



Kardos Zoltán

## Adatbázis kezelési alapismeretek



A követelménymodul megnevezése:

Ügyviteli, irodatechnikai, kommunikációs ismeretek

A követelménymodul száma: 0061-06 A tartalomlelem azonosító száma és célcsoportja: SzT-010-50



## ADATBÁZIS KEZELÉSI ALAPISMERETEK

### ESETFELVETÉS – MUNKAHELYZET

Ön ambiciózus programozó, informatikai szakember. Azt a feladatot kapja, hogy egy induló italdiszkont raktár nyilvántartását oldja meg adatbázis kezelő program segítségével. A programnak tudnia kell kezelni a raktárban található termék állományt, a megrendeléseket, üzleti partnereket, és a privát ügyfélkört. Ezen felül a program képes legyen az állományokból volument számolni, valamint a program bővíthető legyen, más profilok, termékkörök későbbiekben történő kezelésére. A program alkalmas legyen különféle legűjtések elkészítésére is.

### SZAKMAI INFORMÁCIÓTARTALOM

#### Az adatbázis fogalma:

Az adatbázis információk gyűjtésére, rendszerezésére szolgáló eszköz, mely arra is alkalmas, hogy különféle szempontok szerinti rendezéssel tudjunk adatokat, információkra szert tenni. Adatok tárolását, rendezését oldhatjuk meg vele, ez szólhat személyekről, árukészletről, filmek, zenék, számlázási és készlet nyilvántartási adatok, stb nyilvántartásáról, rendezéséről.

Az adatbázis tágabb értelemben egy olyan adathalmaz, melynek elemei egy meghatározott tulajdonságuk révén összetartozóak.

Egy Access-adatbázis is egyetlen fájlban tárolja az adatokat, valamint az ezekhez kapcsolódó különböző objektumokat (például űrlapokat, jelentéseket, makrókat stb).

Egy kis történeti áttekintés, az adatbázis kezelés nem új keletű fogalom, jelenléte végig vonul a történelmi korokon, egészen az ókori társadalmaktól (népszámlálásból származó adatok, terménymennyiségek, adók) napjainkig, ugyanakkor vitathatatlan tény, hogy a számítógépek térnyerése jelentős lendületet adott és addig még nem ismert szintre emelte az adatbázis kezelés alkalmazásait. A papírt felváltotta az elektronikus elven működő adat nyilvántartás.

A régebbi rendszerek nagy hátránya volt, hogy az adathozzáférés nehézkes volt, csak egyféle szempont szerint lehetett adatokat „lekérdezni” és persze nyilvántartani. Lásd például a könyvtárak katalógusait, ahol külön katalógusokat kellett készíteni a cím szerinti, és külön katalógust szerző szerinti nyilvántartás kezelése érdekében.

A számítástechnika hőskorában a jól ismert lyukkártya olvasó és író, majd mágnesszalagos egységek csatlakoztak a rendszerhez. Mágnesszalagokon már nagy mennyiségű adatot tudunk tárolni, legfőbb hátránya a rendszernek a soros hozzáférés volt, ami azt jelentette, hogy egy információ eléréséhez végig kellett olvasni az összes megelőző adatot is a szalagon.

A PC-k megjelenésével és elterjedésével azonban tarthatatlanná vált a régebbi szemlélet, miszerint az adatbázis kezelőknek számítástechnikai szakembernek kell lenniük, a felhasználók nagy száma miatt már nem lehetett elvárni, gyakorlatilag ebből az elgondolásból kiindulva készítették el a nagy szoftverházak a mindenki számára komfortosan kezelhető adatbázis rendszerüket.

### Az adatbázis kezelés alapfogalmai:

Az adatbázis a felhasznált adatok és a közöttük fennálló kapcsolatok, rendszere, nagyon fontos, hogy az adatbázisunk szerkezetét jól megtervezzük, mivel ennek következménye lesz az, hogy a későbbiekben milyen legűjtéseket, szűréseket tudunk elvégezni benne.

A tábla a logikailag összetartozó adatokat tartalmaz, tábla (táblázat) oszlopokból és sorokból épül fel, amely mezőknek, illetve rekordoknak nevezünk.

A rekord az adatbázis egy adatlánc mentén összefüggő sora, itt és így tároljuk az egymással összefüggő adatokat (pl. személy neve, neme, lakcíme, születési helye és ideje stb).

