel kérdezhetjük le. A függvény első argumentumával a függőségi sort, második argumentumával a lekérdezni kívánt elem sorszámát kell deklarálnunk. A sorszámozás egyessel, a legfelső szinttől kezdődik vagy másként fogalmazva a függőségi sorban balról jobbra halad. A függvény harmadik, nem kötelező argumentumával az eredmény adattípusát állíthatjuk be: ha értéke nulla (o) vagy nem adjuk meg, akkor szöveg adattípusú eredményt kapunk, ha értéke egy (1), akkor az eredmény egész szám adattípusú lesz. A "kollégák" tábla "SZINT A, B, C" számított mezői a PATHITEM függvényt alkalmazzák.

A PATHITEMREVERSE függvény csak a függőségi sor elemeinek sorszámozásában tér el a PATH-ITEM függvénytől: a számozás a legalsó szinttől kezdődik, vagy másként fogalmazva a függőségi sorban jobbról balra halad.

A függőségi sor elemeinek számát a PATHLENGTH függvénnyel határozhatjuk meg. Egyetlen argumentuma a függőségi sor deklarációja. Példánkban a PATHLENGTH függvényt a "SZINTEK" mező képletében alkalmaztuk.

A PATHCONTAINS függvénnyel a kulcs mező egy bejegyzését kereshetjük a függőségi sorban. Másként fogalmazva, a függvény első argumentumával meghatározott függőségi sorban szerepel-e, a függvény második argumentumában, idézőjelek között álló, kulcs mező bejegyzés, a kis- és nagybetűk megkülönböztetése nélkül. Igen válasz esetén a függvény eredménye IGAZ, különben HA-MIS. Példánkban a "TAR E" számított mező képletében láthatjuk a függvényt.

A MENTOR számított mező képletében álló LOOKUPVALUE függvénnyel az irányító munkatárs nevét íratjuk ki. A deklarált argumentumokkal a következő művelet elvégzésére utasítjuk a függvényt: keresd ki a "kollégák" táblában azt a rekordot, amelynek "kolléga AZ" mezőjében (második argumentum) a "mentor AZ" (harmadik argumentum) értéke áll és ennek a rekordnak a "név" mezőjében (első argumentum) álló bejegyzést add eredményül. Elszakadva a példától, a LOOKUPVALUE függvény az első argumentumával meghatározott mező egy bejegyzését adja eredményül, abból a rekordból, amelyet további mezőnév-bejegyzés argumentum-párosai határoznak meg.

hierarchia a kimutatásban

Visszatérve példánkhoz, a név és hierarchia szerinti kimutatásos elemzéshez három számított mezőt kell létrehoznunk: SZINT AN/BN/CN. Ez a három mező mutatja majd meg, ki áll közvetlenül az adott kolléga felett a hierarchia különböző szintjein. A függőségi sort és a neveket egyetlen képlettel állapítjuk meg: = LOOKUPVALUE([név]; [kolléga AZ]; PATHITEM(PATH([kolléga AZ]; [mentor AZ]); <szint sorszáma>)).

SZINT AN	×.	SZINT BN	1	SZINT CN	-	Összeg - egység
Kis Géza		(üres)		(üres)		7
		Réz Benő		(üres)		12
		Sós Márk		(üres)		7
				Bán Ernő		15
				Sas Béla		1
Kis Géza Összeg					42	
Pék Áron		(üres)		(üres)		7
		Ács Ábel		(üres)		3
		Pap Hugó		(üres)		12
				Eke Adél		12
				Gál Ádám		2
				Kun Iván		11
		Tar Emil		(üres)		5
Pék Áron Összeg					52	
Végősszeg					94	

165. ábra az elszámolási egységek automatikus összesítése hierarchia szerint

Az ábrát vizsgálva megállapíthatjuk, hogy Kis Géza és alárendeltjeinek együttes teljesítménye negyvenkét elszámolási egység. Maga Kis Géza hét egységnyi munkát végzett el. A függőségi struktúrában a második szinten álló Réz Benő és beosztottjainak együttes teljesítményét már nem tudjuk

leolvasni a kimutatásról, mert az áttekinthetőség javítása érdekében, a további részösszegek megjelenítést letiltottam. Maga Réz Benő szintén hét egységnyi munkát végzett el.

hierarchikus mezőcsoport

Vegyük számba azt a műveletsort, amellyel a hierarchia szerinti kimutatás-elrendezést létrehoztuk. [1] A kulcs mező bejegyzéseivel tároltuk a munkatársak közvetlen feletteseit. [2] A PATH függvénnyel megállapítottuk a függőségi sort. [3] A PATHITEM függvénnyel létrehoztuk a "szint" mezőket, [4] amelyeket azonos kimutatás-területre helyeztük, [5] a függőségi sor szerinti sorrendben.

Az utolsó két művelet elvégzését a bővítmény "hierarchikus mezőcsoport" szolgáltatása segíti. A modult használva a "szint" mezőket, a függőségi sor szerinti sorrendben, csoportba foglalhatjuk. A létrehozott objektum a hierarchikus mezőcsoport. Azonosítása névvel történik. A hierarchikus mezőcsoport a PowerPivot mezőlista része. Mezői a listán önállóan is elérhetők.

A csoport létrehozását kapcsolatnézetben, a tábla címsorának Hierarchia létrehozása parancsgombjával kezdeményezhetjük. Az új objektum az utolsó mezőnév alatt jelenik meg, Hierarchia<sorszám> felirattal, amely a felhasználói nevével felülírható.

Az egyes mezőket menüjük Hozzáadás hierarchiához parancsával vagy húzással helyezhetjük el az új objektumba. Ha húzással állítjuk össze a csoportot, akkor sorrendben, a csoport nevére (nem alája!) kell húzni a mezőneveket. A bővítmény a csoportban álló mezőkhöz két nevet jelenít meg: a hierarchikus mezőcsoportban használatos nevet és zárójelek között, az eredeti nevet. Utóbbi a PowerPivot szóhasználatban a "forrásoszlop név".

A hierarchiában álló mező menüje tartalmazza a mező pozíciójának módosítására, csoportból történő eltávolítására, a csoportban használandó nevének megadására és eredeti nevének elrejtésére szolgáló parancsokat.



166. ábra

hierarchikus mezőcsoport létrehozása és megjelenítése kapcsolatnézetben és a mezőlistán

A kimutatásba felvett hierarchikus mezőcsoport mezői közül mindig csak az első tételei lesznek megjelenítve, az alárendelt mezők tételeit kibontás művelettel tehetjük láthatóvá. Ha csak egyetlen tétel tartalmát akarjuk látni, akkor a tételnév előtt álló plusz jeles (+) pici vezérlővel, ha a teljes mező alárendeltjeire vagyunk kíváncsiak, akkor a *Kimutatáseszközök, Elemzés, Aktív mező, Mező kibontása* utasítással. A "kibontás" művelet ellentéte az "összecsukás". Végrehajtása a mínusz jeles (-) pici vezérlővel vagy *Kimutatáseszközök, Elemzés, Aktív mező, Mező összecsukása* paranccsal történik.
A hierarchikus mezőcscsoportnak a szűrőlistája is hierarchikus, ezért mezőinek számától függetlenül csak egyetlen szűrőlistával rendelkezik. Ennek vezérlője a rangsorban elől álló mező nevének cellájában áll. A hierarchikus mezőlista nem a bővítmény "találmánya", hiszen az Excelben ilyen a dátum adattípusú mezők szűrőlistája is.

kollégák	vezetők	 képviselők 	ügynökök	Összeg – egység
✓ munkatársak	🗄 Kis Géza			42
ezetők	🗏 Pék Áron	🗏 (üres)	(üres)	7
épviselők		🗏 Ács Ábel	(üres)	3
wnökök		🖃 Pap Hugó	(üres)	12
Tauthtimes			Eke Adél	12
I OVADDI MEZOK			Gál Ádám	É <u>r</u> tékszűrők
Kollega AZ			Kun Iván	
mentor AZ		🗏 Tar Emil	(üres)	Keresés: vezetők
SZINT AN SZINT BN SZINT CN				 ₩ Kis Géza Pék Áron (üres) (üres) Ács Ábel Ács Ábel Úres) Pap Hugó Eke Adél Eke Adél Gál Ádám Gál Ádám Cas Faire

167. ábra hierarchikus mezőcsoport a kimutatásban, és a csoport szűrőlistája

A kimutatásban csak Pék Áron alárendeltjeit "bontottam ki". Ezt a struktúrát képezi le, a kép jobb szélén álló, szűrőlista. Következesen szűrőlistát írok, az egyszerűség kedvéért, de természetesen mindenütt az egyedi bejegyzések listájáról van szó! ... amelyet a szűrőlista tartalmaz.

Ahogy láttuk, a kimutatásba felvett hierarchikus mezőcsoportnak a bővítmény csak az első, a hierarchia lefelsőbb szintjét mutató mezőjét jeleníti meg. Ez a módszer javítja a kimutatás áttekinthetőségét, de megnehezíti a struktúra vizsgálatát, mert nem látjuk az alárendelteket. Ezen segít, a rettenetes nevű, "leásás" (egy szinttel lejjebb) művelet, amely megjeleníti a kiválasztott tétel alárendeltjeit. Nem kibontja a tételt, mert a végrehajtást követően a hierarchia első szintje eltűnik és csak a második szintet látjuk.

vezetők	 Összeg – egység 	képviselők 🗐	Összeg – egység	ügynökök 🚽	Összeg – egys
🗄 Kis Géza	42	🗄 (üres)	7	(üres)	
🗄 Pék Áron	52	🗄 Ács Ábel	3	Eke Adél	
Végösszeg	6 94	🕀 Pap Hugó	37	Gál Ádám	1
	100	🕑 Tar Emil	5	Kun Iván	6
		Végösszeg	52	Végösszeg	<i>w</i>

168. ábra

a rettenetes nevű "leásás" (egy szinttel lejjebb) művelet fázisai

A kép bal oldalán a kiindulási helyzetet látjuk: hierarchikus mezőcsoport a kimutatás Sorok területén. Jelenítsük meg Pék Áron alantasait! Kattintunk a tétel-címkén, majd kiadjuk a *Kimutatáseszközök, Elemzés, Aktív mező, Leásás* utasítást. A művelet eredményét a kép közepén látjuk. A hierarchia ezen a szintjén már dönthetünk, hogy visszatérünk az előző megjelenítéshez, a *Felhatolás* (egy szinttel feljebb) utasítással, vagy inkább Pap Hugó alárendeltjeit kívánjuk látni (egy szinttel lejjebb). A jobb oldali kimutatás szerint az utóbbit választottuk. Erről a szintről már csak "visszafelé vezet az út". Választhatunk, hogy a felettünk álló szintet akarjuk-e megjeleníteni (*Felhatolás* utasítás) vagy a nyilas vezérlő listájából választjuk ki a mezőcsoport egyik elemét.

Összefoglalva, ezzel a modullal csak a hatalmi struktúra egyetlen ágának, a hierarchia egy választott szintjén álló elemeit jeleníthetjük meg. Tehát egy szelektív nézetet látunk, amelyet a bővítmény a mezőcsoport szűrőgombján megjelenített pici tölcsérrel jelöl. A kiválasztott tételcímkék mellet megjelenített villámos-nagyítos parancsgom is ennek a modulnak a vezérlője.



169. ábra a "Gyorsvizsgálat" vezérlő és megjelenített parancs-listája

A képen látható panel tartalmazza aktuális tétel címkéjét, az adatbázis teljes mezőlistáját valamint az aktuális tételhez tartozó lehatolás-felhatolás műveletet.

A kimutatás Szűrők területén a bővítmény a mezőcsoportot egyetlen objektumként jeleníti meg, a már ismertetett strukturált szűrőlistával. A hierarchikus mezőcsoport az Értékek kimutatás-területen nem állhat.

egyedi bejegyzések lekérdezése

A DAX öt függvényt biztosít egy mező egyedi bejegyzéseinek összegyűjtésére: DISTINCT, VALUES, FILTERS, ALLSELECTED és ALLNOBLANKROW. A függvények egyetlen argumentuma a vizsgálandó mező: FÜGGVÉNYNÉV('táblanév'[mezőnév]). A mezőben álló üres bejegyzéseket mind az öt függvény "egyedi bejegyzésnek" ítéli. Másként fogalmazva, az üres bejegyzéseket tartalmazó mező vizsgálatakor, az ismétlődések eltávolítása után kapott lista egy üres elemmel egészül ki.

Egy mező megjelenített bejegyzéseit a mező saját, közvetlen szűrője és a többi mezőhöz beállított szűrők, a közvetett szűrők befolyásolják. A közvetlen és közvetett szűrők kezelésének módja a bemutatandó függvények első tulajdonsága.

Talán még emlékszünk rá, hogy a kapcsolat több oldali táblájában állhatnak olyan rekordok is, amelyek nem kapcsolódnak az egy oldali tábla egyetlen rekordjához sem. A kapcsolat nélküli rekordokat a bővítmény egy kalap alá veszi és az egy oldali táblában egy üres rekorddal felelteti meg. Az üres "kategoriát" a bővítmény csak a kimutatásban jeleníti meg, a táblában láthatatlan. A tárgyalt függvények második tulajdonsága, hogy az eredményükben szerepel-e ez az üres kategoria vagy sem.

Az egyedi bejegyzéseket összegyűjtő függvények működését adatnézetben a legegyszerűbb szemléltetni. Vegyük példának a "termékek-eladások" táblapárost.

1	név	Pa 💌	ár		szín 🔽
1	Α			50	fekete
2	В			100	sárga
3	С			150	piros
4	D			200	fekete

	dátum 💌	név 🐕 🖪	darab 💽
1	2018. 01. 12.	E	5
2	2018. 02. 14.	Α	4
3	2018. 05. 27.	С	5
4	2018. 06. 05.	E	1
5	2018. 06. 09.	с	4
6	2018. 06. 14.	E	3
7	2018. 06. 21	D	4

170. ábra a példa táblái: "termékek" és "eladások"

Ahogy a képen látjuk a kapcsolat a táblák "név" mezőivel lett létrehozva. A kapcsolat egy oldalán a "termékek" tábla áll. Az "eladások" táblában megfigyelhetünk egy "E" nevű terméket, amely a "termékek" táblában nincs regisztrálva. Tehát létezik az a bizonyos üres kategoria a "termékek" táblában, amely majd csak a kimutatásban lesz megjelenítve.

Először kérdezzük le az "eladások" tábla "név" mezőjének egyedi bejegyzéseit. Mivel a függvények eredményét nem tudjuk megjeleníteni, ezért az egyéni összesítések képletében megszámláltatjuk az eredményül kapott bejegyzéseket: =COUNTROWS(<függvénynév>('eladások'[név])). A vizsgálatot először szűrés nélkül, majd közvetlen szűréssel, végül közvetett szűréssel végezzük el.

1	dátum 🔽	név 🛛 💀 🔽	darab						1					
1	2018. 01. 12.	E		5	dátum 💽	név 🛛 🐕 🜌	darab					_		
2	2018 02 14	Δ		4	2018. 01. 12.	E		5	dátum	×	név 🏰	* (darab	3
3	2018 05 27	C		5	2018. 02. 14.	с		5	2018. 01.	12.	E			1
Ĕ	2010.03.27.			-	2018, 05, 27,	E		1	2018. 02.	14.	E			1
	DIST_t: 5	FILT_t: 5				-		-	2018. 05.	27.	в			2
	VALU_t: 5	ALLS_t: 5			DIST_t: 3	FILT_t: 3				-		-		_
		ANBR t 5			VALU_t: 3	ALLS_t: 3			DIST_t: 2		FILT_t: 5			
				_		ANBR t:5			VALU_t: 2		ALLS_t: 5			
L								-			ANBR_t: 5			

171. ábra az "eladások" tábla vizsgálata

A példa jól mutatja, hogy a DISTINCT és a VALUES függvények teljesen egyformán működnek a kapcsolat több oldali táblájában: figyelembe veszik a közvetlen és a közvetett szűrőket is. A FILTERS és az ALLSELECTED már csak a közvetlen szűrőt veszi figyelembe, míg a ALLNOBLANKROW egyiket sem.

Ezután vizsgáljuk meg a kapcsolat egy oldalán álló, "termékek" táblát. Az egyéni összesítések képlete a következő: =COUNTROWS(<függvénynév>('termékek'[név])). A vizsgálatot először szűrés nélkül, majd közvetlen szűréssel, végül közvetett szűréssel végezzük el.

1	név 📲 🖬	ár 🔽	szín 💽]				5				
1	A	50	fekete	név	Pa 🜌	ár 🗾	szín 🗾 💌					
2	в	100	sárga	A		50	fekete	név	1	ár 💌	szín	3
3	c	150	piros	в		100	sárga	А		50	fekete	
Ē		150	pilos					D		200	fekete	
	DIST_e: 4	FILT_e: 5		<u> </u>								
	VALU_e: 5	ALLS_e: 5		DIST_	e: 2	FILT_e: 3						_
				VALU	e: 3	ALLS_e: 3		DIST_	e: 2	FILT_e: 5		
-		ANDIC_C. 4				ANBR e:4		VALU_	e: 3	ALLS_e: 5		
L-										ANBR_e: 4		
									_			_

172. ábra a "termékek" tábla vizsgálata

A kép tanúsága szerint a kapcsolat egy oldalán álló táblában a DISTINCT és az ALLNOBLANKROW függvények nem veszik figyelembe a láthatatlan üres kategoriát, a többiek igen.

Az egyedi bejegyzéseket összegyűjtő függvények kimutatásba helyezésével derül fény a függvények harmadik tulajdonságára. Vegyünk egy példát ennek a tulajdonságnak a bemutatására is. Egyetlen táblát vizsgálunk, amelynek neve "A".

AZ	betű	római	szín	DIS	Г	ró	mai 💌				FILT		római	Ŧ		
01	Α	- I.	barna	beti	i [T	Т.	н.	Ш.		betű	-	Т.		н.	ш.
02	Α	- I	fekete	Α			1	1] [Α		2		2	
03	Α	- I.	fekete	В				1			в				2	
04	Α	- I.	kék	C			1		2		С		2			2
05	Α	- II.	kék													
06	В	- II.	barna	VAL	J	rói	mai 💌				ASEL		római	-		
07	В	П.	barna	betí	· [•	Т.	н.	III.		betű	-	ь.		н.	III.
08	В	- 11.	kék	Α			1	1			А		2		2	
09	В	III.	fehér	В				1			в				2	
10	С	- I	kék	С			1		2		С		2			2
11	С	- I.	zöld													
12	С	- I	zöld	szí	1				¥		ANBR		római	Ŧ		
13	С	- I.	zöld								betű	-	ь.		н.	Ш.
14	С	III.	barna	b	arna	а		fehér			Α		6		6	
15	С	III.	kék			_		1.41			В				6	
16	С	III.	kék		eket	e		кек			С		6			6
17	С	III.	kék	s	árga			zöld								
18	С	111.	sárga													

173. ábra egyedi bejegyzéseket összegyűjtő függvények a kimutatásban

A képen a tábla forrását, az elemzésére létrehozott öt kimutatást és közös külső szűrőjüket látjuk. Minden kimutatás egy-egy egyéni összesítést tartalmaz, képletükben a tárgyalt függvényekkel: =COUNTROWS(<függvénynév>(A[szín])). Csak a kék és a sárga rekordokat vizsgáltuk.

A FILTERS, az ALLSELECTED és az ALLNOBLANKROW függvényeket tartalmazó egyéni összesítések nem a megszokott módon viselkednek: nem a sor és oszlopmező tételei által meghatározott rekordok statisztikai értékét, hanem a teljes rekord-halmazra vonatkozó értéket jelenítenek meg, az összes nem üres "kategoriában". Másként fogalmazva: bontatlan statisztikai értéket szolgáltatnak.

Foglaljuk össze az egyes függvények jellemzőit. A DISTINCT és a VALUES figyelembe veszik a közvetlen és közvetett szűrőket is, de a DISTINCT függvény a láthatatlan bejegyzést nem veszi figyelembe az egy oldali táblában. A FILTERS és az ALLSELECTED csak a közvetlen szűrőket figyelik, de mindketten látják az üres kategoriát. Az ALLNOBLANKROW nem veszi figyelembe a szűrőket és eredményéből hiányzik az egy oldali tábla üres kategoriája is. A FILTERS, az ALLSELECTED és az ALL-NOBLANKROW bontatlan statisztikai értéket szolgáltatnak a kimutatásban.

függvény neve	közvetlen szűrés figyelése	közvetett szűrés figyelése	üres kategoria figyelése	kimutatás-szerkezet figyelése		
DISTINCT	ia	o.p.	nem	igan		
VALUES	Ig	en		igen		
FILTERS	igan	nom	igen	nem		
ALLSELECTED	igen	nem				
ALLNOBLANKROW		nem		-		

174. ábra az egyedi bejegyzéseket összegyűjtő függvények rendszerezése

Az ALLSELECTED és az ALLNOBLANKROW függvényeknek van egy tábla-argumentumos változatuk is, de a tárgyalt függvényektől eltérő funkciójuk miatt, a következő fejezetben ismertetem őket.

szűrő-kezelés a képletben

Az egyéni összesítések képletének kiértékelésekor szabályozni tudjuk a szűrők figyelembe vételét. Két függvény áll rendelkezésünkre erre a célra: az ALL, az ALLEXCEPT valamint az előző fejezetben tárgyalt ALLSELECTED és az ALLNOBLANKROW tábla-argumentumos változatai.