A személyt esetenként egyenek is nevezünk. Az egyed az, amelynek az adatait tároljuk és gyűjtjük az adatbázisban.

A mező az adatbázis egy oszlopa, amelyben az egyedek tulajdonságértékeit tárolja.

Az attribútum vagy tulajdonság az egyed valamelyik jellemző tulajdonsága. (Egy személy egy jellemző tulajdonsága lehet például a neve is.)

Egy egyedre vonatkozó tulajdonságok összességét egyedtípusnak nevezünk.

( például a név, életkor, szeme színe együtt).

Elsődleges kulcs: a tábla rekordjainak egyértelmű és egyben legfontosabb azonosítója, értéke egyedi.

Idgen kulcs: egy olyan azonosító vagy hivatkozás, melynek segítségével egy másik tábla másik, jellemző kulcsára hivatkozhatunk

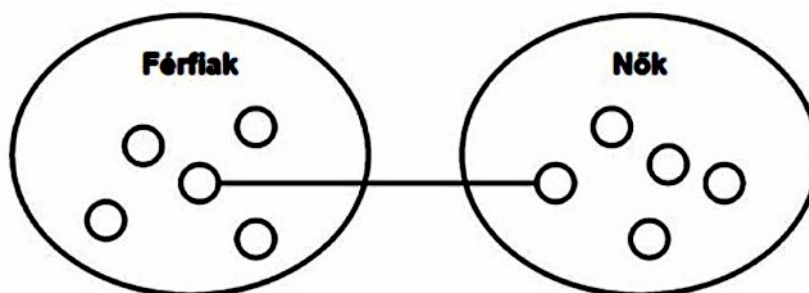
Index: Egy előre megadott paraméter sor, mely a táblákban való keresés és a sorba rendezés gyorsítására alkalmas eszköz.

Az Accessben adott a lehetőség, hogy több mező értékeiből készítsünk indexet.

### Az adatbázis táblái közti kapcsolatok

A táblák közti kapcsolatok az egyedek egymáshoz való viszonyát írják le.

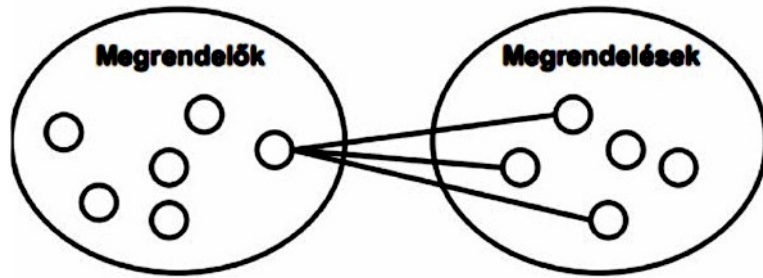
Egy-egy (1:1) (1. ábra) kapcsolat: az egyik tábla egy eleméhez a másik tábla pontosan egy és mindig csak eleme kapcsolódik. Ezt a kapcsolattípust használjuk többek között házastársak nyilvántartására.



1. ábra. Kapcsolatok 1:1

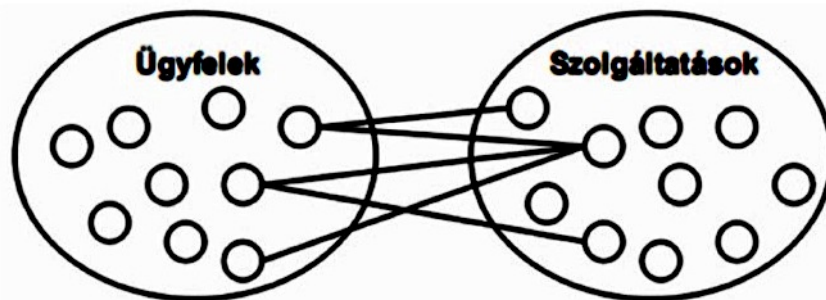
Egy-több (1:N) (2. ábra) kapcsolat: az egyik tábla egy eleméhez a másik tábla több eleme is tartozhat, egy egyed előfordulása egy vagy több egyed előfordulásához van rendelve. Az egyik tábla egy rekordjához, egy másik tábla több rekordja tartozik, ezt a kapcsolattípust alkalmazzuk pl. boltok és szállítók közti adatok nyilvántartásakor.





2. ábra. Kapcsolatok 1:N

Több-több (N:M) (3. ábra) kapcsolat: bármely tábla elemeihez a másik tábla tetszőleges számú eleme tartozhat. Adott egyed egy vagy több előfordulása kapcsolatban áll egy vagy több előfordulásával. Ebben az esetben a kapcsolatot egy harmadik tábla révén tudjuk képezni, ezt nevezzük illesztő táblának. Az illesztő táblába fel kell venni mindkét tábla elsődleges kulcs mezőjét, így megteremtve a kapcsolat leképezésének lehetőségét. Ezt a kapcsolattípust alkalmazzuk pl. egy cég ügyfelei és az ügyfélnek nyújtott szolgáltatások összerendelésére.



3. ábra. Kapcsolatok N:N

A redundancia az adatismétlés többszörös (esetünkben felesleges) alkalmazását jelenti. Például a dolgozók nevét, címét és telefonszámát külön tároljuk például a személyes adatoknál is és az iskolai végzettséget tartalmazó táblánál is. Ezáltal több tároló kapacitás kerül lekötésre, valamint esetlegesen lassabbá válik az adatbázis kezelése.

Ennél jelentősebb probléma az adatok frissítésekor keletkezik abban az esetben, ha az említett adatokat nem minden táblában (megjelenésben) frissítjük.

Ez jelentősen növeli a hibalehetőségeket.

A redundancia (adattöbbszörözés) kiküszöbölésére lehetőséget ad a normalizálás. Normalizálással csökkenthető az adatbázisfájl mérete, az adatbázis tartalma logikailag áttekinthetőbbé válik.

### Adatmodellek

Az adatmodell egyértelműen meghatározza az adatbázis felépítését, az adatok típusát, ezek kapcsolatát.

Napjainkban négyféle adatmodellt ismerünk, használunk: a hierarchikus, a hálós, az objektum orientált és a reláció adatmodellt.

Az adatmodell szerkezeti és műveleti részből tevődik össze, feladata, hogy a körülöttünk zajló világban történő eseményekről, dolgokról számítógéppel könnyen modellezhető szerkezetet (adatbázist) adjon, valamint a későbbiekben felmerülő szempontok szerint bennük keresni lekérdezni tudjunk.

A hierarchikus adatmodellt fa szerkezettel, irányított gráfként\* szemléltetjük, egyszerűen alkalmazható. Ennek az ábrázolásnak az a hátránya, hogy nem minden kapcsolatot tudunk vele szemléltetni.

(például az n az m kapcsolatot nem tudja kezelni).

\*gráf a csomópontok, a csúcsok és a rajtuk értelmezett összeköttetések élek halmaza.

A hálós adatmodell: az adatokat mutatókon keresztül tudjuk elérni és ugyanígy tudjuk bemutatni a köztük lévő kapcsolatokat is. Ez az adatmodell kiküszöböli a hierarchikus adatmodell hátrányát, nevezetesen azt, hogy nem minden kapcsolatot lehet vele leképezni

Objektum orientált adatmodell, olyan adatbázis, amely „intelligens” részekből épül fel, ez azt jelenti, hogy az adatbázis egyes objektumaik képesek „felismerni”, hogy mire használhatóak és azt is, hogy miképpen kapcsolódnak a többi adatbázis eleméhez.

Ez az adatmodell a kezdeti nehézségek ellenére jól fejlődik, bár nem mondható el, hogy konkrétan jelentős szerepet tölt be napjaink adatbázis kezelő protokollok között. Mondhatjuk úgy is, hogy ez egyenlőre a jövő halk zenéje. Hatékonyságában még alulmarad pl. a relációs adatbázisokkal szemben.

### Relációs adatmodell:

A 70-es években látott napvilágot napjainkban már szinte minden rendszer ezt támogatja (lásd MS ACCESS). Jelölésrendszere táblázatokból áll. A kapcsolatok leírására matematikai relációkat használ. Ezt a modellt használja például az SQL programnyelv is.

Relációnak nevezzük az adatelemek megnevezett, összetartozó csoportjából kialakított táblázatot, sorokból és oszlopokból áll, az oszlopok egy-egy tulajdonságot írnak le. A relációs adat modell egy kétdimenziós táblázat.

A relációs adatmodell egy olyan adatmodell, amelynek legfontosabb eleme a matematikai reláció fogalma. Ez az adatmodell is definiálja a működésében jellemző adatszerkezeteket, valamint azokat a műveleteket, amelyek vele kapcsolatban értelmezettek.