Az ALL függvény két változatát használhatjuk: a tábla-argumentumos és a mező-argumentumos változatát. Az előbbi esetében csak egyetlen táblanevet kell megadnunk argumentumként, és a bővítmény figyelmen kívül hagyja a tábla összes szűrőjét a kifejezés kiértékelésekor. A másik változat argumentumai mezőnevek, amelyek szűrőit a bővítménynek nem kell számításba venni, de az argumentumok között nem szereplő mezőkéit igen. Az ALLEXCEPT függvény az ALL mező-argumentumos változatának ellentéte: melyik szűrőt kell figyelembe venni az egyéni összesítés képletének kiértékelésekor. Argumentumlistája: tábla ; mező₁ ; mező₂ ; mező₃... mező_n. A függvény az első argumentumával meghatározott táblában csak azoknak a mezőknek a szűrőit veszi figyelembe, amelyeket a további argumentumai deklarálnak.

Példaként vegyük egy táblát, neve "A", tíz rekorddal és néhány mezővel. Három egyéni összesítés hoztam létre a függvények működésének bemutatására. A táblát adatnézetben vizsgáljuk.

l	AZ 🌌	betűk 🗾	számok 💌	római		AZ 🔽	betűk 🛃	számok 🔽	rć
L	6	Α	1	II.	1	1	Α	2	I.
2	7	Α	2	III. (2	5	Α	2	I.
3	8	В	1	П.	3	6	Α	1	11.
4	9	В	2	I.	4	7	Α	2	П
5	10	Α	1	П.	5	10	Α	1	П
	ALLt: 10	ALLm: 5	ALLE: 10		-	ALLt: 10	ALLm: 10	ALLE: 5	



175. ábra az ALL és az ALLEXCEPT függvények bemutatása

Először az "AZ" mezőhöz állítottam be szűrőt: Csak az ötnél nagyobb azonosítójú rekordokat jelenítsd meg! Ezt láthatjuk a kép bal oldalán. A második esetben a "betűk" mezőt szűrtem: Csak az "A" betűs rekordokra vagyok kíváncsi! Ennek a műveletnek az eredményét a kép jobb oldalán helyeztem el. Az "ALLt nevű egyéni összesítés képletében álló ALL függvény tábla-argumentumos változatának működése egyértelmű. A táblát szűrők nélküli állapotában vizsgálja. Az "ALLm" és az "ALLE" nevű egyéni összesítésekben a mező-argumentumok a figyelmen kívül hagyandó (ALL) illetve a figyelendő (ALLEXCEPT) mezőket határozzák meg. Másként fogalmazva, mely mezők szűrői nem befolyásolják (ALL), illetve, mely mezők szűrői befolyásolják (ALLEXCEPT) az eredményt.

Az ALL mező-argumentumos és az ALLEXCEPT függvény szintaktikáját összehasonlítva látjuk, hogy utóbbi első argumentumával a táblát is deklarálnunk kell. Mindkét függvény a táblanevet is megköveteli a mezőnevek előtt!

Az ALLSELECTED és az ALLNOBLANKROW függvények tábla-argumentumos változatai, az egyetlen argumentumukkal meghatározott tábla rekordjait adják eredményül, az ALLSELECTED a szűrők figyelembe vételével, az ALLNOBLANKROW a szűrők figyelmen kívül hagyásával. A két függvény bontatlan statisztikai értéket jelenít meg, minden nem üres kategoriában. Az ALLSELECTED argumentum nélkül is használható a CALCULATE függvénybe ágyazottan.



176. ábra

az ALLSELECTED és ALLNOBLANKROW függvények tábla-argumentumos változatai kimutatásban

A képen látható kimutatás a 171. ábra táblája alapján készült. A tábla a "római" mező "I." bejegyzéseire lett szűrve. Az "ALLS" képletében az ALLSELECTED argumentum nélkül áll a CALCULATE függvénybe ágyazottan: CALCULATE(COUNT([AZ]); ALLSELECTED()). A másik egyéni összesítés, "ANBR", képlete az ALLNOBLANKROW tábla-argumentumos változatát tartalmazza: CALCULATE(COUNT([AZ]); ALLNOBLANKROW(A)).

Végül lássunk egy példát a szűrőkezelés gyakorlati alkalmazására! A "bevételek" tábla egy vállalat alkalmazottainak árbevételeit rögzíti, a telephely és az osztály feltüntetésével.

	bevétel AZ 🔽	név 🔽	telephely 🔽	osztály 🔽	bevétel 🔽
1	1	Vitéz L.	Vác	A	39 000 Ft
2	2	Pécsi S.	Göd	В	100 000 Ft
3	3	Hajós J.	Göd	С	101 000 Ft
4	-	Witéz L.	Vác	A	129.000 Ft

177. ábra a "bevételek" tábla részlete

Kimutatással elemezni szándékozom az egyes munkatársak teljesítményét. Meg akarom állapítani [1] a kollégák összesített árbevételének arányát a szűréssel kiválasztott munkatársak összesített bevételéhez viszonyítva, [2] valamint a vállalat összesített bevételéhez viszonyítva. Három mezőhöz kívánok szűrési lehetőséget biztosítani: "név", "telephely" és "osztály". A "név" mezőt a kimutatás sorterületére szánom, tehát a szűrése a kimutatás eszközeivel történik, a másik két mező szűrését külső szűrővel oldom meg.

Az árbevételek arányának megállapításához két egyéni összesítést hozok létre. Az első képlete a következő lesz: a vizsgált munkatárs összes bevétele osztva a megjelenített kollégák összesített bevételeinek összegével, figyelembe véve a "telephely", az "osztály" mező szűrőit. A "név" mező esetleges szűrőjével egyenlőre ne foglalkozzunk.

A másik egyéni összesítés képletének megszerkesztése nem fog problémát jelenteni: a vizsgált munkatárs összesített bevételét el kell osztani a vállalat teljes bevételével. A vállalati szintű összesítést nem befolyásolhatják a szűrők! Lássuk a kimutatást és a két egyéni összesítés képletét!



178. ábra az elkészített kimutatás

Pillanatnyilag a kimutatás az érdi A osztály munkatársait mutatja. Vizsgáljuk meg Török Ferenc teljesítményét. "marány": ha a megjelenített dolgozók összesített bevételét 100%-nak vesszük, akkor Török Ferenc bevételei ennek 25,8%-t adják. "tarány": a vállalat teljes bevételének 9,3%-t Török Ferenc bevételei teszik ki, és a vizsgált időszak árbevételeinek 36%-t az érdi A osztály adja.

Amikor e két tulajdonságtól függetlenül, csak meghatározott munkatársakat jelenítünk meg a "név" mező szűrésével, akkor a "marány" értéke megegyezik a "tarány" értékével. Ha a problémát az ALLEXCEPT függvény kiegészítésével próbáljuk orvosolni, tehát a "név" mezőt is felsoroljuk a függvény argumentumaként, akkor az egyéni összesítés minden értéke 100%-ra módosul.

név 🖵	marány	tarány	név	,Т	marány	tarány
Benkő P.	5,0 %	5,0 %	Benkő P.		100,0 %	5,0 %
Győri H.	10,7 %	10,7 %	Győri H.		100,0 %	10,7 %
Hajós J.	8,1 %	8,1 %	Hajós J.		100,0 %	8,1 %
Végösszeg	23,9 %	23,9 %	Végösszeg		100,0 %	23,9 %

179. ábra a név szerinti szűrés eredménye és az ALLEXCEPT függvény kiegészítésének hatása

A megoldás az ALLSELECTED függvény, amely az egyetlen argumentumával meghatározott mező minden megjelenített bejegyzését adja eredményül a közvetlen szűrők figyelembe vételével, de a kimutatás szerkezetétől függetlenül. A "marány" egyéni összesítés végleges képlete: SUM([bevétel]) / CALCULATE(SUM([bevétel]) ; ALLSELECTED('bevételek'[név])).

szűrő-információk

A DAX szűrő-információs függvényei két kérdést tudnak megválaszolni: szűrt-e a vizsgált mező, illetve, ha szűrt, akkor a szelektív megjelenítés egyetlen egyedi bejegyzést eredményezett-e. A válasz, azaz a függvények eredménye, logikai adattípusú.

Az ISFILTERED és az ISCROSSFILTERED függvények akkor adnak IGAZ eredményt, ha az argumentumukkal deklarált mező megjelenítését szűrők befolyásolják. Az ISFILTERED csak a közvetlen szűrőt ismeri fel, az ISCROSSFILTERED a közvetett szűrőket is jelzi. Az ISCROSSFILTERED leírásában szerepel még a kapcsolódó tábla szűrőjének felismerése is, de erről nem tudtam meggyőződni, mert az én gépemen, saját tesztfeladataimban ez a funkció nem működött.

A függvényeket egy vállalat dolgozóinak adatait tartalmazó táblán mutatom be. A tábla neve "munkatársak". Két egyéni összesítést hoztam létre a létszám megállapítására.

1	név 🔽	született 🛛 💽	telephely 🔽	osztály 🗾	státusz 🏼 🖅	angol				
1	Abonyi Emőke	1973. 03. 16.	Ócsa	pénzügy	beosztott					
2	Agócs Aranka	1970. 12. 03.	Paks	tervezés	beosztott					
3	Alföldi Adalbert	1957. 04. 05.	Baja	pénzügy	beosztott					
4	Alföldi Katalin	1977. 11. 02.	Ócsa	szállítás	beosztott	középfokon				
5	Almási Jolán	1950. 12. 22.	Vác	szállítás	beosztott	/				
6	Arató Pál	1959. 06. 04.	Ócsa	szállítás	beosztott					
Γ	létsz ISF: (üres)									
	létsz ISCF: 481									

Iétsz ISF =IF(ISFILTERED([telephely]) ; COUNTA([név])) Iétsz ISCF =IF(ISCROSSFILTERED([telephely]) ; COUNTA([név]))

A kép tanúsága szerint a megjelenített rekordok körét a "státusz" mező szűrője határozza meg. A "létsz ISF" egyéni összesítés képletében álló ISFILTERED függvény eredménye HAMIS, mert a "telephely" mező nincs szűrve, ezért az IF függvény üres eredményt ad. A "létsz ISCF" egyéni összesítés képletében az IF függvény második argumentumában álló művelet lesz végrehajtva, mert az ISCROSSFILTERED függvény a "telephely" mező közvetett szűrése miatt, IGAZ értéket ad.

A függvényeket kimutatásba helyezve furcsa jelenséggel találkozunk. Hozzunk létre két egyéni összesítést! "ISF":=ISFILTERED([telephely]) és "ISCF":= ISCROSSFILTERED([telephely]). Amenynyiben a "telephely" mező a kimutatás sor- vagy oszlopterületén áll, akkor mindkét egyéni összesítés IGAZ értékeket ad a kimutatás minden cellájában, ha van szűrő beállítva a "telephely" mezőhöz, ha nincs. Ha eltávolítjuk a "telephely" mezőt a sor illetve az oszlopterületről, akkor már csak az "ISCF" ad IGAZ eredményt, az "ISF" már minden kategoriában HAMIS. A jelenség magyarázata a kimutatás lényegéből fakad. A Pivot-tábla a sor- és oszlopmező tételei szerinti csoportosításban vizsgálja a statisztikai mező adatait. Egy-egy csoport statisztikai értékének megállapításához a rekordokat szűrni kell a sor- és oszlopmező meghatározott tételeivel. Tehát a kimutatás struktúrája egy szűrési metódust határoz meg a statisztikai számításhoz.

státusz 🔻	Értékek	osztály 💌 pénzügy	szállítás	tervezés
beosztott	ISF	HAMIS	HAMIS	HAMIS
	ISCF	IGAZ	IGAZ	IGAZ
vezető	ISF	HAMIS	HAMIS	HAMIS
	ISCF	IGAZ	IGAZ	IGAZ
felsővezető	ISF	HAMIS	HAMIS	HAMIS
	ISCF	IGAZ	IGAZ	IGAZ

181. ábra az ISFILTERED és az ISCROSSFILTERED függvények kimutatásban

^{180.} ábra az ISFILTERED és a ISCROSSFILTERED függvények adatnézetben

A képen látható kimutatás-elrendezésben tehát a "státusz" és az "osztály" mező csoportosítja, másként fogalmazva, közvetett módon szűri a "telephely" mezőt. Ezt a hatást az ALL függvénnyel semlegesíthetjük az "ISCF" képletében: =CALCULATE(ISCROSSFILTERED([telephely]); ALL('munkatársak'[státusz]; 'munkatársak'[osztály])).

A kimutatás-szerkezet szűrő hatásán felül a közvetlen szűrés hatását is kikapcsolhatjuk, ha az ALL függvény argumentumai közé felvesszük a "telephely" mezőt is, így az "ISCF" csak a közvetett szűrést fogja jelezni IGAZ eredményével.

Nézzünk egy példát az ISFILTERED függvény kimutatásbeli alkalmazásának szemléltetésére. Maradjunk a "munkatársak" tábla vizsgálatánál. A nemzetközi kapcsolatokkal is rendelkező vállalatnál fontos szempont az alkalmazottak angol nyelvtudása. Az "angol" mező bejegyzései erről adnak információt. Kimutatással elemezni szeretném a munkatársak angol nyelvtudását. Kíváncsi vagyok, hányan és milyen szinten beszélik az angolt az egyes termelési egységekben illetve az azonos státuszban dolgozók csoportjaiban.

A kimutatás sor- és oszlopmezői a "telephely" és az "osztály" mező lesznek. Két külső szűrőt hozok létre az "angol" és a "státusz" mezőhöz. Ha az "angol" mező nincs szűrt állapotban, akkor a létszám adatokat szeretném megjeleníteni a kimutatás struktúrájának és a beállított szűrőknek megfelelően. Ha az "angol" mező szűrt állapotban van, akkor látni szeretném a csoport létszámát és az angolul beszélők létszámát is. Valahogy így:



182. ábra a példa kimutatása szűrés nélkül és szűrt állapotban

A felső kimutatás és a bal oldalon álló elemei a szűretlen állapotot mutatják. Ezek szerint a bajai telephely legnépesebb osztálya hatvanöt főt foglalkoztat, ami a vállalati létszámnak több mint a tíz százaléka. Az alsó kimutatás és a jobb oldalon álló elemei már a szűrt állapotot mutatják. A termelési egységek beosztott státuszban dolgozó munkatársai közül azokat számláltattam meg, akik legalább középfokon beszélik az angolt. Ezek szerint a bajai szállítási osztályon dolgozó ötvenkilenc beosztott kolléga egyharmada legalább középfokon beszéli az idegen nyelvet.

Az egyéni összesítés statisztikai vizsgálatát tehát az "angol" mező közvetlen szűrője határozza meg. Amennyiben a mezőnek nincs közvetlen szűrője, akkor a létszám-adatokat kell megjelenítenie, ha az "angol" mező szelektív megjelenítését közvetlen szűrés határozza meg, akkor az angolul tudok számát kell megállapítania, a kimutatás struktúrájának és a szűrőinek megfelelően.



183. ábra az "angol EÖ" képlete

A FORMAT függvénnyel az angolul beszélők számának megjelenítését szabályoztam. A függvény ebben a képletben a program SZÖVEG függvényével azonos funkciót lát el: az első argumentumával

meghatározott számot a második argumentumával deklarált formában írja ki. A mi esetünkben mindig két számjeggyel, a szép megjelenítés érdekében.

A másik kérdésre, azaz a szűrés egyetlen egyedi bejegyzést eredményezett-e, a HASONEFILTER és HASONEVALUE függvények adnak választ. A HASONEFILTER csak a közvetlen, a HASONEVALUE a közvetlen és közvetettet szűrőket is figyelembe veszi. A függvények eredménye azonos a COUNT-ROWS(FILTERS(<mezőnév>) = 1 és a COUNTROWS(VALUES(<mezőnév>) = 1 kifejezések eredményével.

A két függvényt egy három-mezős táblán, melynek neve "A", mutatom be. Három egyéni összesítést készítettem. Mindhárom a "betűk" mezőt vizsgálja.



184. ábra a HASONEFILTER és a HASONEVALUE függvények bemutatása

A kép bal oldalán a "betűk" mezőt szűrtem a "B" betűre. A megjelenített két B betűs rekord ellenére a HASONEFILTER függvény eredménye IGAZ, mert a közvetlen szűrés egyetlen egyedi bejegyzést eredményezett a B betűt. Az ábra jobb oldalán a "számok" mezőhöz állítottam be megjelenítési feltételt: Csak a hármas számot tartalmazó rekordokat mutasd! A HASONEFILTER függvény most HAMIS eredményt ad, annak ellenére, hogy a "betűk" mező csak egyetlen egyedi bejegyzést mutat, az E betűt. A függvény ebben az esetben is ugyanarra a kérdésre válaszol, mint az előbbi helyzetben: A "betűk" mező közvetlen szűrése egyetlen egyedi bejegyzést eredményez? Mivel nincs közvetlen szűrés, ezért a válasz HAMIS.

Nézzünk egy egyszerű példát a HASONEVALUE függvény alkalmazására. Egy budapesti vállalat kerületi egységeinek bevételeit a "bevételek" táblában tárolom. Egy "segéd" nevű táblát is létrehoztam, amelynek segítségével a összesített bevétek számformátumát tudom majd egyszerűen beállítani a kimutatásban. A két tábla között nincs kapcsolat.

i	bevétel AZ 🛛 🔽	kerület 🔽	kerület i 🔽	bevétel 🔽
1	001	V. kerület	1	25 173 400 Ft
2	002	V. kerület	1	20 385 400 Ft
3	003	X. kerület	4	20 530 000 Ft
4	004	IX. kerület		16 351 500 Et

185. ábra a két tábla adatnézetben

A bevételeket, kerületenkénti bontásban, kimutatással fogom összesíteni. Mivel a bevételek már a rekordokban is milliós nagyságrendűek, ezért a kimutatásban egy a "pénznem" mezőn alapuló külső szűrővel, lehetővé teszem az ezerforintos és a millióforintos megjelenítést is.

kerület	-	ÖSSZ	pénznem 🌾
V. ke	rület	397 785 eFt	
VII. ke	rület	341 267 eFt	Ft
IX. ke	rület	269 098 eFt	
X. ke	rület	395 692 eFt	eFt
XI. ke	rület	254 765 eFt	
XV. ke	rület	287 667 eFt	mFt
Végösszeg		1 946 275 eFt	

186. ábra a kimutatás a "pénznem" mező külső szűrőjével

A vállalatnak hat kerületben vannak részlegei, ezek összesített bevételeit láthatjuk a képen, ezerforintos megjelenítésben. Az ezer és a millió Forintos megjelenítés egy osztáson alapul. Az összesített bevételt elosztjuk a kiválasztott pénznem "szám" mezőjében álló bejegyzéssel. A képlet kiértékelése azonban hibát eredményez, ha szűréssel nem választunk pénznemet, ezért a végrehajtást feltételhez kötöttük. Az "osztás" egyéni összesítés képlete: =IF(HASONEVALUE('segéd'[szám]); SUM([bevétel]) / VALUES('segéd'[szám])). Az "össz" nevű egyéni összesítés képletében ellenőriztem a szűrés eredményét, előírtam a számok csoportosításának módját és meghatároztam a megjelenítendő pénznem jelölőt.



187. ábra az "össz EÖ" egyéni összesítés képlete

szűrőfüggvények

A DAX szűrőfüggvényeivel egy táblából válogathatjuk ki azokat a rekordokat, amelyeket a bővítménynek a gazda-képlet kiértékelésekor figyelembe kell vennie. A szelektálás a felhasználó által szerkesztett feltételeken alapul. A szűrő-kifejezéseket az alkalmazott függvény feltétel-argumentumaival kell deklarálnunk.

Kezdjük a nevében is szűrő FILTER-rel, amely már többször előfordult a példáinkban. Elemezzünk egy kéttáblás adatbázist: "munkahelyek" és "munkatársak". Kimutatással vizsgálni szeretném a vállalatnál dolgozó, fiatal nők helyzetét.





Mivel a munkatársak táblában nincs rögzítve sem a munkatársak neme, sem születési dátuma, ezért a kollégák nemét és életkort a "személyi szám" mező segítségével állapítottam meg. Az "fnők" egyéni összesítés képlete két beágyazott FILTER függvényt tartalmaz: a külső leválogatja az 1979 után születetteket, azokból a női rekordokból, amelyeket az első argumentumában álló, belső FIL-TER függvény választ ki a munkatársak táblából. Az így kapott rekordokat a COUNTROWS függvénynyel számoltatjuk meg.



189. ábra a feladat megoldása egymásba ágyazott FILTER függvényekkel

A FILTER függvény egyetlen szűrő-argumentumában több feltételt is megadhatunk logikai operátorok használatával. =COUNTROWS(FILTER('munkatársak'; 'munkatársak'[személyi szám] > 2000000000 && MOD(TRUNC('munkatársak'[személyi szám] / 100000000); 100) > 79))

> 190. ábra ÉS operátor a FILTER szűrő-argumentumában

A képen az ÉS logikai operátort kék betűszínnel emeltem ki. A fenti módszerrel egyetlen FILTER függvénnyel is eredményre jutottunk. Most vegyünk egy összetett feltétel-rendszert tartalmazó feladatot. Állapítsuk meg a bajai és a váci telephely dolgozóinak számát, de csak azokat a kollégákat vegyük számba, akinek legalább kilenc bónusz-pontjuk van!

> =COUNTROWS(FILTER('munkatársak' ; RELATED(munkahelyek[város]) = "baja" || RELATED(munkahelyek[város]) = "vác" && 'munkatársak'[bónusz] > 8))

> > 191. ábra logikai operátorok a FILTER szűrő-argumentumában

Adatnézetben a teljes táblára vonatkozóan az egyéni összesítés eredménye száznegyvennyolc. Ahhoz képest, hogy a munkatársak táblában összesen tíz rekordban találunk kilences vagy tízes bejegyzést, ez kissé soknak tűnik. A jelenség magyarázata: a DAX-ban a két logikai operátor nem egyenrangú, tehát a kiértékelés nem balról jobbra halad, hanem először az ÉS azután a VAGY kifejezések lesznek kiértékelve. Természetesen zárójelek vagy az IN operátor használatával már a helyes eredményre jutunk.

> =COUNTROWS(FILTER('munkatársak'; (RELATED(munkahelyek[város]) = "baja" || RELATED(munkahelyek[város]) = "vác") && 'munkatársak'[bónusz] > 8)) =COUNTROWS(FILTER('munkatársak'; RELATED(munkahelyek[város]) IN { "baja"; "vác" } && 'munkatársak'[bónusz] > 8))



A felső képletben zárójelekkel, az alsóban az IN operatorral kényszerítettük ki a helyes kiértékelési sorrendet. Ha az IN operátor listája a szokásos, és nem kapcsos zárójelek között áll, akkor a #HIBA karakterláncot kapjuk eredményül. A hiba okaként ezt az üzenetet olvashatjuk: «Szemantikai hiba: A következő operátor vagy kifejezés ebben a környezetben nem támogatott: "()".» Természetesen ez nem szemantikai, hanem szintaktikai hiba.

Ne felejtsük el, hogy a DAX-nak nincs NOT IN operátora, tehát az IN tagadása közvetve történik. Például, hányan dolgoznak a vállalatnál a bajai és a váci telephely munkatársai kivételével?

> =COUNTROWS(FILTER('munkatársak' ; NOT RELATED(munkahelyek[város]) IN { "baja" ; "vác" }))

> > 193. ábra az IN operátor listájának tagadása

A feltétel deklarálása a CONTAINSROW függvénnyel is történhet. Hányan dolgoznak az "előkészítés"-ben illetve a "gyártás"-ban. Ez a két terület a munkahelyek tábla részleg mezőjének egy-egy tétele.

> =COUNTROWS(FILTER('munkatársak' ; CONTAINSROW({ "előkészítés" ; "gyártás" } ; RELATED(munkahelyek[részleg]))))

> > 194. ábra VAGY logikai viszony a CONTAINSROW függvényben

A függvény első argumentuma kapcsos zárójelek között, pontosvesszővel elválasztva tartalmazza a második argumentumában álló mező számításba veendő tételeit. A kapcsos zárójelek itt is a VAGY logikai viszony alkalmazását jelentik. De a függvény kezelni tudja a logikai ÉS viszonyt is. Lássunk erre is egy példát.

=COUNTROWS(FILTER('munkatársak' ; CONTAINSROW({ ("baja" ; "előkészítés" ; 7) ; ("vác " ; "gyártás" ; 6) } ; RELATED(munkahelyek[város]) ; RELATED(munkahelyek[részleg]) ; 'munkatársak'[bónusz])))

> 195. ábra ÉS és VAGY logikai viszony a CONTAINSROW függvényben

Ezzel a képlettel a hét bónusz-pontos bajai előkészítők és a hatpontos váci gyártók létszámát kapjuk meg. A CONTAINSROW függvény első argumentuma ebben a képletben is a számításba veendő tételeket tartalmazza kapcsos zárójelek között, de most két zárójeles csoportban. A két zárójeles egységet és a zárójelek között álló tételeket is pontosvessző választja el. A két zárójeles egység VAGY, a zárójeleken belül álló tételek ÉS logikai viszonyban vannak egymással. A függvény további argumentumai a szűrt mezők, a zárójelek között álló tételekkel azonos sorrendben.

 $\begin{aligned} & \quad \text{CONTAINSROW}(\\ \{(\text{tétel}_1; \text{tétel}_A; \text{tétel}_1; \dots); (\text{tétel}_2; \text{tétel}_B; \text{tétel}_1; \dots); (\text{tétel}_3; \text{tétel}_C; \text{tétel}_{III}; \dots) \dots \};\\ & \quad \text{mez}\tilde{o}_1; \text{mez}\tilde{o}_2; \text{mez}\tilde{o}_3; \dots)\end{aligned}$

196. ábra a CONTAINROW függvény argumentumai

Figyelem, figyelem! A CONTAINSROW függvény leírása a névkiegészítőben, szerkesztés közben megjelenített argumentumlistája és hibaüzenetei hibásak!

A CALCULATE függvénnyel már találkoztunk. Működésének lényege: az első argumentumával meghatározott műveletet a további argumentumaival deklarált szűrőkkel kiválasztott rekordokban végzi el. Az eredeti, fiatal nők elemzésére létrehozott egyéni összesítést ezzel a függvénnyel is megfogalmazhatjuk.



197. ábra a fiatal nők számának meghatározása a CALCULATE függvénnyel

Vegyük sorra a CALCULATE függvény legfontosabb tulajdonságait: [1] első argumentuma kiértékelésének egyetlen adatot kell eredményeznie, [2] további argumentumai szűrőfeltételek, vagy [3] táblanevek, vagy [4] egy táblát eredményező kifejezés, [5] amennyiben a függvény a kapcsoló mezők azonos bejegyzéseit használja "szűrőként", akkor a második argumentum elhagyható, [6] a szűrő-argumentumok logikai ÉS viszonyban állnak egymással, [7] egy argumentum több, ugyanarra a mezőre vonatkozó szűrőt is tartalmazhat logikai operátorokkal összekapcsolva.

Nézzünk egy egyszerű példát az ötödik tulajdonság bemutatására: mikor hagyható el a CALCU-LATE függvény második argumentuma? Van két táblánk: "T1" és "T2".

4	T1 AZ 💽	betűk 🚡 💌	ÖSSZ1 🛛 💌	ÖSSZ2 🛛 💌
1	1	Α	15	15
2	2	В	10	10

1	T2 AZ 🗾 🔽	betűk 🦷	🛛 🗹 🖬	mok 🔽
1	1	Α		7
2	2	Α		3
3	3	В		6
4	4	В		4
5	5	Α		5

198. ábra a két táblát a "betűk" mezők azonos értékei kapcsolják össze

A "T1" tábla ÖSSZ1 számított mezőjének képlete az eddigi ismereteink szerint így alakulna: CALCU-LATE(SUM(T2[számok]); T2[betűk] = EARLIER([betűk])). A CALCULATE függvény szűrő-argumentuma ebben a képletben a kapcsoló mezők azonos bejegyzéseit deklarálja. Ha ez így van, akkor a CALCULATE függvény második argumentuma elhagyható. ÖSSZ2: CALCULATE(SUM(T2[számok])).

A hetedik tulajdonság szerint egy szűrő-argumentumban több, ugyanarra a mezőre vonatkozó, logikai operátorokkal összekapcsolt feltétel is állhat. Nézzünk erre egy példát: a "vevők" táblában meg szeretnénk állapítani a dunántúli kereskedők számát.

1	vevő AZ 🛛 🔽	vevő neve 🔽	vevőtípus 🔽	régió 🗾
1	1	Roboz Zoltán	magán	KM
2	2	Szirtes Edvin	vendéglős	KM
3	3	Rédei Matild	kiskereskedő	КМ
4	4	Szőnyi Lujza	kiskereskedő	КМ
5	5	Gál Szilárd	nagykereskodő	KM

199. ábra a példa táblája

Az egyéni összesítés képletében a CALCULATE függvény szűrő-argumentumaival ki kell választanunk a Nyugat-, a Közép- és a Dél-Dunántúl régiók kis- és nagykereskedőit, majd a függvény kifejezésargumentumával meg kell számláltatni a kiválasztott rekordok "vevő AZ" mezőbejegyzéseit.



200. ábra a dunántúli kereskedőket elemző egyéni összesítés képlete

A CALCULATE függvény tulajdonságainak felsorolásakor harmadikként említett táblanevekkel a számításkor figyelembe veendő kapcsolatokat deklaráltuk. Ha a számításban érintett táblák között több kapcsolat is létezik, akkor az USERELATIONSHIP függvénnyel tudjuk az inaktív kapcsolatot a képlet kiértékelésének idejére aktiválni. A függvény argumentumai a kapcsoló mezők.

A függvény működését egy kéttáblás adatbázison mutatom be. Az "eladások" és a "naptár" táblák között két kapcsolatot hoztam létre.



201. ábra a példa táblái kapcsolatnézetben

Az adatbázis aktív kapcsolata az 'naptár'[dátum] - 'eladások'[megrendelés], inaktív kapcsolata pedig a 'naptár'[dátum] - 'eladások'[kiegyenlítve].

		eladás AZ	megrendelés	szállítás	kiegyenlítve	darab
		1	2015.01.04	2015.01.07	2015.01.14	110
hónap név 💌	Osszeg - darab	dbk	2015.01.15	2015.01.22	2015.01.29	60
január	675	455	2015.01.18	2015.01.23	2015.01.27	285
február	545	220	2015.01.27	2015.02.06	2015.02.14	220
március	1010	990	2015.02.22	2015.03.07	2015.03.14	240
április	380	565	2015.02.26	2015.03.02	2015.03.13	305
május	630	1010	2015.03.04	2015.03.13	2015.03.22	210
június	290	290	2015.03.13	2015.03.17	2015.03.28	235
Végösszeg	3530	3530	2015.03.18	2015.03.31	2015.04.09	305
		10	2015.03.21	2015.03.25	2015.04.02	205
		11	2015-02-25	2015-04.02	2015.04.17	55

202. ábra a tábla forrása a kimutatással

A kimutatás az eladások darabszámát mutatja havi bontásban. Az automatikus összesítés az aktív kapcsolat "megrendelés" mezője alapján történik, míg a "dbk" az inaktív kapcsolat "kiegyenlítve" mező bejegyzései alapján csoportosít. Az egyéni összesítés képlete: =CALCULATE(SUM([darab]) ; USERELATIONSHIP('eladások'[kiegyenlítve] ; 'naptár'[dátum])). Tehát a USERELATIONSHIP függvény az 'eladások'[kiegyenlítve] - 'naptár'[dátum] kapcsolat használatát írja elő a CALCULATE függvény számára.

A CALCULATE függvénynek van egy, eddig még nem említett tulajdonsága: a szűrőargumentumában szereplő mezők szűrhetőségüket és csoportosító szerepüket is elvesztik a kimutatásban. Lássunk erre a tulajdonságra is egy egyszerű példát!



203. ábra a "karakterek" tábla forrása és a tábla elemzésére létrehozott kimutatások

A felső kimutatás "CAL" egyéni összesítése a "B" és az "D" betűs rekordok "szám" mezőjének bejegyzéseit adja össze. A bővítmény betűk szerint nem bontja a megállapított statisztikai értéket csak a "római" mező tételei szerint. A "betűk" mező a kimutatásban szűrhető marad, de látható tételeinek statisztikai értékét ez a művelet nem befolyásolja.

A szokásos megjelenítést és működést a KEEPFILTERS függvénnyel állíthatjuk elő. Ezt a megoldást látjuk az alsó kimutatás "CKEEP" képletében. A KEEPFILTERS függvény argumentuma a CAL-CULATE függvény egy vagy több szűrő-argumentuma. A függvény "helyreállítja" az argumentumaiban szereplő mezők szűrhetőségét is, a szűrt rekordok halmazában. Tehát a példánál maradva, szűréssel választhatunk a "B" vagy "D" rekordok között.

Az ALLEXCEPT függvénnyel azokat a mezőket deklarálhatjuk, amelyek szűrőit a képlet kiértékelésekor figyelembe kell venni. A CALCULATE függvény szűrő-argumentumában azonban a függvény szűrést generál: az argumentumaival meghatározott mezők, a mezők aktuális értékeikkel lesznek szűrve. Lássunk egy egyszerű példát!

J.	AZ 🔽	betűk 🔽	logikai 💌	ALLE 💌
1	01	С	IGAZ	2
2	02	В	HAMIS	3
3	03	В	HAMIS	3
4	04	В	HAMIS	3
5	05	C	IGAZ	2
6	06	Α	HAMIS	2
7	07	C	HAMIS	1
8	08	В	IGAZ	1
9	09	Α	HAMIS	2

1	AZ 🔽	betűk 🖃	logikai 💽	ALLE 🔽
1	06	Α	HAMIS	2
2	09	Α	HAMIS	2
3	02	В	HAMIS	3
4	03	В	HAMIS	3
5	04	В	HAMIS	3
6	08	В	IGAZ	1
7	01	С	IGAZ	2
8	05	С	IGAZ	2
9	07	С	HAMIS	1

ALLE =CALCULATE(COUNTA([AZ]) ; ALLEXCEPT(A ; A[betűk] ; A[logikai]))

204. ábra

a példa kilenc rekordos táblája (A) rendezetlenül és a "betűk" mező bejegyzései alapján rendezetten

A rendezett táblában jól megfigyelhető az ALLEXCEPT függvény hatása az ALLE számított mező képletében: hány olyan rekord található a táblában, amelynek betűje és logikai értéke azonos az aktuális rekord betűjével és logikai értékével. Ha például a "o2" kulcsú rekord "betűk" és "logikai" mezőjében álló bejegyzések alapján szűrnénk a táblát, akkor három rekordot kapnánk.

A számított mező képletében álló ALLEXCEPT(A ; A[betűk] ; A[logikai]) képletrész egyenértékű két, egymásba ágyazott, FILTER függvénnyel: FILTER(FILTER(A ; [betűk] = EARLIER([betűk])) ; [logikai] = EARLIER([logikai])).

A CROSSFILTER függvény szűrő-kezelés eszköze több a többhöz viszonyban álló táblák egyéni összesítéseiben. Csak beágyazottan alkalmazható, ezért itt, és nem a funkciója szerinti fejezetben mutatom be. Első két argumentuma az adatbázis egy kapcsolatának kapcsoló mezői, több oldali kapcsoló mező és egy oldali kapcsoló mező sorrendben. Harmadik argumentumával, a bővítmény szóhasználatával, a "kereszt-szűrés irányát" határozhatjuk meg. Lehetséges értékei a következők. Both: a megadott kapcsolat mindkét táblájában beállított szűrők befolyásolják a képlet eredményét. OneWay: csak a kapcsolat egy oldalán álló tábla szűrői lesznek figyelembe véve a képlet kiértékelésekor. None: egyik tábla szűrője se befolyásolja a képlet eredményét.

Lássunk egy egyszerű modellt a függvény bemutatására: színházak adatbázis, négy táblával.



205. ábra a CROSSFILTER függvény működését bemutató modell

Ahogy azt a képen látjuk, a "színészek" és a "színdarabok" táblák több a többhöz viszonyban állnak egymással. Kapcsolótáblájuk a "szerepek" tábla. Az "előadások" tábla egy hónap előadásait rögzíti. A kép közepén álló kimutatással statisztikát készítettem az egyes színészek foglalkoztatásáról. A pivot-táblához két külső szűrő tartozik, az egyik a "szerepek" tábla "énekes" mezőjének, a másik a "színdarabok" tábla "cím" mezőjének tételeivel. A kimutatást a kép szűretlen állapotban mutatja.

A "hányat játszik" egyéni összesít képlete, az eddigi ismereteink szerint: CALCULATE(COUNTA('előadások'[előadások AZ]); szerepek). A képlet kiértékelésekor a bővítmény mindkét szűrő állapotát figyelembe veszi.

A kimutatásban közvetlenül két táblát szerepeltettünk: a "színészek" táblát, amelynek mezője a sorterületen áll és az "előadások" táblát, amelynek egyik mezőjét elemezzük. A CROSSFILTER függvénnyel csak a másik két tábla szűrőinek hatását szabályozhatjuk: CALCULATE(COUNTA('előadások'[előadások AZ]); CROSSFILTER(szerepek[színdarab AZ]; 'színdarabok'[színdarab AZ]; Both)).



206. ábra a CROSSFILTER függvény alkalmazásának sémája

A módosított és az eredeti képlet azonos statisztikai értéket eredményez. Mindkét szűrő hatása érvényesül. A függvény harmadik argumentumának OneWay értéke csak a "színdarabok" tábla, None értékénél egyik tábla szűrői sem lesznek figyelve.

A CALCULATE függvény első argumentumában csak egyetlen adatot eredményező kifejezés állhat. Ha táblát adó kifejezést szeretnénk definiálni szűrt vagy többtáblás környezetben, akkor a CAL-CULATETABLE függvényt kell használnunk. A függvény argumentumlistája azonos a CALCULATE függvény argumentumlistájával. Az alábbi képen a CALCULATETABLE függvény alkalmazására látunk egy példát.

	vásárlás AZ 💌	vevő AZ 🛛 🔽	dátum 🔽	fajta 🗾	évjárat 🗾	karton 🗾	nettó ár 🛛 🔽
1	001	383	2016. 01. 03.	Chardonnay	2010	1	1 500 Ft
2	002	351	2016. 01. 03.	Királylányka	2010	1	2 940 Ft
3	003	075	2016. 01. 03.	Királylányka	2008	1	2 940 Ft
4	004	065	2016. 01. 03.	Olaszrizling	2012	2	3 240 Ft
5	005	067	2016-01.05.	Szürkebarát	2009	1	3 000 Ft

=COUNTROWS(CALCULATETABLE(DISTINCT('vásárlások'[vevő AZ]) ; MONTH('vásárlások'[dátum]) = 5))

207. ábra

a CALCULATETABLE függvény a "vásárlások" tábla májusi vevőit elemző egyéni összesítés képletében

A bemutatott három szűrőfüggvény közül tehát a CALCULATE egyetlen adatot ad eredményül, a CALCULATETABLE egymezős táblát, míg a FILTER táblát eredményez. A két CALCULATE függvény szűrőfunkciója az első argumentumukkal deklarált kifejezés hatókörének, konkrétan a feldolgozandó adatok halmazának meghatározására szolgál, míg a FILTER függvény egy származtatott vagy egy létező tábla rekordjainak szelektálására használható. A FILTER függvény utóbbi funkciója azonban már átvezet bennünket a kereső függvények csoportjába.

kereső függvények

A DAX kereső függvényei az argumentumaikkal meghatározott mezőbejegyzést illetve mezőbejegyzéseket adják eredményül. A LOOKUPVALUE függvény az első argumentumával megadott mező bejegyzését adja eredményül, abból a rekordból, amelyet további mezőnév-bejegyzés argumentum-párosai határoznak meg. Ez egyértelmű megfogalmazás, de mi lesz a függvény eredménye, [1] ha nincs a feltételeknek megfelelő rekord, [2] ha több ilyen rekord is található, azonos eredmény-mező bejegyzéssel és [3] ha több ilyen rekord is létezik, különböző eredmény-mező bejegyzéssel? Próbáljuk ki!

1	AZ		betűk 🗾	számok 💽	logikai 💽	római 🔽					
1		1	С	2	HAMIS	Ш.					
2		2	A	5	IGAZ	П.					
3		3	E	1	IGAZ	П.					
4		4	A	3	IGAZ	П.					
5		5	E	1	HAMIS	Ш.					
	LOOK1:	(üres)	LOOK2: II.	LOOK3: #HIBA 4							
		~									
l	LOOK1 =LOOKUPVALUE([római] · adatok[betűk] · "C" · adatok[számok] · 6)										
L	LOOK2 =LOOKUPVALUE([római] ; adatok[betűk] ; "A" ; adatok[logikai] ; TRUE)										
L	LOOK3 =LOOKUPVALUE([római] ; adatok[betűk] ; "E" ; adatok[számok] ; 1)										

208. ábra a LOOKUPVALUE függvény eredménye speciális esetekben

A tábla három egyéni összesítése válaszol a feltett kérdéseinkre. Hatos számot tartalmazó "C" betűs rekord nincs a táblában, ezért a "LOOK1" üres bejegyzést eredményezett. Van-e a táblában "A" betűs, IGAZ bejegyzésű rekord és ha van, mi a bejegyzése a "római" mezőben? – kérdeztem a "LOOK2" egyéni összesítéssel. A válasz, azaz a függvény eredménye egy római kettes. Tehát van ilyen rekord, nem tudjuk hány darab, de az biztos, hogy a "római" mezőben mindegyiknek kettes a bejegyzése. A "LOOK3" egyéni összesítéssel az egyes számot tartalmazó, "E" betűs rekordok "római" mezőjében álló bejegyzésére voltam kíváncsi. A függvény talált ilyen rekordokat, de mivel az eredmény-mezőben különböző bejegyzések álltak, hibát jelzett. Az üzenet szövege azonban értelmetlen: "Több értéket tartalmazó tábla van megadva a várt egyetlen érték helyett". Helyesen: A képlet kiértékelése több adatot eredményezett egyetlen érték helyett. A LOOKUPVALUE nem szűrőfüggvény, tehát csak konstanst fogad el a mezőnév-bejegyzés párosok második argumentumaként, relációs jelet nem használhatunk!