A relációs adatmodell csak egy logikai adatbázis kereteit definiálja, nem kompetenciája az adatok tényleges tárolásának „megszervezése” a folyamatok gyakorlati lebonyolítása.

Egy adatmodell akkor relációs adatmodell, ha azt a halmazok Descartes szorzatával lehet definiálni.

Relációs adatbázis az olyan elméleti adatbázis, amely több összekapcsolt, egymásra épített táblázatból áll.

Az alábbi táblázat (4. ábra) egy kapcsolattal rendelkező relációt mutat, ahol a kapcsolat az „azonosító” oszlop értékei.

**Dolgozók**

Azonosító	Név	Kategória	Születési dátum	Születési hely	Fizetés
101	Kis Mariann	A	1951-12-21	Piripócs	23.345-
→ 102	Kis Ferenc	C	1971-01-11	Piripócs	43.555-
→ 103	Tóth Aranka	C	1946-06-26	Messzifalva	48.252-
→ 104	Pék Jenő	F	1972-02-03	Messzifalva	63.290-
→ 105	Pék Tihamér	E	1974-04-24	Messzifalva	74.432-

**Szabadság**

Azonosító	Dátumtól	Dátumig	Munkanap összesen	Fizetett
→ 104	1999-06-30	1999-07-05	3	Igen
→ 102	1999-07-12	1999-07-16	5	Nem
→ 105	1999-07-27	1999-07-31	5	Igen

4. ábra. Azonosítók adattáblák között

#### Az adatbázisok tervezésének lépései

Az adatbázisokkal szemben – a teljesség igénye nélkül – az alábbi elvárásokat fogalmazza meg a felhasználó, megbízó:

redundancia mentesség (adat többszörözés elkerülése)

adatvédelem

helyreállíthatóság,

adatfüggetlenség (fizikai és logikai értelemben egyaránt)

következetesség a korrekt adatokat tekintetében

adatok közti kapcsolatok megbízható kezelése

az adatforgalom optimalizálása

több felhasználó esetén egyidejű hozzáférést biztosítása



A következőkben a legszélesebb körben elfogadott adatbázis tervezési modellt, annak lépéseit ismertetjük, a tartalom adta lehetőség figyelembevételével, mintegy érintő fogást véve az ismeretanyagon.

A modell (7. ábra) három fő szakaszból áll, első szakasz a tervezés (1. , 2. , 3. lépések). második szakasz a feltételek megadása ( 4. , 5 , 6.) és a harmadik szakasz az adatbázis elkészítése. ( 7. lépés).

#### 1. lépés: Követelményelemzés

Ebben a lépésben határozzuk meg a megoldandó feladatot, definiáljuk az adatbázis elkészítésének paramétereit. A, mit és hogyan problémakör meghatározása itt történik, egyeztetés a megbízóval és a leendő felhasználókkal.

Tanulmányozzuk, hogy milyen adatokat kapnak jelenleg a felhasználók, hogyan dolgozzák fel, gyűjtsük össze a jelenleg használatos dokumentációkat, táblázatokat, űrlapokat stb. Az elvárások meghatározása az új rendszer bevezetésével kapcsolatban.

Határozzuk meg, hogy mikről, kikről kell adatokat tárolni.

#### 2. lépés: Egyedek, táblák meghatározása

Az összegyűjtött adatok rendszerezése, azon információk keretrendszerének meghatározása, ahol ezeket használni fogjuk.

Egyed típus meghatározása, redundancia elkerülése.

Ha például egy személyi nyilvántartást készítünk, akkor pl. a következő egyedeket határozzuk meg: egyének, munkahelyek, lakóhelyek, iskolai végzettségek, telefonszám stb.

Ebben (5. ábra) az esetben a személy egy egyed típus, annak egy példánya lehet például Nagy Miklós és a hozzá kapcsolódó információk az alábbi táblázat szerint.

**Egyed-előfordulások**

Neve	Életkora	Testmagassága	Szeme színe	Haja színe
Kis Ede	29 év	183 cm	Kék	Barna
Nagy Miklós	31 év	179 cm	Fekete	Fekete

*5. ábra. Egyed előfordulások*

### 3. lépés: Attribútumok, mezők meghatározása

Megtervezzük a táblákat és a táblákban kötelezően szereplő mezőket, ezzel konkrétan definiáljuk az egyedtípusokat.