A leggyakoribb alkalmazásuk miatt, a bővítmény a FIRSTNONBLANK és a LASTNONBLANK függvényeket az időszak-kezelő függvények csoportjába sorolja, de funkciójuk a kereső függvényekkel rokon: az első argumentumukkal deklarált mező, a vizsgált csoportban található, első/utolsó bejegyzését adják eredményül, ahol a második argumentumukban álló kifejezés kiértékelése nem üres eredményre vezet. A függvények leírása adattípus-korlátozást nem tartalmaz!

A két függvény működését a LASTNONBLANK függvénnyel mutatom be. Adott két tábla: "raktár" és "naptár". A kapcsoló mezők a táblák "dátum" mezői. Készítenem kell egy kimutatást, amely a hónapok naptári utolsó napjára vonatkozó készletet tartalmazza.

termé	k	Ŧ	dátu	um	Ŧ	készlet	-		zár				termék	Ŧ	
Α			201	15.05	.05		31		év	Ŧ	hónap	Ŧ	Α		В
В			201	15.05	.06		37		2015		május		33		
В			201	15.05	.10		79				július				63
Α			201	15.05	.25		89								
Α			201	15.05	.31		33		LNB				termék	-	
В			201	15.06	.05		96		év	Ŧ	hónap	Ŧ	Α		В
A			201	15.06	.08		30		2015		május		2015.05	.31	2015.05.10
В			201	15.06	.15		56				június		2015.06	.08	2015.06.21
В	k		201	15.06	i.18 39					július		2015.07	.23	2015.07.31	
В			201	15.06	.21		21								
Α			201	15.07	.03		28		zárok				termék	Ŧ	
В			201	15.07	.06		93		év	-	hónap	-	Α		В
Α			201	15.07	.15		95		2015		május		33		79
Α			201	15.07	.23		79				június		30		21
В			201	15.07	.31		63				július		79		63
	Z	ár		=CA	LCU	JLATE(SU	JM([ké:	szlet]) ; L	AST	TDATE('na	apt	ár'[dátur	n]))	
	LNB =LASTNONBLANK('napta			tár'[dátur	n] ;	CALCULA	TE(SUM([ké	szl	et])))					
	zárok				OK	UPVALUE	([ké	İszl	et] ; [dátı	nw]	; [LNB])				

209. ábra a "raktár" tábla forrása és az elemzésére létrehozott kimutatások

Először a már ismertetett LASTDATE függvénnyel próbálkoztam a "zár" egyéni összesítésben, de ez nem vezetett eredményre, mert a kimutatás csak azokat a készlet-bejegyzéseket mutatja, amelyek a hónap utolsó napján lettek rögzítve.

A második kimutatás "LNB"-ben kipróbáltam, hogyan működik a LASTNONBLANK. A függvény első argumentuma tehát az eredmény-mező neve, második argumentuma pedig egy kifejezés, amelynek kiértékelése után a bővítmény megkeresi az adott dátumegységen belül azt az utolsó rekordot, ahol a kifejezés kiértékelése nem üres bejegyzés, majd ennek a rekordnak a függvény első argumentumával meghatározott mezőjében álló bejegyzést kapjuk eredményül.

A harmadik kimutatásban a "zárok" szolgáltatja a hónap utolsó napján érvényes készletet. Mindhárom egyéni összesítés a "raktár" táblában tárolódik.

A RELATED és a RELATEDTABLE kereső függvényeket már ismerjük. Foglaljuk össze használatukat a különböző táblákban álló, de logikailag kapcsolódó adatokkal végzett számításokban. Csak egy a többhöz kapcsolatban álló táblákat használunk. A több a többhöz kapcsolatban álló táblák összetartozó adatainak lekérdezését már megismertük.



210. ábra

összetartozó adatok lekérdezése számított mezők képletében, egy a többhöz kapcsolatban álló táblákban

Az adott régióba tartozó megyék számát meghatározó M SZÁMA és az adott régióhoz tartozó városok számát adó V SZÁMA1 számított mezők képletének szerkezete azonos. A bővítmény a CAL-CULATE függvény segítségével keresi ki a másik táblában álló kapcsolódó adatokat. Nem okoz az sem problémát, ha a keresett adatokat tartalmazó tábla nem közvetlenül kapcsolódik a számított mezőt tartalmazó táblához.

A szintén az adott régió városainak számát adó V SZÁMA2 és V SZÁMA3 számított mezők a CAL-CULATETABLE illetve a RELATEDTABLE függvényekkel kérdezik le a kapcsolódó bejegyzéseket, majd az így létrejött egymezős virtuális tábla rekordjait a COUNTROWS függvénnyel számlálják meg.

A "városok" tábla két számított mezője, az M ARÁNY és az R ARÁNY, az adott város népességének és a város megyéje illetve régiója népességének arányát számítja ki.

A képleteket vizsgálva ne felejtsük el: a kapcsolat egy oldalán álló tábla tetszőleges rekordjához egy vagy több rekord tartozhat, míg a több oldali tábla tetszőleges rekordjához csak egy! Másként fogalmazva a kapcsolódó adatok lekérdezésének módját a képletet tartalmazó tábla kapcsolati pozíciója határozza meg.

A következő "kereső" függvény az EARLIER. Funkcióját már ismerjük, most elegendő ha csak zanzásítva rögzítjük: ciklust generál a feldolgozás alatt álló rekord meghatározott bejegyzésével. Emlékeztetőül és gyakorlásképen nézzünk egy egyszerű példát. A "gyártás" tábla egy vállalat telephelyeinek napi termelését rögzíti. Hozzunk létre egy számított mezőt, amelyben megállapítjuk a darabszám és az adott telephely legnagyobb darabszámának arányát!

1	gyártás AZ 🖬	telephely 🔽	dátum 🔽	darab 🔽	ARÁNY 1 🛛 🔽	ARÁNY 2 🛛 🔽
1	01	С	2017. 01. 03.	40	57,1%	57,1%
2	02	С	2017. 01. 13.	50	71,4%	71,4%
3	03	В	2017. 01. 27.	10	12,5%	12,5%
4	04	С	2017. 02. 02.	30	42,9%	42,9%
5	05	Α	2017_02. 18.	30	30,0%	30,0%

 ARÁNY 1
 =[darab] / CALCULATE(MAX([darab]); FILTER('gyártás'; [telephely] = EARLIER([telephely])))

 ARÁNY 2
 =[darab] / MAXX(FILTER('gyártás'; [telephely] = EARLIER([telephely])); [darab])

	211. á	bra	
a "gyártás"	tábla két	számítot	t mezővel

A legmagasabb értéket megkereshetjük közvetett módon a MAX függvénnyel a CALCULATE függvénybe ágyazottan, illetve közvetlenül a MAXX függvénnyel. A program MAX függvényétől eltérően a DAX MAX függvényének csak egyetlen argumentumot adhatunk meg: a vizsgálandó mező nevét. A második számított mező, ARÁNY 2 képletében álló MAXX függvénynek nincs az Excelben megfelelője: az első argumentumával meghatározott táblában, minden rekordban kiértékeli a második argumentumában álló kifejezést, és az eredmények közül a legnagyobbat választja ki.

Végül az EARLIEST függvényről kellene írnom, de nem tudok. A függvény leírása rendkívül szűkszavú és mintapéldát se publikált a Microsoft. Belső ciklust az EARLIER függvények egymásba ágyazásával nem tudunk létrehozni. Valószínűleg ezt a funkciót látja el az EARLIEST, csak az a kérdés: hogyan. Ha valaki tudja, ossza meg velünk a metódust!

statisztikai függvények

A DAX statisztikai függvényeinek többsége numerikus adatok elemzésére szolgál. Egy numerikus adattípusú mező bejegyzéseinek összegét, szorzatát, számtani és mértani közepét, darabszámát, minimumát, maximumát, szórását, varianciáját, mediánját és percentilisét statisztikai függvényekkel képezhetjük. A függvények egyetlen argumentuma az elemzett mező neve.

A felsorolt statisztikai vizsgálatokat elvégezhetjük egy tábla minden rekordjában kiértékelt kifejezés eredményhalmazán is. Ezek az "X"-es függvények már két-argumentumosak. Első argumentumukkal a táblát, második argumentumukkal a kifejezést kell deklarálnunk.

statisztikai érték	mezőbejegyzések elemzése	eredményhalmaz elemzése
összeg	SUM	SUMX
szorzat	PRODUCT	PRODUCTX
számtani és mértani közép	AVERAGE, GEOMEAN	AVERAGEX, GEOMEANX
darabszám	COUNT	COUNTX
minimum, maximum	MIN, MAX	MINX, MAXX
szórás	STDEV.P, STDEV.S	STDEVX.P, STDEVX.S
variancia	VAR.P, VAR.S	VARX.P, VARX.S
medián	MEDIAN	MEDIANX
percentilis	PERCENTILE.EXC, PERCENTILE.INC	PERCENTILEX.EXC, PERCENTILEX.INC

212. ábra

numerikus adatok elemzésére szolgáló statisztikai függvények

A felsorolt függvények működése, beleértve az üres bejegyzések kezelését is, megegyezik a program azonos funkciójú függvényeinek működésével: SUM (SZUM), PRODUCT (SZORZAT), AVERAGE (ÁTLAG), GEOMEAN (MÉRTANI.KÖZÉP), COUNT (DARAB), MAX, MIN, STDEV.P (SZÓR.S), STDEV.S (SZÓR.M), VAR.P (VAR.S), VAR.S (VAR.M), MEDIAN (MEDIÁN), PERCENTILE.EXC (PERCENTILIS.KI-ZÁR) és PERCENTILE.INC (PERCENTILIS.TARTALMAZ).

A statisztikában jártas felhasználók választhatnak a szórás és a variancia függvények "P" (population) és "S" (sample), valamint a percentilis függvények végpontokat kizáró "EXC" illetve a végpontokat is figyelembe vevő "INC" változatai között. Dátumok átlagának képzésekor előfordulhat, hogy a kapott számot nem tudjuk dátumként megjeleníteni, ekkor a képletben kell a megjelenítést szabályoznunk: FORMAT(ROUND(AVERAGE([dátum]); o); "General Date").

A logikai adattípusú mezők statisztikai elemzéséhez az AVERAGE, a COUNT, a MAX és a MIN függvények "A" betűs változatait használhatjuk: AVERAGEA, a COUNTA, a MAXA és a MINA. A számításkor az IGAZ érték egy, a HAMIS értéke nulla. A COUNTA függvényt szöveg adattípusú mezők bejegyzéseinek megszámlálására is használhatjuk.

Megszámlálást végez a COUNTBLANK és a COUNTROWS függvény is. Előbbi egy tetszőleges adattípusú mező üres bejegyzéseit, utóbbi egy valós vagy virtuális tábla rekordjait számlálja meg. Az elemzendő objektumot a függvények egyetlen argumentumával deklaráljuk.

Numerikus adatok elemzésekor gyakori művelet egy szám az őt tartalmazó halmaz számai rangsorában elfoglalt pozíciójának meghatározása. Például, Balogh János előző évi árbevétele, hányadik a vállalat többi üzletkötőjének árbevételeihez viszonyítva. Második, ha csak egyetlen árbevétel nagyobb az övénél. A rangsor-pozíció tehát egy sorszám. A szokásos eljárásban a legnagyobb szám kapja az egyest, és az egyre kisebb számok egyre nagyobb sorszámúak, de dönthetünk úgy is, hogy a legkisebb számé lesz az egyes, és a számok növekedésével a sorszámuk is növekszik.

A DAX három rangsor-kezelő függvénnyel rendelkezik: RANK.EQ, a RANKX és a TOPN. Egy szám az őt tartalmazó számok rangsorában elfoglalt pozíciójának sorszámát a RANK.EQ függvénnyel tudjuk megállapítani. A függvény első argumentuma a vizsgált szám, a második az őt tartalmazó halmaz megadására szolgál. A két argumentum az esetek többségében azonos, a rangsorolásra kerülő számokat tartalmazó mező neve.

működés	egyes sorszámú elem	argumentum
sorszámozás a rangsor szerint	legnagyobb	nincs megadva / o / DESC
sorszámozás a rangsorral ellentétesen	legkisebb	1/ASC

213. ábra

a RANK.EQ függvény harmadik, a sorszámozás irányát meghatározó, argumentuma

A függvény leírása szerint az első argumentum kifejezés is lehet, sőt a második argumentumban hivatkozhatunk származtatott mezőre is. Utóbbi lehetőség nem vonatkozik a ADDCOLUMNS, a ROW és a SUMMARIZE függvényekkel létrehozott mezőkre.

Az elemzett halmazban álló azonos számok rangsor-pozíciója is azonos. Az őket követő szám rangsor-pozíciója az ismétlődő sorszám darabszáma és a többször előforduló rangsor-pozíció öszszege lesz. Például három kilences sorszám után a tizenkettes következik: 3 + 9. Az azonos rangsorpozíciók kezelésének ezt a módját SKIP módszernek nevezzük.

	1	AZ 🔽	szám		RSOR 💽	FORD
	1	01		40	8	1
BEOB		02	-11	80	2	5
FORD	=RANK.EQ((szám) ; (szár (szám) : (szár	nj) nl:1)	90	1	8
	4	04		80	2	5
	5	05		60	6	3
	6	06		70	5	4
	7	07		50	7	2
	8	08		80	2	5

214. ábra a RANK.EQ függvény működésének bemutatása két számított mezővel

A RANKX függvény, a többi X-es függvényhez hasonlóan, egy kifejezést értékel ki egy tábla minden rekordjában, majd az így nyert számhalmaz rangsorában megállapítja egy szám pozícióját. A függvény öt argumentuma, sorrendben a következő.

[1] A táblát vagy a táblát adó kifejezés. [2] A minden rekordban kiértékelendő kifejezés. [3] A szám, amelynek rangsor pozícióját kell megállapítani a második argumentum kiértékelése során kapott eredményhalmazban. [4] A sorszámozás irányát meghatározó argumentum. Szintaktikája azonos a RANK.EQ függvény azonos funkciójú argumentumával. [5] A sorszámozás módját előíró argumentum.

működés	azonos számok sorszámai	argumentum
szakadozott sorszám-kiosztás	1, 2, 3, 4, 5, 5, 5, 8, 9, 10, 11, 11, 11, 14	nincs megadva / Skip
folytonos sorszám-kiosztás	1, 2, 3, 4, 5, 5, 5, 6, 7, 8, 9, 9, 9, 10	Dense

215. ábra

a RANKX függvény ötödik, a sorszámozás módját meghatározó, argumentuma

Ahogy a képen megfigyelhetjük a DENSE módszert alkalmazva az ismétlődő sorszámot követő elem az ismétlődő sorszámot követő sorszámot kapja.

A RANKX függvénynek csak az első két argumentuma kötelező. A harmadik argumentum, amely kifejezés is lehet, hiánya a leggyakoribb vizsgálat végrehajtását írja elő: a számított érték pozícióját keresd a számított értékek rangsorában! Ha a nem kötelező argumentumok közül az argumentumlista közbenső elemét hagyjuk el, ezt üres argumentummal kell jeleznünk. Például nem adjuk meg a harmadik és negyedik elemet: RANKX(<tábla> ; <kifejezés> ; ; ; DENSE).

4	munkatárs 🔽	kezdés 🔽	zárás 💽	darab 🔽	RSOR 🗾	RSOR D
1	Kun Balázs	2017. 03. 13.	2017. 03. 16.	12	1	1
2	Kő Alfréd	2017. 03. 15.	2017. 03. 17.	12	3	3
3	Tar Mátyás	2017. 03. 11.	2017. 03. 13.	10	2	2
4	Pap Elemér	2017. 03. 12.	2017. 03. 16.	24	3	3
5	Nagy Zoltán	2017. 03. 10.	2017. 03. 14.	32	7	5
6	Kis Kálmán	2017. 03. 14.	2017. 03. 17.	18	3	3
7	Tóth Bálint	2017. 03. 12.	2017. 03. 14.	14	6	4

216. ábra a minőség-ellenőrzés adatait tartalmazó "selejt" tábla

A képen látható tábla két számított mezője a napi selejt értéke alapján rangsorolja a munkatársakat. Az RSOR szakadozott sorszámot (SKIP) ad: RANKX(selejt ; [darab] / ([zárás] - [kezdés]) ; ; 1), míg az RSOR D folyamatos sorszámozást (DENSE) végez: RANKX(selejt ; [darab] / ([zárás] - [kezdés]) ; ; 1 ; DENSE).

A kimutatás tételeink rangsorát is a RANKX függvénnyel állapíthatjuk meg. Vegyünk egy egyszerű adatbázist a "termékek" és a "eladások" táblákkal!

				eladás A	XZ 🔽	termék AZ 🕴	₽ 🔽	darab	1		
	termák A7 🖳 🗖	náv				04				5	
1		A				03				3	
-	01	~				02				7	
2	02	в				03				7	
3	03	с				02				-	
4	04	D				01	ne	v	-	Osszeg - darab	rangsor
5	05	E				01	- 1	4		16	3
			1		U	02		5		23	1
			8	08		04		5		10	4
				00			- I	D		17	2
			9	09		04				9	5
			10	10		05				6	
			11	11		02			_	5	

217. ábra a példa két táblája és az elemzésükre létrehozott kimutatás

Először egy automatikus összesítéssel megállapítottam a számok összegét betűk szerinti bontásban, majd a "rangsor" egyéni összesítéssel megállapítottam a betűk rangsorát az összesített számok alapján. rangsor:= RANKX(ALL('termékek'[név]); 'eladások'[Összeg - darab]).

Mindhárom rangsor-kezelő függvénynek, tehát a később tárgyalandó TOPN-nek is, van egy közös tulajdonsága: képesek szöveg-halmazok elemeit is rangsorolni. Ez az alapja a karakterláncok ABC szerinti rendezésének. Növekvő rendezés esetén az egyes sorszámot a rendezett halmaz utolsó eleme kapja, tehát az ABC végén álló betűkkel kezdődő karakterláncok.

A következő képen látható "termények" tábla "RSOR 1" számított mezője rangsorolja a "termény" mező bejegyzéseit a RANK.EQ függvénnyel. Az "RSOR 2" számított mező a "termény" mező bejegyzéseinek második karaktere alapján állapítja meg a szövegek rangsorár a RANKX függvény segítségével.

1	termény A	Z 🔽	termény 🔽	RSOR 1 💽	RSOR 2					
1	01		lencse	1	2					
2	02		bab	4	3					
3	03		lencse	1	2					
4	04		borsó 3		1					
_					1					
	RSOR 1 =RANK.EQ([termény];[termény])									
	RSOR 2 =RANKX('termények' ; MID([termény] ; 2 ; 1) ; ; ; Dense)									

218. ábra szövegek rangsorának kezelése a RANK.EQ és a RANKX függvényekkel

A DAX MID függvényének funkciója és szintaktikája azonos a program KÖZÉP függvényével.

A mindennapi életben a rangsor és a hierarchia összetartozó fogalmak. Bemutatok egy típusfeladatot, amelynek algoritmusában összemosódik a rangsor- és a hierarchia kezelés. Van egy táblánk, amelynek egyik mezőjében minden bejegyzés egyedi. Például a "megyék" táblában a "megye" mező bejegyzései. Ezek a bejegyzések csoportokba sorolhatók, a tábla egy másik mezőjének bejegyzései alapján. A példánknál maradva, a megyék a "régió" mező bejegyzései alapján csoportosíthatók. A PowerPivot-kimutatással a csoportokat kívánom elemezni, de láttatni szeretném a csoport tagjait is. Azaz, meg kívánom állapítani a régiók városlakóinak számát, úgy, hogy a régiók neve mellett látható legyen a régióhoz tartozó megyék listája is. Valahogy így!

	régió 🗾 🔽	megye 🔽	városi 🔽
1	Közép-Magyarország	Pest	2 367 753
2	Észak-Alföld	Hajdú-Bihar	431 311
3	Észak-Magyarország	Borsod-Abaúj-Zemplén	416 526
4	Dél-Alföld	Bács-Kiskun	346 716
5	Dél-Alföld	Csongrád	314 162
6	Észak-Alföld	Szabolcs-Szatmár Pereg	280 539

régió	Ŧ	MEGYÉK	Ŧ	Összeg - városi 💌
Dél-Alföld		Bács-Kiskun Békés Csongrád		936 722
Dél-Dunántúl		Baranya Somogy Tolna		557 666
Észak-Alföld		Hajdú-Bihar Jász-Nagykun-Szolnok Szabolcs-Szatmár-Bereg		987 002
Észak-Magyarország		Borsod-Abaúj-Zemplén Heves Nógrád		658 076
Közép-Dunántúl		Fejér Komárom-Esztergom Veszprém		646 338
Közép-Magyarország	3	Pest		2 367 753
Nyugat-Dunántúl		Győr-Moson-Sopron Vas Zala		568 470



A feladat megoldását az egy régióba tartozó megyék rangsorolásával kezdjük. Ezután a sorszámok alapján listát hozunk létre a megyék nevéből, majd a listát a PATH függvénnyel íratjuk ki. A példa algoritmusa a következő.

[1] A megyék sorszámozása, régiónként újrakezdve. RS: =RANKX(FILTER(ALL('megyék'); [régió] = EARLIER([régió])); [megye];; 1; DENSE).

[2] A régió rangsorában az adott megyét megelőző megye sorszámának képzése. FEL RS: =IF([RS] <> 1; [RS] - 1).

[3] A felettes megye nevének képzése. P1: =CALCULATE(FIRSTNONBLANK('megyék'[megye]; 1); FILTER(ALLEXCEPT('megyék'; 'megyék'[régió]); [RS] = EARLIER([FEL RS]))).

[4] A következő lépés az adott megye függőségi sorának képzése. P2: =PATH([megye]; [P1]).

[5] Az utolsó művelet minden megyéhez kiíratni a saját régiójában, a rangsor végén álló megye függőségi sorát. MEGYÉK: =CALCULATE(PATH([megye] ; [P1]) ; CALCULATETABLE(FILTER('megyék'; [RS] = MAX([RS])) ; ALLEXCEPT('megyék'; 'megyék'[régió]))).

	régió 🔽	megye 🔽	városi 🔽	RS 💌	FEL RS 💌	P1 💌	P2
1	Dél-Alföld	Bács-Kiskun	346 716	1			Bács-Kiskun
2	Dél-Alföld	Békés	275 844	2	1	Bács-Kiskun	Bács-Kiskun Békés
3	Dél-Alföld	Csongrád	314 162	3	2	Békés	Bács-Kiskun Békés Csongrád
4	Dél-Dunántúl	Baranya	258 582	1			Baranya
5	Dél-Dunántúl	Somogy	164 252	2	1	Baranya	Baranya Somogy
6	Dél-Dunántúl	Tolna	134 832	3	2	Somogy	Baranya Somogy ToIna
7	Észak-Alföld	Haidú-Bihar	431 311	1			Haidú-Bihar

220. ábra a számított mezőkkel kiegészített tábla részlete

A képen a megyéket, régiónként újrakezdve, ABC sorrendben ábrázoltam az algoritmus könnyebb megértése érdekében. A feladat megoldása tehát egy hierarchia-szimuláción alapult, amelyet az egy kategoriába tartozó elemek sorszámozása tett lehetővé. Az elemek csoportonkénti indexelése részösszegek képzését is lehetővé teszi. Ezt a lehetőséget mutatom be a következő példán.

1	AZ 🔽	ügynök 🖃	eladás 🛛 💌	SSZÁM 💽	ELDB 💌	UTOLSÓ 💽	RÖSSZ 🗾
1	01	Α	2	1	3	HAMIS	
2	03	Α	2	2	3	HAMIS	
3	07	Α	3	3	3	IGAZ	7
4	02	В	4	1	2	HAMIS	
5	05	В	1	2	2	IGAZ	5
6	04	С	2	1	2	HAMIS	
7	06	С	2	2	2	IGAZ	4

221. ábra

az "eladások" tábla a részösszeg-képzés négy számított mezőjével

A képen látható tábla eladásait szeretnénk összegezni ügynökök szerint. Az áttekinthetőség érdekében a feladatot négy lépésben, külön-külön számított mezőkkel oldjuk meg!

[1] A tábla rendezése a csoportosító mező (ügynök) értékei alapján.

[2] Az ügynök eladásainak sorszámozása. SSZÁM: = RANKX(FILTER(ALL('eladások'); [ügynök] = EARLIER([ügynök])); [AZ];; 1; DENSE).

[3] Az ügynök eladásai darabszámának képzése. ELDB: = COUNTROWS(FILTER(ALL('eladások') ; [ügynök] = EARLIER([ügynök]))).

[4] Az ügynök utolsó eladásának megállapítása. UTOLSÓ: = [SSZÁM] = [ELDB].

[5] Részösszeg képzése. RÖSSZ: = IF([UTOLSÓ]; CALCULATE(SUM([eladás]); ALLEXCEPT('eladások'; 'eladások'[ügynök]))). 222. ábra a részösszeg számítás egyetlen képletben

A DAX harmadik rangsor-kezelő függvénye a TOPN függvény. A függvény, a Microsoft megfogalmazása szerint, egy virtuálisan rendezett tábla első n rekordját adja eredményül. A függvény argumentumlistája négy elemből áll. [1] A visszaadandó rekordok számát adó kifejezés vagy konstans. [2] A tábla neve vagy táblát eredményező kifejezés. [3] A rendező-mező neve. [4] Nem kötelező argumentum a rendezési sorrend megfordítására: ha nem a rangsor szerint csökkenő, hanem emelkedő rendezést szeretnénk, akkor az argumentum értéke egy (1).

1	város AZ 🛛 🔽	város 🔽	lakos 🔽
1	001	Abony	15 679
2	002	Ajka	31 805
3	003	Aszód	6 428
4	004	Bácsalmás	7 650
5	005	Baja	37 916
6	006	Baktalórántháza	4 136
7	007	Balassagyarmat	18 474
	3In: 399 556	31k: 5 096	
		~ ~	

223. ábra a "városok" tábla két egyéni összesítéssel

A képen látható tábla a magyar városok és a fővárosi kerületek lakosságát tartalmazza. Kíváncsiak vagyunk a három legnagyobb és a három legkisebb lakosságú város lakosainak összegére. Az első kérdésünkre a "3ln" válaszol: SUMX(TOPN(3; 'városok'; [lakos]); [lakos]). A három legkisebb lakosságú város lakóinak összegét a "3lk" adja: SUMX(TOPN(3; 'városok'; [lakos]; 1); [lakos]). A két képlet majdnem azonos, csak a TOPN függvény negyedik argumentumában különbözik. Az első képletből hiányzik ez az argumentum, tehát a rendezés sorrendje növekvő, míg a második képlet csökkenő sorrendű rendezést ír elő.

A TOPN függvény képes több-kulcsos virtuális rendezésre is, sőt a rendezés alapja számított érték is lehet. Nézzünk egy példát e két tulajdonság szemléltetésére is!

1	ügynök AZ 🛛 🔽	kategória 🗾	bevétel 💽	kezdés 💽	zárás 🗾 💌
1	01	A	800 000 Ft	2019. 01. 23.	2019. 02. 17.
2	02	В	1 050 000 Ft	2019. 05. 23.	2019. 07. 12.
3	03	A	1 100 000 Ft	2019. 05. 14.	2019. 06. 20.
4	04	A	1 150 000 Ft	2019. 04. 24.	2019. 06. 20.
5	05	С	900 000 Ft	2019. 04. 30.	2019. 06. 06.
6	06	С	1 250 000 Ft	2019. 01. 24.	2019. 03. 25.
7	07	В	1 050 000 Ft	2019. 05. 02.	2019. 06. 20.
	13ö: 2 900 000				

^{224.} ábra a "ügynökök" tábla

Az "ügynökök" tábla kategorizált üzletkötők, különböző időszakokra eső árbevételét tartalmazza. A munka hatékonyságát jól jellemzi az árbevétel egy napra eső hányada, amelyet a határnapok segítségével egyszerűen képezhetünk. Az "13ö" egyéni összesítéssel a legmagasabb kategoriába (A) sorolt üzletkötők, három legnagyobb napi árbevételt produkált bevételének összegét számoljuk ki: =CALCULATE(SUM([bevétel]) ; TOPN(3 ; 'ügynökök' ; [kategoria] ; 1 ; [bevétel] / ([zárás] - [kezdés]))). A TOPN függvény argumentumlistájában tehát a második elem után, az argumentumok párban állnak: rendező értékek – rendezési sorrend. A szokásos rendezési sorrend a csökkenő, ezt az argumentumlista végén nem kell megadnunk, közbenső argumentum esetén nullával (0) vagy üres argumentummal kell jelölnünk.

A rangsor kezelő DAX függvények tehát a skip és a dense módszerrel rangsorolják a halmaz azonos elemeit. Mindkét eljárás ismétlődő sorszámokat eredményez és ez nem minden helyzetben elfogadható. Az ismétlődéseket nem tartalmazó rangsor-pozíció képzése két statisztikai érték összegéből áll: [1] az elemzett szám hányszor fordul elő a halmazban az elemzett szám pozíciójáig, az elemzett számot is beleértve, [2] hány szám nagyobb a halmazban az elemzett számnál. Nézzünk egy példát!

Adott egy kilenc-rekordos tábla. Két mezője: a kulcs mező és egy számokat tartalmazó mező. Az ismétlődések nélküli rangsor-pozíciókat, a képletek jó olvashatósága érdekében, három számított mezővel képezzük. Először az EDDIG számított mezővel megállapítjuk az egyes számok előfordulásának darabszámát a vizsgált számot tartalmazó rekorddal bezárólag. Ezután a NAGYOBB számított mezővel megszámláljuk, hány szám nagyobb a táblában a vizsgált számnál. Végül a RSOR számított mezővel összeadjuk a két statisztikai értéket. A negyedik, FORD nevű, számított mezővel fordított rangsort képeztem.

1	AZ 🔽	szám 🔽	EDDIG 🗾	NAGYOBB 🔽	RSOR 🗾	FORD 🔽			
1	01	54	1	3	4	6			
2	02	72	1		1	8			
3	03	72	2		2	9			
4	04	41	1	4	5	3			
5	05	69	1	2	3	7			
6	06	33	1	7	8	2			
7	07	15	1	8	9	1			
8	08	41	2	4	6	4			
9	09	41	3	4	7	5			
				1	1	1			
ì	=COUNTRO	WS(FILTER(F	FILTER('számok	' ; [AZ] <= EARLIE	R([AZ])) ; [szár	n] = EARLIER([sz			
BB	=COUNTROWS(FILTER('számok' ; [szám] > EARLIER([szám])))								
	=[EDDIG] +	[NAGYOBB]							
	=[EDDIG] +	COUNTROWS	SCELLTERC'szám	ok' · [szám] < FA	RUER([szám])	11			

225. ábra az "számok" tábla és négy számított mezőjének képlete

A TOPN függvényt most már a saját számításon alapuló rangsorral tudjuk használni. Például képeznünk kell a három legnagyobb szám összegét: =SUMX(TOPN(3; A; [RSOR]); [szám]).

Természetesen az ismétlődések nélküli rangsor-pozíció képzésnek ez a módszere, csak akkor használható, ha a tábla rendelkezik egy olyan numerikus- vagy számmá konvertálható adattípusú mezővel, amely folyamatosan növekvő értékeket tartalmaz az egyes rekordokban.

szöveg-kezelő függvények

A karakterláncok feldolgozását végző DAX függvényeket két csoportba sorolhatjuk: [1] a szöveg valamely tulajdonságát vizsgálókra és [2] a kiinduló karakterláncot átalakítókra.

Az UNICODE függvény az egyetlen argumentumával megadott szöveg első karakterének számát adja az UNICODE rendszerben. Az ellentétes műveletnek, számból karaktert, nincs függvénye.

Az első csoportba tartozik a bővítmény LEN függvénye, amely az egyetlen argumentumával meghatározott szöveg karaktereinek darabszámát adja eredményül.

Karakterláncok összehasonlítását az EXACT függvénnyel kérhetjük. Az összehasonlítandó szövegeket a függvény két argumentumával adjuk meg. A vizsgálat eredménye logikai érték: ha a két szöveg azonos, akkor IGAZ, különben HAMIS. Az alábbi ábrán a függvény működést mutatom be.

1	AZ 🔽	szöveg1 🔽	szöveg2 🔽	AZONOS 🔽	logmű 🛛 🔽
1	01	tűrő	TŰRŐ	HAMIS	IGAZ
2	02	tűrő	turo	HAMIS	HAMIS
3	03	tűr ő	tűr ő	IGAZ	IGAZ
4	04	tűr ő	tűr ő	HAMIS	HAMIS

226. ábra az AZONOS számított mező képlete: EXACT([szöveg1] ; [szöveg2])

A kép tanúsága szerint a függvény megkülönbözteti a kis- és nagybetűket, az ékezet nélküli és az ékezetes betűket, sőt még a különböző kódú szóközkaraktereket is. Utóbbi ok miatt kaptunk HA-MIS eredményt a negyedik rekordban. Amennyiben az összehasonlításban nincs szerepe a kis és a nagybetűk eltérésének, használjunk logikai műveletet: = [szöveg1] = [szöveg2]. Ennek a kifejezésnek az eredményét láthatjuk a kép LOGMŰ számított mezőjében.

Szövegben karakterláncot a FIND és a SEARCH függvénnyel kereshetünk. Argumentumaik azonosak: [1] A keresett szöveg meghatározása. Ha konstans, idézőjelek között kell állnia. [2] A gazdaszöveg meghatározása. [3] A keresés induló-karakterének sorszáma a gazdaszövegben. Másként fogalmazva, a gazdaszöveg hányadik karakterétől kezdődően keressen a függvény. Nem kötelező argumentum. Szokásos értéke egy. Elhagyását üres argumentummal kell jelölni. [4] A függvény eredménye ha nincs találat. Nem kötelező argumentum. A bővítmény csak előjeles egész számot vagy a BLANK függvényt fogadja el ebben az argumentumban.

A függvény eredménye: [1] sikeres vizsgálatot követően, a keresett szöveg első karakterének sorszáma a gazdaszövegben, [2] sikertelen vizsgálatot követően, a negyedik argumentumban megadott szám vagy üres bejegyzés. Sikertelen vizsgálatot követően, ha a negyedik argumentum nincs megadva, a vizsgálat hibaértéket eredményez.

A keresés kezdő pozíciójának deklarálása a találat pozíciójának sorszámozását nem befolyásolja.

1	AZ 🔽	keresett sz 🛛 💌	gazdasz 🛛 🔽	FIND 🗾	SEARCH 💽
1	01	lapos	kalaposinas	3	3
2	02	szent	PESTSZENTIMRE		5
3	03	MÁR	KALAMÁRIS	5	5
4	04	aj tó	tutaj tóval		
5	05	Tojás			
6	06		Tóbiás	1	1

227. ábra szöveg keresése a FIND és SEARCH függvényekkel

A képen látható tábla két számított mezőjének képlete azonos: FIND/SEARCH([keresett sz]; [gazdasz]; ; BLANK()). A függvények alapvetően azonosan működnek, de a SEARCH nem tesz különbséget kis- és nagybetűk között. Ezt látjuk a második rekordban. Mindkét függvény az eltérő kódú szóközöket különböző karaktereknek tekinti. Ezt a tulajdonságot a negyedik rekordban figyelhetjük meg. Az üres, vagy nulla hosszúságú gazdaszöveg sikertelen vizsgálatot, az üres vagy nulla hosszúságú keresett szöveg egyet (1) eredményez.

A SEARCH függvény első és második argumentumában helyettesítő karakterek (*, ?) is állhatnak. A mezőnevekhez és kifejezésekhez szövegösszefűzéssel kapcsolhatjuk a joker-karaktereket: "?" & [mezőnév]. Ha magát a csillagot vagy a kérdőjelet keressük a gazdaszövegben, akkor ezt tilde karakterrel jelöljük (~*, ~?). A hullámvonal karaktert is szövegösszefűzéssel kell a mezőnévhez vagy a kifejezéshez kapcsolni.

Mint említettem, a negyedik argumentum csak előjeles egész szám vagy üres bejegyzés lehet. A korlátozást az IFERROR függvénnyel nem tudjuk kikerülni, mert a bővítmény ennél a függvénynél sem enged az első argumentumától eltérő adattípusú eredményt deklarálni. A megoldást az IF és az ISERROR függvények kombinációja adja: IF(ISERROR(FIND/SARCH(<keresett szöveg> ; <gazdaszöveg>)) ; <alternatív adattípusú eredmény>).

A szövegkezelő függvények második csoportjába azokat a függvényeket soroljuk, amelyek a kiindulásként kapott karakterlánc átalakítását végzik. A REPT függvény az első argumentumával meghatározott karakterláncot fűzi össze a másolataival. Az eredményül kapott karakterlánc, a függvény második argumentumával meghatározott darabszámban fogja tartalmazni a kiinduló szöveget.

A CONCATENATE függvény a két argumentumával meghatározott szöveget fűzi össze egyetlen karakterlánccá. Ha több elemet szeretnénk egyesíteni, használjuk inkább az ampersand (&) operátort.

4	AZ 🔽	osztály 🔽	csoport 🔽	KÓD1 💽	KÓD2 💽
1	01	В	1	В	B1
2	02	С	2	CC	C2
3	03	Α	1	Α	A1
4	04	В	3	BBB	B3
5	05	Α	1	Α	A1
6	06	В	2	BB	B2

228. ábra a KÓD1 és KÓD2 számított mezők képlete: REPT/CONCATENATE([osztály] ; [csoport])

A CONCATENATEX függvény a tábla minden rekordjában kiértékel egy kifejezést, majd az eredményeket szövegként összefűzve jeleníti meg. A lista elemeinek sorrendjét és a szeparációját a felhasználó határozhatja meg. Argumentumlistája a következő. [1] Egy tábla vagy táblát eredményező kifejezés. [2] A tábla minden rekordjában kiértékelendő kifejezés. [3] A lista elemeit elválasztó karakter vagy karakterlánc, idézőjelek között megadva. Nem kötelező argumentum. [4] A rendezés alapját adó mező neve vagy egy kifejezés, amelynek eredményei alapján lesz rendezve a lista. Nem kötelező argumentum. [5] A rendezés irányát meghatározó argumentum. Lehetséges értékei: ASC, DESC. Nem kötelező argumentum. A rendezés több kulcs alapján is történt, ezért a négyes és az ötös argumentum ismételhető.

A függvény működését "szerszámok" táblán mutatom be. A tábla tartalmazza a szerszám nevét, gyártóját, árát és az eladás darabszámát. Készítsünk kimutatást, amely összegzi az eladásokat gyártók szerint! A pivot-táblában jelenítsük meg az adott szerszám választékát is!

	szerszám AZ 💌	név	¥	gyártó	×	ár 💌	darab	×		név	gyártó	ár
1	01	fúró		O&A		60		7		fúró	Bart	30 HUF
-								-		fúró	0&A	60 HUF
2	02	veso		Berry		200		8		fúró	Sempl	150 HUF
3	03	fúró	név	/	-	vála	szték		Osszeg - darab	fogó	Taro	360 HUF
4	04	véső	f	ogó		Taro Bart			25	fogó	Bart	270 HUF
-			f	úró		Sempl 0&/	A Bart		42	véső	AloCo	350 HUF
5	05	furesz	fi	űrész		Perry Sós	T&G Zo	pe	56	véső	Certi	50 HUF
6	06	fogó	V	éső		AloCo Berr	y Certi		79	véső	Berry	200 HUF
7	07	fúró		Bart		30		8		fűrész	Zope	280 HUF
	00			Denne		200				fűrész	T&G	350 HUF
8	08	veso		Berry		200		4		fűrész	Sós	350 HUF
9	09	fűrész		Perry	_	420		4		fűrész	Perry	420 HUF

választék =CONCATENATEX(SUMMARIZE('szerszámok' ; [név] ; [gyártó] ; [ár]) ; [gyártó] ; " | " ; [ár] ; DESC)

229. ábra a példa objektumai

A "választék" egyéni összesítés képletében a SUMMARIZE függvény összesítő táblát hoz létre a "szerszámok" táblából a "név", a "gyártó" és az "ár" mezők egyedi bejegyzései alapján. Ezt a virtuális táblát láthatjuk a kép bal oldalán. Az UPPER és a LOWER függvények a karakter-konverzió eszközei. Előbbi a kiindulásként kapott karakterlánc kisbetűit nagybetűkre, utóbbi a karakterlánc nagybetűit kisbetűkre cseréli.

A LEFT, a RIGHT és a MID függvények az első argumentumukkal megadott karakterlánc egy darabját adják eredményül. A LEFT és a RIGHT függvények az első argumentumukkal meghatározott szöveg elejéről illetve végéről, a második argumentumukkal megadott darabszámú karaktert adják eredményül. A szöveg közepéről a három-argumentumos MID függvénnyel "vághatunk" ki egy darabot. A gazdaszöveget az első, az eredmény-szöveg első karakterének sorszámát a második és az eredmény-szöveg karaktereinek számát a harmadik argumentummal határozhatjuk meg. Tehát: MID(<gazdaszöveg>; <az eredmény-szöveg első karakterének sorszáma a gazdaszövegben>; <az eredmény-szöveg hossza karakterben>).

	város AZ 🛛 🔽	város 🛛 🔽	megye 🔽	régió 🔽	RÉGIÓR 🛛 🔽
1	001	Abony	Pest	Közép-Magyarország	KM
2	002	Ajka	Veszprém	Közép-Dunántúl	KD
3	003	Albertirsa	Pest	Közép-Magyarország	КМ
4	004	Aszód	Pest	Közép-Magyarország	КМ
5	005	Bábolna	Komárom-Esztergom	Közép-Dunántúl	KD
6	005	Bácsalmás	B≜cs-Kiskun	Dél-Alföld	DA

230. ábra példa a MID függvény alkalmazására a RÉGIÓR számított mezőben

Nézzünk egy komplex feladatot, a MID függvény alkalmazására. A fenti tábla magyar városok adatait tartalmazza. Hozzunk létre egy számított mezőt, amely a város régiójának nevét rövidítve mutatja. Minden régió neve két, nagybetűvel kezdődő karakterláncból áll, kötőjellel elválasztva. A rövidítés a két karakterlánc első karaktereiből álljon!