Az attribútumok meghatározása:

egyszerű, tovább nem bontható, és az összetett, mely több egyszerű értékből adódik össze..

( pl. összetett a lakcím, mely irányító számból, városnévből, utcanévből, házzámból áll.)

egyértékű csak egy értéket vehet fel, többértékű mely minden megjelenésénél más értéket vehet fel. (pl. adott személy különféle iskolai végzettségei)

tárolt érték, amelyet az adatbázis folyamatosan tárol, származtatott érték, melyet más attribútumok segítségével határozhatunk meg.

Fontos, hogy:

minden szükséges, de csakis a szükséges adatot vegyünk fel

ne vegyünk fel származtatott vagy kalkulált adatokat,

törekedjünk az egyszerű attribútumok alkalmazására.

#### 4. lépés: Az azonosítók meghatározása

A felvett táblák közötti relációk kialakítása, a táblákban tárolt információk egyértelmű definiálása.

(Összekapcsolhatunk vásárlót és az általa megvásárolt áruval)

Fontos az elsődleges kulcs meghatározása, ez olyan azonosító mely egy adott táblában csak egyszer szerepel. Az elsődleges kulcs általában egy rekord egy mezője.

Az egyednek elvileg több kulcsa is lehet, de általánosságban csak egyet (elsődleges kulcs) szokás használni, azt amelyik leginkább alkalmas az egyértelmű azonosításra.

#### 5. lépés: A kapcsolatok meghatározása

Kapcsolatok meghatározása a táblák kialakítása során. Itt kell megadni az elsődleges azonosítókat, a kiemelten fontos mezők meghatározása, a táblák relációiban álló rekordjainak összekapcsolása egymással,

#### 6. lépés: Ellenőrzés

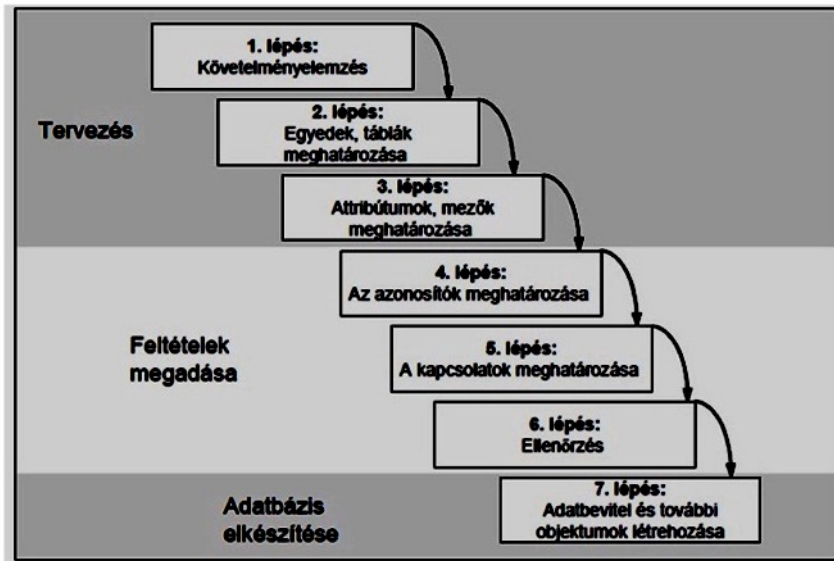
A táblák, mezők, kapcsolatok definiálása és elkészítése után ellenőrzést végzünk, hogy minden fontos szempontot, adatot, relációt szerepeltetünk-e? Nem maradt-e benne valamilyen hiba. Itt még egyszerűbben lehet korrigálni a hibákat, hiányosságokat, mint amikor már táblák feltöltésre kerültek, amikor „élesben” fut a rendszer.

Ebben a szakaszban lehet a rendszert próba jelleggel üzemeltetni, tesztelni, esetlegesen fals adatok rögzítését követően.

#### 7. lépés: Adatbevitel és további objektumok létrehozása

A 6. lépésben megadott ellenőrzés és korrekció után felvihetjük a valós adatokat, az adatbázist feltölthetjük a már létező táblákba.

Elkészítjük a kimenő dokumentumokat, táblák formai megjelenése, űrlapok, lekérdezések, statisztikák.