A rövidítés első betűje a "régió" mező bejegyzésének első karaktere: LEFT([régió]; 1). A rövidítés második betűjét a mező minden bejegyzésében a kötőjel előzi meg. Állapítsuk meg az elválasztó karakter pozícióját a "régió" mezőben: FIND("-"; [régió]). A rövidítés második karakterének pozíciója a "régió" mezőben: FIND("-"; [régió]) + 1. Magát a karaktert a pozíciója ismeretében a MID függvénnyel képezhetjük: MID([régió]; FIND("-"; [régió]) + 1; 1). A rövidítés két betűjének összefűzését a CONCATENATE függvénnyel végezzük: CONCATENATE(LEFT([régió]; 1); MID([régió]; FIND("-"; [régió]) + 1; 1)).

A REPLACE és a SUBSTITUTE függvények a kiindulásként kapott karakterlánc egy darabját lecserélik egy másik karakterláncra. Az átalakítandó karakterlánc a gazdaszöveg. A gazdaszöveg azon darabja, amely törlésre kerül, a cserélendő szöveg. A törlés után a gazdaszövegbe illesztendő karakterlánc az új szöveg. A REPLACE függvény alkalmazásakor a cserélendő szöveg pozícióját és hoszszát, a SUBSTITUTE esetén magát a cserélendő szöveget kell megadnunk.

A REPLACE függvény argumentumai a következők. [1] A gazdaszöveg deklarációja. [2] A cserélendő szöveg első karakterének sorszáma a gazdaszövegben. [3] A cserélendő szöveg karaktereinek darabszáma. [4] Az új szöveg deklarációja. Üres karakterláncot ("") adjunk meg, ha csak a cserélendő szöveg törlése a célunk!

A SUBSTITUTE függvény argumentumai a következők. [1] A gazdaszöveg deklarációja. [2] A cserélendő szöveg deklarációja. [3] Az új szöveg deklarációja. [4] Egy pozitív egész szám, a cserélendő szöveg cserélendő előfordulásának sorszáma. Másként fogalmazva, ezzel az argumentummal határozhatjuk meg azt, hogy a gazdaszövegben többször előforduló cserélendő szöveg hányadik előfordulása legyen kicserélve. Nem kötelező argumentum. Ha nincs megadva a cserélendő szöveg összes előfordulásában ki lesz cserélve az új szövegre. A SUBSTITUTE függvény a cserélendő szöveg keresésekor a kis és nagybetűket megkülönbözteti. Ha a függvény nem találja a gazdaszövegben a cserélendő szöveget, akkor a gazdaszöveget adja eredményül.

Lássunk egy komplex feladatot, amelynek megoldásában a SUBSTITUTE függvény is szerepel! Adott egy kétmezős tábla, amelyben autóbusz járatok azonosítóját ("járat" mező) és az útjuk során érintett városok listáját ("útvonal" mező) tartalmazza, szóközökkel elválasztva. Hozzunk létre számított mezőt, INDULÓ ÁLL néven, amely az útvonal első városát mutatja!

1	járat 💌	útvonal 💌	INDULÓ ÁLL 💽
1	BU015	Budapest Hatvan Miskolc Nyíregyháza	Budapest
2	SO055	Szolnok Békéscsaba Lököshaza	Szolnok
3	DE013	Debrecen Záhony Csao	Debrecen
4	BU654	Budapest Pusztaszabolcs Pécs	Budapest
5	SÉ052	Székesfehérvár Siófok Nagykanizsa	Székesfehérvár
6	K0007	Komárom Győr Sopron	Komárom

231. ábra a példa táblája az INDULÓ ÁLL számított mezővel

A bővítmény nyelvén így fogalmazhatnánk meg a feladatot: ki kell íratnunk az "útvonal" mező bejegyzéséből az első szóközig tartó darabot. Az első szóköz pozíciójának sorszámát a bejegyzésben a FIND függvénnyel határozhatjuk meg: FIND(""; [útvonal]). Ha a kapott számból kivonunk egyet az első város karaktereinek darabszámát kapjuk. Ennek ismeretében a LEFT függvénnyel már kiírathatjuk az induló állomás városának nevét: LEFT([útvonal]; FIND(""; [útvonal]). - 1).

A második feladat a végállomás városának kiíratása. Vegyük sorra a megoldás lépéseit! [1] Megszámláljuk, hány szóközt tartalmaz az útvonal. [2] Az utolsó szóközt lecseréljük egy olyan karakterre, amely biztosan nem szerepel az útvonal karakterei között. [3] Megállapítjuk a beillesztett karakter pozíciójának sorszámát az útvonalban. [4] A sorszámból és az útvonal karaktereinek számából kiszámítjuk az utolsó városnév karaktereinek darabszámát. [5] Kiíratjuk a az útvonal végéről a kapott darabszámú karaktert.

[1] Az egyes lépéseket, az áttekinthetőség érdekében, külön-külön számított mezőkkel oldjuk meg. Az útvonalban álló szóközök számát az útvonal karaktereinek száma és a szóközöktől megtisztított útvonal karakterei számának különbsége adja. A szóközök eltávolítását a SUBSTITUTE függvénnyel végezzük: SUBSTITUTE([útvonal]; ""; ""). A két szöveg karaktereinek számát a LEN függvénnyel kérdezzük le. SZÓKÖZÖK SZÁMA: =LEN([útvonal]) - LEN(SUBSTITUTE([útvonal]; ""; "")).

[2] Megállapítottuk tehát az útvonalban szereplő szóközök számát. Ez a darabszám azonos a szóköz utolsó előfordulásának sorszámával. Ezt a szóközt kell lecserélnünk, egy olyan karakterre, amely biztosan nem szerepel az útvonalban. Legyen ez a karakter a felkiáltójel. MÓDOSÍTOTT ÚT-VONAL: =SUBSTITUTE([útvonal]; ""; "!"; [SZÓKÖZÖK SZÁMA]).

[3] A felkiáltójel pozíciójának sorszámát az útvonalban a FIND függvénnyel állapítjuk meg. FEL-KIÁLTÓJEL SORSZÁMA: =FIND("!"; [MÓDOSÍTOTT ÚTVONAL]).

[4] A felkiáltójel pozíciójának sorszáma azonos az útvonal karaktereinek számával a felkiáltójellel bezárólag. Ha ezt a számot kivonjuk az útvonal karaktereinek számából, akkor megkapjuk az útvonal utolsó városának nevét alkotó karakterek számát. CÉL KARAKTEREINEK SZÁMA: =LEN([útvonal]) - [FELKIÁLTÓJEL SORSZÁMA].

[5] Az útvonal utolsó városnevének karakter-hosszát ismerve már ki tudjuk íratni a városnevet is. CÉL ÁLL: =RIGHT([útvonal]; [CÉL KARAKTEREINEK SZÁMA]).

> =RIGHT([útvonal]; LEN([útvonal]) - FIND("!"; SUBSTITUTE([útvonal]; ""; "!"; LEN([útvonal] - LEN(SUBSTITUTE([útvonal]; ""; "")))))

> > 232. ábra a feladat megoldásának képlete

Az utolsó szöveg-alakító függvény a TRIM, a program KIMETSZ függvényének DAX megfelelője. A függvény eltávolítja a 32-es kódszámú szóközöket az egyetlen argumentumával megadott karakterlánc elejéről és végéről, valamint a karakterláncban egymás után álló ismétlődéseit. Utóbbit másként fogalmazva, törli a szövegben az egymás után álló szóközöket, az első kivételével. A 160-as kódú szóközöket a függvény nem veszi figyelembe. A 160-as kódú szóközöket tartalmazó karakterláncokat a SUBSTITUTE függvény segítségével tudjuk megtisztítani a felesleges szóközöktől: TRIM(SUBTITUTE([szöveg]; ""; "")). A SUBSTITU-TE függvény második, idézőjelek között álló argumentumát az ALT billentyű folyamatos nyomása mellett, a numerikus billentyűzeten, a 0160 kóddal kell "beírnunk".

Tehát a szöveg-kezelő függvényeket funkciójuk alapján két nagy csoportba sorolhatjuk. Az első csoportba tartoznak a kiinduló karakterlánc vagy karakterláncok valamely tulajdonságát vizsgálók, a második csoportba pedig a kapott karakterláncot átalakítók.

csoportosító funkció	függvény neve	függvény funkciója		
	UNICODE	karakter UNICODE számát képzi		
	LEN	szöveg karaktereinek megszámlálása		
tulajdonsag	FIND	szäveg korosása a karakterlánsban		
iekei üezese	SEARCH	szöveg kelesese a kalaktellancbali		
	EXACT	két karakterlánc összehasonlítása		
	REPT	szöveg összefűzése másolataival		
	CONCATENATE	két karakterlánc összefűzése		
	UPPER	kich at" na sub at" konvortálác		
	LOWER	Kisbetu-Hagybetu Konvertalas		
szöveg	LEFT			
átalakítása	MID	karakterianc egy darabjanak kivagasa		
	RIGHT	a karaktenane elejeroi, kozeperoi, vegeroi		
	REPLACE	czävogccoro z karaktorlánchan		
	SUBSTITUTE	SZOVEBCSELE A KALAKTELIALICDALI		
	TRIM	szóközök eltávolítása		

233. ábra szöveg-kezelő függvények rendszerezése

Ahogy látjuk, a táblázatból hiányzik CONCATENANTEX függvény, mert se ide, nem tartozik, meg se oda nem tartozik. Lényegét tekintve nem szövegkezelő függvény, de statisztikai függvénynek is csak fenntartásokkal nevezhetnénk... Ezért azután ebben a fejezetben mutattam be, de a rendszerező táblázatban már nem mertem szerepeltetni.

tábla-kezelő függvények

A SAMPLE függvénnyel teszt-rekordokat kérhetünk egy rendezett, valóságos vagy virtuális táblából. A függvény három kötelező argumentumával tudjuk deklarálni [1] a kért rekordok számát, [2] a táblát és [3] a rendező-mezőt. A függvény a rendező-mező bejegyzései alapján csökkenő sorrendet állít be. További argumentumokkal többmezős rendezést is kérhetünk, de akkor meg kell határoznunk a rendezés irányát is: [csökkenő sorrend] üres argumentum/o/FALSE/DESC, [emelkedő sorrend] 1/TRUE/ASC.



234. ábra a SAMPLE függvény fekete betűszínű kötelező és szürke betűszínű opcionális argumentumai

Többmezős rendezés esetén az utolsó rendező mező után elhagyható az üres argumentum. A függvény eredménye a kiinduló tábla a minta-rekordokra "szűrve". A minta-vétel rendje: az első és az utolsó rekord, a rendezésnek megfelelően. A további rekordok sorszáma egy számtani sorozat elemei. Az egymást követő rekordok sorszámának különbsége kerekítve: <a tábla összes rekordjának száma> / (<a minta-rekordok száma> - 1). Az ADDCOLUMNS függvénnyel virtuális számított mezőkkel bővíthetünk egy valóságos vagy virtuális táblát. A függvény első argumentuma a tábla-deklaráció, ezután az argumentumok párban állnak: név-kifejezés. Vizsgáljuk meg egy példán a függvény működését! Az elemzendő adatbázis két táblából áll: "kollégák", "munkák". A táblákat a "kolléga" mezővel kapcsoltam össze.

			dátum 🔽	kolléga 🛛 🕆 🔽	bevétel 🛛 🔽
	kolléga 🛛 🖥 🔽	bér 🔽	2018. 08. 03.	Α	580 000 Ft
1	Δ	200.000 Ft	2018. 08. 06.	Α	617 000 Ft
2	B	180 000 Ft	2018.08.06.	С	409 000 Ft
2	с С	210 000 Ft	2018.08.08.	С	434 000 Ft
5	C	210 000 11	2018. 08. 13.	Α	636 000 Ft
		6	2018.08.16	A	677.000 Ft



Ahogy a jobb oldali táblában látjuk a bevételek az egyes kollégákhoz köthetők. A munkatársak havi fizetése a "bér" mezőben álló fix összeg és az általuk produkált bevételek tíz százaléka. Készítsünk kimutatást az egyes kollégák havi fizetéséről!

kolléga	-	FIZETÉS
Α		581 900 HUF
В		308 800 HUF
С		394 600 HUF
Végösszeg		1 285 300 HUF



236. ábra a kimutatás és a FIZETÉS egyéni összesítés képlete

A SELECTCOLUMNS függvény egy virtuális táblát hoz létre az első argumentumával deklarált tábla alapján. A virtuális tábla mezőit a felhasználó határozza meg a függvény további argumentum-párosaival. A párok első elemeivel a mező nevét kell megadnunk idézőjelek között. A párosok második elemei kifejezések, amelyeket a függvény első argumentumával meghatározott tábla minden rekordjával kiértékel. A SELECTCOLUMNS függvény tehát egy új táblába helyezi a képzett mezőket, míg az ADDCOLUMNS függvény a képzett mezőkkel kiegészíti az első argumentumával deklarált táblát.

A ROW függvénnyel egyrekordos virtuális táblát hozhatunk létre. A függvény szintaktikája: ROW("név₁"; kifejezés₁; "név₂"; kifejezés₂; ...; "név_n"; kifejezés_n). A függvény működését a "bevételek" tábla segítségével mutatom be: egy boltban a vásárlók készpénzzel vagy átutalással fizethetnek, de a jó vevők "hozomra" is vásárolhatnak.

1	bevétel AZ 🛛 🔽	bevétel 🗾 🔽	jelleg 🗾 🔽
1	01	165 000 HUF	hitel
2	02	199 000 HUF	átutalás
3	03	207 000 HUF	átutalás
4	04	191 000 HUF	késznó na

237. ábra a "bevételek" tábla

Állapítsuk meg virtuális tábla segítségével a ténylegesen bevételt, azaz a hitelek nélküli bevétel öszszegét! A virtuális táblát a ROW függvénnyel hozzuk létre!

```
=SUMX(
ROW( "készpénz" ;
CALCULATE( SUM( [bevétel] ) ; 'bevételek'[jelleg] = "készpénz" ) ;
"átutalás" ;
CALCULATE( SUM( [bevétel] ) ; 'bevételek'[jelleg] = "átutalás" )) ;
[készpénz] + [átutalás] )
```

238. ábra a tényleges bevételt eredményező egyéni összesítés

A DATATABLE függvénnyel egy komplett virtuális táblát deklarálhatunk, rekordokkal együtt. Argumentum-listájának első elemei párban állnak: mezőnév idézőjelek között, és a mező adattípusának deklarációja: integer, double, string, boolean, currency, datetime (egész szám, tizedes tört szám, szöveg, logikai, pénz, dátum). A mezők meghatározását követően, kapcsos zárójelek között, a rekordok felsorolása következik. Minden rekordot, külön-külön, kapcsos zárójelekkel kell kereteznünk. A mezőbejegyzések csak konstansok lehetnek. Egyetlen kivétel a BLANK függvény, amellyel üres bejegyzést deklarálhatunk. Ugyanezt megtehetjük üres argumentummal is. A currency adattípusú mező bejegyzéseiben a pénznem-jelölőt a bővítmény nem fogadja el!

=COUNTROWS(
DATATABLE("név" ; string ;
"eredmény"; integer;
"értékelés" ; double ;
"fizetés" ; currency ;
"dátum" ; datetime ;
"tartozás" ; boolean ;
{
{ "Pali" ; BLANK() ; 2,5 ; 300000 ; "2019-11-04" ; TRUE } ;
{ "Mari" ; -4 ; 3,75 ; 400000 ; "2019-04-17 15:30" ; } ;
{ "Gabi" ; 0 ; ; 200000 ;"2019-07-07" ; FALSE }
}))

név	eredmény	értékelés	fizetés	dátum	tartozás
Pali		2,5	300000	2019-11-04 00:00	TRUE
Mari	-4	3,75	400000	2019-04-17 15:30	
Gabi	0		200000	2019-07-07 00:00	FALSE

239. ábra a virtuális tábla deklarációja és megjelenítése

A DATATABLE függvényt nem mindig jeleníti meg a bővítmény a névkezelő listájában. Ennek ellenére, ha begépeljük a képletbe a nevét, akkor a felismerés jeleként, átszínezi a betűit. A szerkesztés közbeni képlet-ellenőrzéssel is baj van: a jól megadott argumentumok is hibásnak vannak jelölve. Mindkét jelenség, sajnos, más függvényeknél is előfordul.

A SUMMARIZE függvénnyel virtuális összesítő táblát készíthetünk egy létező vagy számított tábla elemzésére. Az összesítő tábla csoportosító-, kategorizáló-, statisztikai- és segédmezőkből áll. A függvény argumentumlistájának bemutatásához használjunk egy tizenhat rekordos, "termékek" nevű táblát. Az alábbi kép a tábla forrását mutatja, amelyen színezéssel próbáltam az egyes tulajdonság-típusok megkülönböztetését segíteni.

termék AZ	név	szín	méret	minőség
01	Α	fehér	kicsi	١.
02	А	fehér	kicsi	П.
03	А	fehér	nagy	Ι.
04	Α	fehér	nagy	П.
05	А	fekete	kicsi	Ι.
06	А	fekete	kicsi	Ι.
07	А	fekete	kicsi	П.
08	А	fekete	kicsi	П.
09	Α	fekete	nagy	Ι.
10	А	fekete	nagy	l.
11	В	fehér	kicsi	Ι.
12	В	fehér	kicsi	Ι.
13	В	fehér	nagy	١.
14	В	fekete	kicsi	Ι.
15	В	fekete	nagy	Ι.
16	В	fekete	nagy	П.

240. ábra a "termékek" tábla színezett forrása

A tábla tehát két termék (A és B) három tulajdonságának (szín, méret, minőség) két-két értékét (fehérfekete, kicsi-nagy, I.-II.) tárolja. A SUMMARIZE függvény argumentumlistájának bemutatását kezdjük egy egyszerű feladat megoldásával. Hozzunk létre virtuális összesítő táblát, amely a termékek darabszámát tartalmazza, név és szín szerinti bontásban! A statisztikai mező neve DARAB legyen!

A SUMMARIZE függvény argumentumlistájának első eleme az elemzendő tábla deklarációja, majd ezt követik a csoportosító mezők nevei. A csoportosító mezők után állnak a statisztikai mezők argumentum-párosai: név-képlet. A neveket idézőjelek között kell megadnunk.

MMARIZE('termékek' ;	[név]	; [szín] ; "C	DARAB" ; CO
né	v	szín	DARAB
A	1	fehér	4
A	1	fekete	6
E	3	fehér	3
E	3	fekete	3

241. ábra

csoportosító és statisztikai mezők a megjelenített virtuális összesítő táblában

Az összesítő tábla tehát két csoportosító (név, szín) és egy statisztikai mezőből (DARAB) áll. Rekordjainak száma négy, amely a csoportosító mezők egyedi bejegyzései számának szorzata.

A csoportosító mezők tételei által meghatározott részhalmazok elemeit további mezők egyedi bejegyzései alapján osztályozhatjuk. Ezeket a kategorizáló mezőket, a csak a SUMMARIZE függvény argumentumaként alkalmazható, ROLLUP függvénnyel adhatjuk meg. A függvény argumentumai tehát mezőnevek, a felsorolás a SUMMARIZE függvény egy argumentuma, amely az argumentumlistában a statisztikai mezők deklarációja előtt áll.

A ROLLUP függvény működésének bemutatásához egészítsük ki az összesítő táblát olyan rekordokkal, amelyek az azonos nevű és színű termékek csoportjait méret szerint kategorizálják! A tábla tehát kiegészül majd a "méret" mezővel és rekordjainak száma az egyes név-szín kategoriában található egyedi méret-bejegyzések számának összegével fog növekedni.

név	szín	méret	DARAB		
Α	fehér		4		
Α	fehér	kicsi	2		
Α	fehér	nagy 2			
Α	fekete	6			
Α	fekete	ete kicsi 4			
Α	fekete	fekete nagy			
В	fehér	3			
В	fehér	kicsi 2			
В	fehér	nagy	1		
В	fekete	te			
В	fekete	ete kicsi 1			
В	fekete	nagy 2			

SUMMARIZE('termékek' ; [név] ; [szín] ; ROLLUP([méret]) ; "DARAB" ; COUNTA([termék AZ]))

242. ábra a virtuális összesítő tábla kiegészítése kategorizáló mezővel

Az ábra tanúsága szerint a kategorizáló mező átalakította a tábla szerkezetét is: hierarchiát teremtett a csoportosító és a kategorizáló mezők tételei között. A ROLLUP függvénnyel deklarált második és további kategorizáló mezők tovább strukturálják a virtuális táblát.

MMARIZE('termékek' ; [név] ; [sz	ín] ; ROLLUP	([méret] ;	[minőség]); "DARAB
név	szín	méret	minőség	DARAB
A	fehér			4
A	fehér	kicsi		
A	fehér	kicsi	l. –	1
A	fehér	kicsi	П.	1
A	fehér	nagy		2
A	fehér	nagy	н. Г.	1
A	fehér	nagy	П.	1
A	fekete			6
A	fekete	kicsi		4
A	fekete	kicsi	١.	2
A	fekete	kicsi	П.	2
A	fekete	nagy		2
A	fekete	nagy	l.	2
В	fehér			3
В	fehér	kicsi		2
В	fehér	kicsi	I.	2
в	fehér	nagy		1
В	fehér	nagy	١.	1
В	fekete			3
В	fekete	kicsi		1
В	fekete	kicsi	١.	1
В	fekete	nagy		2
В	fekete	nagy	١.	1
В	fekete	nagy	П.	1

243. ábra a virtuális összesítő tábla két kategorizáló mezővel

Az ábrát tanulmányozva azt látjuk, hogy a második kategorizáló mező nem a várakozásunknak megfelelően, a csoportosító mezők tételeit egészítette ki újabb kategoriákkal, hanem az első kategorizáló mező tételeit osztotta tovább. Ezáltal a tábla hierarchiája kiegészült a kategorizáló mezők tételeinek rangsorával. Amennyiben a második illetve további kategorizáló mezőkkel a csoportosító mezők tételeit kívánjuk tovább osztályozni, akkor a csak a SUMMARIZE függvényben alkalmazható, ROLL-UPGROUP függvénnyel kell a hierarchia-építést megakadályoznunk. A függvény argumentumai a kategorizáló mezők nevei.

Alakítsuk át az összesítő táblát úgy, hogy a minőség-kategoriák a méret-kategoriák mellett, az azonos nevű és színű csoportokon belül álljanak!

név	szín	méret	minőség	DARAB
Α	fehér			4
Α	fehér	kicsi	l.	1
Α	fehér	kicsi	П.	1
Α	fehér	nagy	- I.	1
Α	fehér	nagy	II.	1
Α	fekete			6
Α	fekete	kicsi	l.	2
Α	fekete	kicsi	П.	2
Α	fekete	nagy	l.	2
В	fehér			3
В	fehér	kicsi	l. –	2
В	fehér	nagy	l. –	1
В	fekete			3
В	fekete	kicsi	l. –	1
В	fekete	nagy	l. –	1
В	fekete	nagy	II.	1

SUMMARIZE('termékek' ; [név] ; [szín] ; ROLLUP(ROLLUPGRUP([méret] ; [minőség])) ; "DARAB" ; COUNTA([termék AZ]))

244. ábra a virtuális összesítő tábla hierarchia nélküli kategorizáló mezőkkel

Mivel a deklarált összesítő táblát nem tudjuk megjeleníteni, ezért képletünk helyességéről, csak közvetve, a tábla egyes tulajdonságainak ellenőrzésével tudunk meggyőződni. Vegyük számba ezeket a lehetőségeket! Lekérdezhetjük az összesítő tábla rekordszámát: COUNTROWS (<tábla-deklaráció>). Ellenőrizhetjük az azonos bejegyzésű rekordok darabszámát: COUNTROWS (FILTER (FILTER (FILTER (< tábla-deklaráció > ; mező₁ = érték₁) ; mező₂ = érték₂) ; mező_n = érték_n)). Meggyőződhetünk egy meghatározott rekord létrejöttéről: CONTAINS (<tábla-deklaráció> ; mező₁ ; érték₁ ; mező₂ ; érték₂... mező_n ; érték_n). Lekérdezhetjük egy mezők üres bejegyzéseinek számát: COUNT-ROWS (FILTER (<tábla-deklaráció> ; mező = BLANK ())). Utóbbi képlet eredményéből a tábla struktúrájára következtethetünk, mivel a csoportosító mezők tételeit tartalmazó rekordokban a kategorizáló mezők tételeit tartalmazó rekordokban az alacsonyabb rangú kategorizáló mezők üresek.

Az egyes rekordok a tábla struktúrájában elfoglalt helyzetéről ad információt, a csak a SUMMA-RIZE függvény argumentumaként használható, ISSUBTOTAL függvény. Egyetlen argumentuma egy mezőnév. A függvény TRUE logikai értéket ad eredményül, ha a vizsgált rekord tétele tartalmazza az argumentumával meghatározott mező kategoriáit. Az eredményt tároló mező nevét a felhasználó adja meg idézőjelek között, az ISSUBTOTAL argumentumot megelőzően. A név-ISSUBTOTAL argumentum-párosok a SUMMARIZE függvény argumentumlistájának végén állnak. A tartalmazott kategoriákat detektáló mezőt deklarálhatjuk az ISBLANK függvény is.

Egészítsük ki a 140. oldalon álló 243. ábra összesítő tábláját ellenőrző mezőkkel, amelyek detektálják a méret és a minőség mezők tartalmazott kategoriáit!

név	szín	méret	minőség	DARAB	ÜMÉR	ÜMIN
Α	fehér			4	TRUE	TRUE
Α	fehér	kicsi		2	FALSE	TRUE
Α	fehér	kicsi	l. –	1	FALSE	FALSE
Α	fehér	kicsi	П.	1	FALSE	FALSE
Α	fehér	nagy		2	FALSE	TRUE
Α	fehér	nagy	l. –	1	FALSE	FALSE
Α	fehér	nagy	П.	1	FALSE	FALSE
Α	fekete			6	TRUE	TRUE
Α	fekete	kicsi			FALSE	TRUE
Α	fekete	kicsi	l. –	2	FALSE	FALSE
Α	fekete	kicsi	П.	2	FALSE	FALSE
Α	fekete	nagy		2	FALSE	TRUE
Α	fekete	nagy	l. –	2	FALSE	FALSE
P	fehér			3		TRUE

SUMMARIZE('termékek' ; [név] ; [szín] ; ROLLUP([méret] ; [minőség]) ; "DARAB" ; COUNTA([termék AZ]) ; "ÜMÉR" ; ISSUBTOTAL([méret]) ; "ÜMIN" ; ISSUBTOTAL([minőség]))

A SUMMARIZE függvény argumentumlistája tehát a következő elemekből épül fel, sorrendben: [1] tábla-deklaráció, [2] csoportosító mezők felsorolása, [3] kategorizáló mezők felsorolása a ROLL-UP függvény argumentumaiként, [4] az egy logikai egységként kezelendő kategorizáló mezők felsorolása a ROLLUPGROUP függvénnyel, a ROLLUP argumentumaként, [5] statisztikai mezők deklarációja név-képlet argumentum-párosokkal, [6] a tartalmazott kategoriákat detektáló mezők deklarációja név-ISSUBTOTAL argumentum-párosokkal.

SUMMARIZE (
táblanév/táblakifejezés ;
mezőnév ; mezőnév ; ;
ROLLUP(mezőnév ; mezőnév ; ; ROLLUPGROUP(mezőnév ; mezőnév ; ;)) ;
"név" ; kifejezés ; "név" ; kifejezés ; ;
"név" ; ISSUBTOTAL(mezőnév) ; "név" ; ISSUBTOTAL(mezőnév) ;)

246. ábra a SUMMARIZE függvény szintaktikája

A SUMMARIZE függvénybe ágyazott X-es statisztikai függvények kiértékelése hibát eredményezhet. Ebben az esetben a GROUPBY függvénnyel kell dolgoznunk. Ez a függvény is egy virtuális összesítő táblát eredményez. Argumentumai: [1] tábla, [2] csoportosító mezők, pontosvesszővel elválasztva, [3] név-képlet argumentum-párosok, a név idézőjelek között, a képlet az X-es statisztikai függvénnyel. A képletben az X-es függvény tábla argumentuma az argumentum nélküli CURRENTGROUP függvény. Nézzünk egy egyszerű példát!



247. abra a példa háromtáblás adatbázisa

Az ábrán látható adatbázisban virtuális táblákkal megállapítjuk a megyék városainak átlagos lélekszámát, majd régiónként kiválasztjuk a legmagasabb átlagú megyéket.

^{245.} ábra összesítő tábla a tartalmazott kategoriák ellenőrzésével




A kiszámolt értékeket a MAXX függvénnyel jeleníthetjük meg. Egyéni összesítés:= MAXX(GROUP-BY(GROUPBY(...)); [MAX])

A GROUPBY függvény figyelmen kívül hagyja az üres sorokat. Amennyiben a virtuális táblában ezekre a rekordokra is szükségünk van az X-es függvény kifejezés argumentumában elágazást kell alkalmaznunk: ahol a statisztikai függvény kiértékelése üres eredményre vezet, ott a kifejezés értéke legyen egyenlő nullával.

A SUMMARIZECOLUMNS függvény lényegében a SUMMARIZE függvény javítása. Eredménye egy virtuális összesítő tábla. Alkalmazását egy háromtáblás adatbázis elemzésén mutatom be!

			e	ladás AZ	dátu	ım	r e		név	18 🔽	darab		×	egységár		
		1	01		2	017.	07. :	10.	Α				5		70	
1	dátum			nan	2	017.	07. :	11.	С				2		50	
1	2017	07	10	hátfő	2	017.	07. :	11.	Α				3		70	
1	2017.	07.	10.	kedd	2	017.	07. :	11.	Α				5		70	
2	2017.	07.	11.	sterda	2	017.	07. :	12.	С				5		50	
3	2017.	07.	12.	szerua	2	017.	07. :	12.	В				né		caír	
	2017.	07.	13.	nántok	 - 2	017.	07. :	12.	С			4		V 12 M	520 kók	
2	2017.	1.	14.	ренцек	2	017.	07. :	12.	в			1	•		NCN CÓN	
		9	09)	2	017.	07. :	13.	С			4	0		Sal	5a
		10	10)	2	017.	07. :	13.	С			2	-		Sal	ga
		11	11		2	017.	07.	13	В				_1		90	

249. ábra a "naptár", az "eladások" és az "árucikkek" táblák

Létre kell hoznunk egy virtuális összesítőtáblát, amely napok és színek szerinti bontásban tartalmazza az árbevétel összegét. A SUMMARIZE függvény fent említett hibája miatt a feladatot az ADDCOLUMNS függvénnyel kell megoldanunk: ADDCOLUMNS(SUMMARIZE('eladások' ; 'naptár'[nap] ; 'árucikkek'[szín]); "BEVÉTEL"; CALCULATE (SUMX ('eladások'; 'eladások'[darab] * 'eladások'[egységár]))). De a SUMMARIZECOLUMNS függvénnyel tovább egyszerűsíthetjük a képletet.

> SUMMARIZECOLUMNS('naptár'[nap]; 'árucikkek'[szín]; "BEVÉTEL"; SUMX('eladások'; 'eladások'[darab] * 'eladások'[egységár]))

250. ábra virtuális összesítő tábla két csoportosító és egy statisztikai mezővel

A SUMMARIZECOLUMNS függvény argumentumlistájának első elemei a csoportosító mezők. A mezőneveket táblanevükkel együtt kell megadnunk. Az argumentumlista végén a statisztikai mezők argumentum-párosai állnak: név idézőjelek között és kifejezés. Kategorizáló mezőket a ROLLUPADD-ISSUBTOTAL függvénnyel deklarálhatunk. A függvény nem csak a kategorizáló mezőket, de a részösszeg detektálására szolgáló logikai adattípusú mezőket is létrehozza az eredmény-táblában. Argumentumai párban állnak: kategorizáló mező neve, táblanévvel és a részösszeg-detektáló mező neve, idézőjelek között. A ROLLUPADDISSUBTOTAL függvény tehát a SUMARIZECOLUMNS függvény egy argumentuma, amely a csoportosító mezők deklarációját követi az argumentum-listában. A ROLLUPADDISSUBTOTAL argumentum és a statisztika mezők deklarációi között, állnak a függvény szűrő-argumentumai. Például egy vagy több FILTER függvény.

A függvény argumentum-listáját a SUMMARIZE függvénynél megismert "termékek" táblán mutatom be.

> =SUMMARIZECOLUMNS('termékek'[név]; 'termékek'[szín]; ROLLUPADDISSUBTOTAL('termékek'[méret]; "R_NSZ"; 'termékek'[minőség]; "R_NSZMÉ"); FILTER('termékek'; [méret] = "kicsi"); "DARAB"; COUNTA([termék AZ]))

> > 251. ábra a SUMMARIZECOLUMNS függvény argumentumai

Az argumentum-lista elején tehát két csoportosító mező áll. Ezt követi ROLLUPADDISSUBTOTAL argumentum két kategorizáló mezővel és a hozzájuk tartozó részösszeg detektáló mező nevével, majd a szűrőt tartalmazó argumentum következik, végül a listát a statisztikai mező argumentum párosa zárja. Az összesítő táblából hiányoznak a csak a csoportosító mezők adat nélküli tételei. Ha ezeket az üres csoportokat is szerepeltetni szeretnénk, akkor a statisztikai kifejezést, a csak a SUM-MARIZECOLUMNS függvényben alkalmazható, IGNORE függvény, egyetlen argumentumában kell szerepeltetni: ... IGNORE(SUMX('eladások'; 'eladások'[darab] * 'eladások'[egységár])).

Az ADDMISSINGITEMS függvénnyel is virtuális összesítő táblát képezhetünk, de az eredménytáblában a statisztikai értékek nélküli rekordok is szerepelnek. Argumentumai sorrendben: csoportosító mezők, tábla, kategorizáló mezők és a hozzájuk tartozó részösszeg-mezők, és egy vagy több szűrő-tábla. A kategoria mezőket a ROLLUPISSUBTOTAL függvénnyel kell deklarálnunk. Argumentumai párban állnak: <kategorizáló mező> ; "<részösszeg-mező neve>". Az argumentumlista végén álló szűrő-táblák nem kötelező argumentumok, sok-táblás adatbázisban a felesleges kapcsolódó rekordok kizárását teszik lehetővé.

A SUBSTITUTEWITHINDEX függvény két, összekapcsolt táblát dolgoz fel. Az első argumentumával deklarált több oldali táblát kiegészíti a harmadik argumentumával megadott egy oldali tábla, összetartozó rekordjainak indexével, magyarul a sorszámával. A képzett mező nevét az argumentumlista második elemeként, idézőjelek között kell megadnunk. A sorszámozás nullával kezdődik. Az argumentum-lista negyedik eleme egy kifejezés, amelyet a függvény az egy oldali tábla minden öszszetartozó rekordjában kiértékel. A rekordok a kifejezés eredménye alapján történő csökkenő rendezése adja az indexek kiosztásának rendjét. Emelkedő sorrend szerinti rendezést az ötödik, nem kötelező, argumentum ASC/1/TRUE bejegyzésével írhatunk elő. Az argumentum-lista negyedik és ötödik elemének ismétlésével többmezős rendezést is deklarálhatunk. A függvény tábla-argumentumai táblát eredményező kifejezések is lehetnek. A "több oldali táblát" ebben az esetben is az első argumentummal kell meghatároznunk.

1	név 🛛 🚡 🔽	tavasz 🛛 🔽	nyár 🔽		név	· 🛛 👘 🔽	ősz 🗾	tél 🔽
1	Éva	4	3	1	Béla	3	1	4
2	Géza	1	4	2	Géz	а	2	1
3	Béla	2	1	3	Éva		3	2
4	Léna	3	2	4	Béla	3	4	3

SUBSTITUTEWITHINDEX(tobb ; "INDEX" ; egy ; [tavasz] + [nyár] ; ASC)

252. ábra az "egy" és a "tobb" nevű táblák valamint a "tobb" táblát virtuálisan kiegészítő kifejezés

A képen látható táblák neve megegyezik a kapcsolati pozíciójukkal. A függvény negyedik argumentumában álló kifejezés az "egy" tábla rendezési kulcsa. Ha az ötödik argumentum hiányzik vagy értéke DESC/o/FALSE a rendezés csökkenő sorrendű.

név	tav + nyár	INDEX
Rála	3	0
Dela	3	
Géza	5	1
Éva	7	2
LVG	'	-

253. ábra a sorszámok kiosztása és a virtuálisan kiegészített "tobb" tábla

A NATURALINNERJOIN függvény összefűzi két tábla összetartozó rekordjait, majd a rekordokat egy virtuális táblába helyezi. A tábla tartalmazza a két tábla összes mezőjét. A függvény két argumentumával a táblákat kell deklarálnunk. A függvény csak összekapcsolt táblákat tud feldolgozni.

A NATURALLEFTOUTERJOIN függvény működése megegyezik a NATURALINNERJOIN függvényével, de az eredmény-táblába a kapcsolódó rekordokon kívül a függvény első argumentumával deklarált tábla nem kapcsolódó (több oldali tábla) illetve társtalan (egy oldali tábla) rekordjai is bekerülnek. Az argumentumok felcserélése tehát eltérő eredményhez vezethet. A két függvényt egy kéttáblás adatbázison mutatom be: "betűk" és "számok".

1	AZ 🛛 🖥 🔽	betű 🔽	Γ	I	AZ 👘 🔽	szám	
1	01	Α	:	1	04		1
2	02	В	:	2	04		2
3	03	с	:	3	05		3
4	04	D	4	4	06		4

NATURALINNERJOIN('betűk' ; 'számok')

betűk'[AZ]	[betű]	számok'[AZ]	[szám]
04	D	04	1
04	D	04	2

N

NATURALLEFTOUTERJOIN('betűk' ; 'számok')

betűk'[AZ]	[betű]	számok'[AZ]	[szám]
01	Α		
02	В		
03	С		
04	D	04	1
04	D	04	2

ATURALLEFTOUTERJOIN('számok' ; 'betűk')
----------------------	----------------------

számok'[AZ]	[szám]	betűk'[AZ]	[betű]
04	1	04	D
04	2	04	D
05	3		
06	4		

254. ábra virtuális egyesítő táblák a NATURALJOIN és a NATURALLEFTOUTERJOIN függvényekkel képezve

A GENERATE függvény virtuális táblát hoz létre két tábla egyesítésével. A függvény szintaktikája: GENERATE(tábla-deklaráció₁ ; tábla-deklaráció₂). A virtuális tábla rekordjait a függvény a következő módszerrel képzi. Veszi az első tábla első rekordját és egyenként összefűzi a másik tábla minden rekordjával, majd veszi az első tábla második rekordját és egyenként összefűzi a másik tábla minden rekordjával. És így tovább. Magyarul: mindkét tábla minden rekordja össze lesz fűzve a másik tábla minden rekordjával. A virtuális tábla mezőinek száma a két tábla mezőszámának összege, rekordjainak száma pedig a két tábla rekordszámának szorzata.

A függvény működését egy egyszerű példán mutatom be. Képzeljünk el egy színházteremet, amelynek nézőtere öt sorból áll, soronként tíz székkel. A jegyárak a színpadtól távolodva, soronként csökkenek. A sorok tulajdonságait a "sorok" tábla tartalmazza. A székek azonosítása a sorokban, balról és jobbról kezdődően, sorszámozással történik. A széleken álló három-három szék, a színpad nem tökéletes láthatósága miatt, csökkent értékű, ezért ezekért a helyekért csak a jegyár háromnegyed részét kell megfizetni. A székek tulajdonságait a "székek" tábla tartalmazza. A táblák között nincs kapcsolat. Hozzunk létre kimutatást, amely a jegybevételt összegzi sorok szerint, teltházat feltételezve!

sor		ár 🔽]	1	szék 🔽	szorzó	-		
1	1	5 000 Ft		1	B1	75	%		
2	2	4 000 Ft		2	B2	75	%		
3	3	3 000 Ft		3	B3	75	%		
4	4	2 000 Ft		4	B4	100	%		
5	5	1 000 Ft		5	B5	100	%	cor	
			2	6	J5	100	%	1	
				7	J4	100	%	2	
				8	J3	75	%	3	
				9	12	75	%	4	
				-	14		~	5	
				10	J1	75	%	Végösszeg	

255. ábra a példa két táblája és az elemzésükre készült kimutatás

A valóságos táblák egyesítésekor valóságos rekordok kerülnek összefűzésre, de ha a GENERATE függvény egyik vagy mindkét argumentuma kifejezés, akkor előfordulhat, hogy a feldolgozás alatt álló rekordhoz nincs mit hozzáfűzni, mert a másik számított tábla üres, azaz nem rendelkezik rekordokkal. Nevezzük ezeket az egyedül maradt rekordokat feldolgozhatatlan rekordoknak.

Vizsgáljuk meg a jelenséget egy egyszerű példán. A "termékek" tábla három terméket és azonosítójukat tartalmazza, a "színek" táblában pedig az azonosítójukkal jelölt termékek gyártásban lévő színei állnak. A táblákat a "termék AZ" mezővel kapcsoltam össze.

1	termék AZ 🛛 🖥 🔽	termék 🔽
1	01	Α
2	02	В
3	03	С

1	szín AZ 🛛 🔽	termék AZ 🛛 🐕 🔽	szín 🔽
1	01	01	fekete
2	02	01	fehér
3	03	02	sárga
4	04	01	sárga
5	05	02	fekete

256. ábra a példa "termékek" és "színek" táblája

Az ábrát vizsgálva látjuk, hogy az "A" terméket három színben (fekete, fehér és sárga), a "B" terméket két színben (sárga, fekete) gyártják. A "C" termék jelenleg nincs gyártásban. Szeretnék egy olyan virtuális táblát, amely az összes előforduló termék-szín variáció összefűzött rekordjait tartalmazza! Készítsünk egyéni összesítést, "G" néven, amely megszámlálja a GENERATE függvénnyel létrehozott virtuális tábla rekordjait! Készítsünk egyéni összesítést az előző képlet másolásával, "GALL" néven, amelyben a GENERATE függvény helyett a GENERATEALL függvényt alkalmazzuk!

termék AZ	termék	szín AZ	termék AZ	szín
01	А	01	01	fekete
01	А	02	01	fehér
01	А	04	01	sárga
02	В	03	02	sárga
02	В	05	02	fekete
03	С			



Az ábra tanúsága szerint a feldolgozhatatlan rekordot csak a GENERATEALL függvénnyel létrehozott virtuális tábla tartalmazza, a GENERATE függvénnyel képzett nem. Ellenőrizzük a virtuális tábla feldolgozhatatlan rekordjának meglétét!

=CONTAINS(
GENERATEALL('termékek' ; CALCULATETABLE('színek')) ;
'termékek'[termék AZ]; "03";
[termék]; "C";
[szín AZ]; BLANK();
'színek'[termék AZ]; BLANK();
[szín]; BLANK())

258. ábra a feldolgozhatatlan rekord meglétének ellenőrzése

A képlet TRUE értéket ad eredményül. A "termékek" és a "színek" táblák egyesítése két azonos nevű mezőt eredményezett a virtuális táblában: "termék AZ". A névazonosság azonban nem okoz zavart, mert a bővítmény nyilvántartja a mezők származását. Ha tehát az egyik, duplikált nevű mezőre hivatkozunk a képletben, akkor adjuk meg a mezőt tartalmazó tábla nevét is, ahogy ezt a fenti ábra képletében is látjuk.

A CROSSJOIN függvény funkciója azonos a GENERATE függvényével, de az egyesítendő táblák száma nincs korlátozva. A függvény szintaktikája: CROSSJOIN(tábla-deklaráció₁; tábla-deklaráció₂; ... ; tábla-deklaráció_n).

A függvény működését egy elképzelt polcrendszer adatait tartalmazó adatbázison mutatom be. A polcok különböző hosszúságban, szélességben és vastagságban rendelhetők. A rendelkezésre álló méreteket, centiméterben, a "hosszak", a "szélességek" és a "vastagságok" nevű táblák tartalmazzák. A táblák között nincs kapcsolat.

	hossz AZ 🛛 🔽	hossz 💽	👔 szélesség AZ 💽	szélesség 🔽	Γ	vastagság AZ 💽	vastagság 🔽
1	01	70	1 01	14	-	1 01	3
2	02	80	2 02	17		2 02	5
3	03	90	3 03	20			

259. ábra

a példa táblái: "hosszak", "szélességek", "vastagságok"

Készítsünk kimutatást a választékról! A Pivot-tábláról legyen leolvasható az egyes polcok mérete és ára. Egy köbméter fa ötven-ezer Forintba kerül.

hossz	🔻 szélesség	👻 vastagság 💌	ár
70	14	3	147 Ft
70	14	5	245 Ft
70	17	3	179 Ft
70	17	5	298 Ft
70	20	3	210 Ft
70	20	5	350 Ft
80	14	3	168 Ft
80	14	5	280 Ft
80	17	3	204 Ft

ár:=SUMX(CROSSJOIN(hosszak ; 'szélességek' ; 'vastagságok') ; [hossz] * [szélesség] * [vastagság] * 50000 / 1000000)



A képen látható Pivot-tábla legfontosabb megjelenítési tulajdonságai a következők: [1] tagolt kimutatás-elrendezés, [2] rész- és végösszegek megjelenítésének tiltása, [3] kibontó/összecsukó gombok megjelenítésének tiltása, [4] tételcímkék táblázatos megjelenítése, [5] tételcímkék ismétlése. A kimutatás áttekinthetőségét egyéni szegélyezéssel javítottam.

A CROSSJOIN függvénnyel létrehozott virtuális táblában nem szerepelnek a feldolgozatlan rekordok. Ha szükségünk van rájuk akkor a CROSSJOIN helyett alkalmazzunk egymásba ágyazott GE-NERATEALL függvényeket. Ezt a lehetőséget mutatom be a következő példán.

[]	betű AZ 🛛 🖥 🔽	betű 🔽	[1	szám AZ 🛛 🖥 🔽	betű AZ 🛛 🥵 🔽	szám 🔽]	7	szín AZ 🛛 🔽	szám AZ 🖷 🔽	szín 🔽
1	01	Α		1	01	01	1		1	01	01	fehér
2	02	В		2	02	02	2		2	02	02	fekete
З	03	С		3	03	01	3		3	03	03	barna
		,						9	4	04	01	kék

261. ábra a példa három táblája: "betűk", "számok", "színek"

Virtuális táblában egyesítsük a betűk tábla rekordjait a számok tábla minden kapcsolódó rekordjával, majd az így képzett rekordokat fűzzük össze a színek tábla sárga rekordjaival! Az eredménytáblába kerüljenek bele a feldolgozhatatlan rekordok is!

A megoldás előtt vizsgáljuk meg a táblákat, hány rekordra számíthatunk az eredményben. Az "A" betűnek kettő, a "B" betűnek egy kapcsolódó rekordja áll a "számok" táblában. A "C" betűnek nincs kapcsolódó rekordja. Az első két tábla feldolgozása tehát négy rekordot eredményez, de ez a végeredmény is, mivel a "színek" táblának nincs "sárga" rekordja.

A CROSSJOIN függvénnyel képzett egyesítő tábla nem tartalmaz feldolgozhatatlan rekordokat, erről a COUNTROWS függvénnyel győződhetünk meg. CJ:=COUNTROWS (CROSSJOIN('betűk'; RELATEDTABLE('számok'); FILTER('színek2'; [szín] = "sárga"))) → (üres). Tehát a GENERATEALL függvénnyel kell próbálkoznunk. GALL2:=COUNTROWS (GENERATEALL(GENERATEALL('betűk'; RELATEDTABLE('számok')); FILTER('színek'; [szín] = "sárga"))) → 4.

betű AZ	betű	szám AZ	betű AZ	szám	szín AZ	szám AZ	szín
01	Α	01	01	1			
01	Α	03	01	3			
02	В	02	02	2			
03	С						

262. ábra az egymásba ágyazott GENERATEALL függvényekkel létrehozott virtuális tábla megjelenítése

Az UNION függvény azonos felépítésű táblák vagy azonos felépítésű táblákat eredményező kifejezések virtuális egyesítésére szolgál. A függvény eredménye az argumentum-lista első elemével deklarált virtuális tábla, amely tartalmazza a többi tábla rekordjait is. Másként fogalmazva, a függvény az első argumentumával meghatározott táblába helyezi el a további argumentumaival meghatározott táblák rekordjait is. A függvényt egy háromtáblás adatbázison mutatom be.

1	tantárgy 🔽	létszám 🔽	1	kurzus 💽	fő 🔽	1	képzés 💽	diák	
1	filozófia	25	1	magyar	24	1	etológia		17
2	esztétika	18	2	angol	15				

263. ábra a példa három táblája: "első", "második" és "harmadik"

Adjuk össze a három tábla diákjait! A legegyszerűbb megoldást választjuk: a három tábla rekordjait az UNION függvénnyel egyetlen virtuális táblába helyezzük el, majd összeadjuk a létszám-adatokat.

vizsgázók	=SUMX(UNION('első' ; 'második' ; harmadik) ; [létszám])
VIZSgazok	-SOMA(ONION(EISO ; Masodik ; Manhadik); [letszam]

tantárgy	létszám
filozófia	25
esztétika	18
magyar	24
angol	15
etológia	17

264. ábra

az egyéni összesítés képlete és az UNION függvénnyel létrehozott virtuális tábla megjelenítése

Ahogy látjuk az UNION függvény számára az eltérő mezőnevek nem okoznak problémát, de a táblák mezőszámának és a mezők adattípusának már azonosnak kell lenniük. Ez a megállapítás a többi azonos felépítésű táblákkal dolgozó DAX függvényre is igaz.

Az INTERSECT és az EXCEPT függvények két azonos felépítésű táblát vizsgálnak. Két argumentumuk a két tábla. Az INTERSECT függvény az első argumentumával meghatározott tábla rekordjait adja vissza, amelyek szerepelnek a második argumentumával megadott táblában. Az EXCEPT függvény az első argumentumával meghatározott tábla rekordjait adja vissza, amelyek nem (!) szerepelnek a második argumentumával megadott táblában. Magyarul a két függvény meghatározása azonos kivéve a nem szócskát. Van két táblánk: "A" és "B".

/ szín	×	alak 💌	1	szín	akak			
1 sárga		kör	1	sárga	kör			
2 kék		kör	2	kék	kör			
3 zöld		kör	3	barna	négyzet	-00	INTROWS(IN	TERSE
4 sárga		háromszög				=COL	JNTROWS(IN	TERSE
5 kék		háromszög				=COL	JNTROWS(EX	CEPT
6 zöld		háromszög				=COL	JNTROWS(EX	CEPT

265. ábra az INTERSECT és az EXCEPT függvények bemutatása

tábla-kezelő függvények rendszerezése

Az előző fejezetben megismert függvények tulajdonságainak összehasonlítása előtt vezessünk be néhány új fogalmat és fogalmazzunk meg néhány általános, az egész függvény-családra érvényes megállapítást.

A tábla-kezelő függvények egy vagy több valóságos vagy virtuális táblával dolgoznak. Ezeket az adat-feldolgozás sémája szerinti bemeneti vagy magyarul feldolgozandó táblákat, nevezzük a függvény alap-tábláinak! A családba tartozó összes függvény kimenete, másszóval eredménye, egy virtuális tábla. Ezt a memóriában létrehozott adat-szerkezetet nevezzük eredmény-táblának!

Három függvény alap-táblái azonos felépítésű táblák. Ezt a fogalmat a programban az azonos nevű oszlopokat azonos sorrendben tartalmazó táblázatok megnevezésére hoztuk létre, de a bővítményben, ahogy ezt az előző fejezet végén már rögzítettük, nem a mezőneveknek, hanem a mezők számának és az azonos sorszámú mezők adattípusának kell azonosnak lennie.

Az eredmény-tábla, az egyes függvények funkciója szerint, lehet [1] az alap-tábla mintarekordokkal, [2] a kiegészített alap-tábla, [3] az alap-táblák egyesítése, [4] egy új tábla [5] csoportosító illetve kategorizáló mezők tételei által strukturált összesítő tábla.

a függvény neve	alap-táblái	eredmény-táblája		
SAMPLE	ogy tábla	alap-tábla a minta-rekordokkal		
ADDCOLUMNS	egy tabla	alap-tábla		
SUBSTITUTEWITHINDEX	két összekapcsolt tábla	kiegészítve az új mezőkkel		
UNION	meghatározatlan számú, azonos felépítésű			
INTERSECT	két azonos	az első alap-tábla a kiválogatott rekordokkal		
EXCEPT	felépítésű tábla	a kivalogatott rekordokkai		
NATURALINNERJOIN				
NATURALLEFTOUTERJOIN				
GENERATE	Ket tabla	új tábla a táblák mezőivel az összefűzött rekordokkal		
GENERATEALL				
CROSSJOIN	meghatározatlan számú tábla			
SELECTCOLUMNS	egy tábla	<i></i>		
ROW	meghatározatlan számú tábla	új tábla a képzett rekordokkal		
DATATABLE	nincs alap-táblája			
SUMMARIZE				
GROUPBY	maghatározatlan czámú tábla	öcczocít ž táblo		
SUMMARIZECOLUMNS	megnatarozatian szamu tabla	osszesítő tábla		
ADDMISSINGITEMS				

266. ábra

a tábla-kezelő függvények rendszerezése alap- és eredmény-táblájuk szerint

A tábla-kezelő függvények családjába tartoznak azok a függvények is, amelyek csak meghatározott függvénybe ágyazottan alkalmazhatók. Az előbbiek legyenek a szolga-, az utóbbiak a gazda-függvények. Tekintsük át őket!

gazda-függvény	szolga-függvény	szolga-függvény funkciója		
GROUPBY	CURRENTGROUP	tábla-argumentum helyettesítésére		
SUMMARIZE	ROLLUP	kategorizál		
	ROLLUPGROUP	kategorizál, de nem épít hierarchiát		
	ISSUBTOTAL	részösszeget detektál		
SUMMARIZECOLUMNS	ROLLUPADDISSUBTOTAL	kategorizál		
	IGNORE	megjeleníti a statisztikai érték nélküli csoportokat		

267. ábra a gazda- és szolgafüggvények rendszerezése

Az összesítő táblát eredményező függvények tárgyalása során tapasztalhattuk, hogy a DAX még távolról sem tökéletes. Gondoljuk csak a javítás, javításának a javítására (SUMMARIZE, SUMMARIZE-COLUMNS, ADDMISSINGITEMS).

függvények előfordulása

A

ABS 72, 90, 96 ACOS 99 ACOSH 99 ACOT 99 ACOTH 99 ADDCOLUMNS 126, 137, 143, 150 ADDMISSINGITEMS 144, 150, ALL 52, 110, 111, 114, 128, 129, 130, ALLEXCEPT 110, 111, 112, 120, 121, 129, 130 ALLNOBLANKROW 108, 109, 110, 111 ALLSELECTED 108, 109, 110, 111, 112 AND 94, 95, 98, ASIN 99 ASINH 99 ATAN 99 ATANH 99 AVERAGE 49,67, 125, 126 **AVERAGEA 126** AVERAGEX 125, 143

В

BETA.DIST 99 BETA.INV 99 BLANK 96, 97, 132, 138, 141, 147

С

CALCULATE 63, 64, 65, 67, 111, 112, 114, 118, 119, 120, 122, 124, 125, 129, 130, 131, 138, 143 CALCULATETABLE 122, 124, 129, 147 **CALENDAR** 79 **CALENDARAUTO 79 CEILING** 96, 97, 98 CHISQ.DIST 99 CHISQ.DIST.RT 99 CHISQ.INV 99 CHISO.INV.RT 99 **CLOSINGBALANCEMONTH 81 CLOSINGBALANCEQUARTER 81 CLOSINGBALANCEYEAR 81, 84 COMBIN 99 COMBINA 99** CONCATENATE 77, 133, 134, 136 **CONCATENATEX 133 CONFIDENCE.NORM 99 CONFIDENCE.T 99** CONTAINS 92, 93, 94, 141, 147 CONTAINSROW 117, 118 **COS** 99 **COSH** 99 **COT** 99

COTH 99 COUNT 111, 118, 119, 125, 126 COUNTA 49, 65, 114, 122, 126, 144 COUNTAX COUNTBLANK 126 COUNTROWS 46, 87, 92, 109, 110, 115, 116, 117, 118, 124, 126, 130, 138, 141, 148 COUNTX 125 CROSSFILTER 121, 122 CROSSJOIN 147, 148, 150 CURRENCY 99 CURRENTGROUP 142, 143, 150 CUSTOMDATA

D

DATATABLE 138, 150 DATE 71, 72, 77, 78 DATEADD 84, 85, 86, 87, 89 DATEDIFF 72 **DATESBETWEEN 88, 89 DATESINPERIOD** 87, 88, 89 **DATESMTD 85, 89 DATESQTD 85, 89 DATESYTD** 85, 89 DATEVALUE 23, 72, 99 DAY 71, 72, 77 DEGREES 99 DETAILROWS DISTINCT 108, 109, 110 DISTINCTCOUNT 49, 65 **DIVIDE 97**

Ε

EARLIER 54, 55, 63, 119, 121, 124, 125, 129, 130 EARLIEST 125 EDATE 71, 72 ENDOFMONTH 82, 88 ENDOFQUARTER 82, 88 ENDOFYEAR 82, 88 EOMONTH 71, 72 EVEN 98 EXACT 131, 132, 136 EXCEPT 149, 150 EXP 99 EXPON.DIST 99

F

FACT 99 FALSE 94 FILTER 53, 54, 64, 65, 92, 116, 117, 118, 121. 122, 129, 130, 141, 144, 148 FILTERS 108, 109, 110, 115 FIND 132, 133, 134, 135, 136 FIRSTDATE 82, 83, 88 FIRSTNONBLANK 123, 129 FIXED 99 FLOOR 97, 98 FORMAT 76, 77, 99, 100, 101, 102, 103, 114, 116, 126

G

GCD 97 GENERATE 145, 145, 147, 150 GENERATEALL 146, 147, 148, 150 GEOMEAN 125 GEOMEANX 125 GROUPBY 142, 143, 150

Н

HASONEFILTER 115 HASONEVALUE 87, 115, 116 HOUR 90

Ι

IF 23, 63, 73, 74, 77, 78, 87, 91, 92, 93, 95, 96, 113, 114, 116, 129, 130, 132, 133 **IFERROR** 96, 132 **IGNORE** 144, 150 INT 23, 97, 98 **INTERSECT** 149, 150 ISBLANK 92, 94, 141 **ISCROSSFILTERED** 113, 114 **ISEMPTY** 92, 94 ISERROR 23, 91, 94, 96, 133 **ISEVEN** 91, 94 **ISFILTERED** 113, 114 **ISLOGICAL** 91, 94 **ISNONTEXT** 91, 94 **ISNUMBER** 91, 94 **ISO.CEILING** 97, 98 **ISODD** 91, 94 **ISONORAFTER** 93, 94 **ISSUBTOTAL** 141, 142, 150 **ISTEXT** 91, 94

Κ

KEEPFILTERS 120 KEYWORDMATCH 94

L

LASTDATE 82, 83, 88, 124 LASTNONBLANK 123, 124 LCM 97 LEFT 94, 134, 135, 136 LEN 131, 135, 136 LN 99 LOG 99 LOG10 99 LOOKUPVALUE 105, 122, 123 LOWER 134, 136

Μ

MAX 49, 125, 126, 129, 143 MAXA 126 MAXX 125, 143 MEDIAN 125 MEDIANX 125 MID 128, 134, 136 MIN 49, 125, 126 MINA 126 MINUTE 90, 91 MINX 125 MOD 97, 116, 117, 118 MONTH 71, 72, 77 MROUND 97, 98

Ν

NATURALINNERJOIN 145, 150 NATURALLEFTOUTERJOIN 145, 150 NEXTDAY 83, 88, 89 NEXTMONTH 84, 89 NEXTQUARTER 84, 89 NEXTYEAR 84, 89 NOT 23, 92, 94 NOW 90

0

ODD 98 OPENINGBALANCEMONTH 81, 88 OPENINGBALANCEQUARTER 81, 88 OPENINGBALANCEYEAR 81, 82, 88 OR 94, 95

Ρ

PARALLELPERIOD 85, 89 PATH 104, 105, 106, 129 PATHCONTAINS 105 PATHITEM 105, 106 PATHITEM REVERSE 105 PATHLENGTH 105 PERCENTILE.EXC 125 PERCENTILE.INC 125 PERCENTILEX.EXC 125 PERCENTILEX.INC 125 PERCENTILEX.INC 125 PERMUT 99 PI 97 POISSON.DIST 99 POWER 97 PREVIOUSDAY 83, 88, 89

PREVIOUSMONTH 84, 89 PREVIOUSQUARTER 84, 89 PREVIOUSYEAR 84, 89

Q

QUOTIENT 53, 97

R

RADIANS 99 RAND 98 **RANDBETWEEN 98** RANK.EQ 126, 127, 128 RANKX 126, 127, 128, 129, 130 RELATED 46, 63, 65, 117, 118, 124 **RELATEDTABLE** 46, 124, 137, 148 **REPLACE** 134, 136 **REPT** 133, 136 RIGHT 81, 103, 134, 135, 136 ROLLUP 139, 140, 142, 150 ROLLUPADDISSUBTOTAL 143, 144, 150 ROLLUPGROUP 140, 142, 150 **ROLLUPISSUBTOTAL 144** ROUND 50, 52, 67, 74, 97, 98, 126 ROUNDDOWN 97, 98 ROUNDUP 74, 77, 86, 97, 98 ROW 126, 137, 138, 150

S

SAMEPERIODLASTYEAR 87 SAMPLE 136, 150 SEARCH 73, 132, 136 SECOND 90 SELECTCOLUMNS 137, 150 **SIGN** 90, 96 **SIN** 99 **SINH 99** SQRT 97 STARTOFMONTH 82,88 **STARTOFQUARTER 82,88 STARTOFYEAR 82, 88** STDEV.P 49, 125 STDEV.S 49, 125 STDEVX.P 125 STDEVX.S 125 SUBSTITUTE 134, 135, 136 SUBSTITUTEWITHINDEX 144, 150 SUM 49, 50, 54, 55, 63, 64, 81, 82, 90, 91, 112, 116, 120, 125, 129, 130, 131, 138 SUMMARIZE 126, 133, 138, 139, 140, 141, 142, 143, 144, 150 **SUMMARIZECOLUMNS** 143, 144, 150 SUMX 52, 53, 54, 125, 130, 131, 137, 138, 143, 144 SWITCH 74, 77, 95, 96, 116

Т

TAN 99 TANH 99 TIME 91 TIMEVALUE 23, 91, 99 TODAY 53, 71, 72 TOPN 126, 128, 130, 131 TOTALMTD 81, 88, 89 TOTALQTD 81, 82, 88, 89 TOTALYTD 81, 88, 89 TOTALYTD 81, 88, 89 TRIM 135, 136 TRUE 94 TRUNC 91, 97, 98, 116, 117, 118

U

UNICODE 131 UNION 148, 149, 150 UPPER 134, 136 USERELATIONSHIP 119, 120 USERNAME USERPRINCIPALNAME

V

VALUE 103 VALUES 87, 108, 109, 110, 115, 116 VAR.P 49, 125 VAR.S 49, 125 VARX.P 125 VARX.S 125

W

WEEKDAY 71, 72, 77, 78 WEEKNUM 71, 72, 77, 78

X XIRR

XNPV

Υ

YEAR 71, 72, 77, 78 YEARFRAC 72