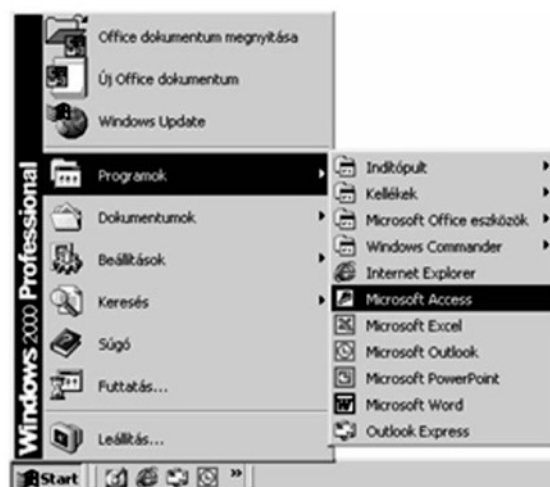


6. ábra. Az adatbázis tervezésének lépései

### ACCESS 2000 a program indítása

Mint ahogy a bevezetőben és az adatmodellek fejezetben említettük a dolgozat hátra lévő részében betekintést adunk a Windows-os környezetben futó és méltán népszerű, a már jól bevált ACCESS 2000 működésébe.

Az ACCESS program a microsoft fejlesztése és szerves része a fejlesztő által kiadott OFFICE program csomagnak. Az ACCESS egy relációs adatbázis-kezelő szoftver.



7. ábra. Az access indítása

A program elindítása után (7. ábra) megjelenik egy párbeszéd panel, ahol az adatbázis létrehozásának három lehetséges változata érhető el. Üres (ÚJ) ACCESS adatbázist hozhatunk létre, használhatjuk az adatbázis varázslót, vagy egy korábban elkészített adatbázist nyithatunk meg (8. ábra).



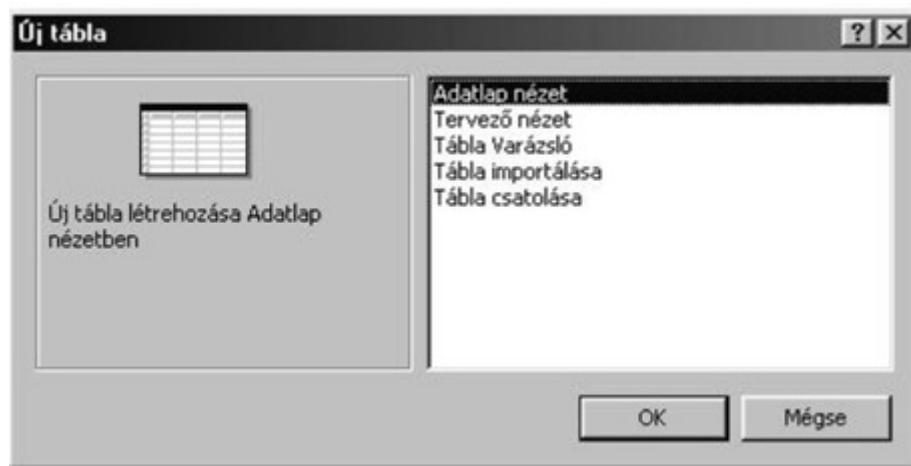
8. ábra. Adatbázis létrehozása

### Táblák

Az ACCESSben táblák tartalmazzák az adatbázis adatait, csoportosításuk a feladat jellegétől függően eltérhet. Például egy kórházi nyilvántartás esetén az egyik táblában tároljuk a betegek adatait egy másik táblában a műtétekre vonatkozó adatokat, akkor egy azonosító segítségével össze tudjuk rendelni adott beteghez rendelt műtétet. Mindkét tábla eltérő adattartalommal rendelkezik, csak az azonosító szerepel mindkettőben. Ezzel a módszerrel a hatékonyságot növelhetjük, a hiba lehetőségeket csökkenthetjük. A betegek tábla mezői azonos adatokat tartalmaznak (név, cím, életkor, műtét stb) ezek összességét nevezzük rekordnak.

Adatbázis megnyitása/Táblák/Új tábla (adatbázis létrehozása) , ezt követően 5 lehetőség jelenik meg a párbeszéd panelben (9. ábra).





9. ábra. Táblák kiválasztása

Adatlap nézet: Ritkán használatos nézet, gyakorlatilag a tábla szerkezetének kialakítása nélkül tölthetjük meg adatokkal.

Tervező nézet: a leggyakrabban használatos nézet, ki tudjuk alakítani a tábla struktúráját, meg tudjuk adni az oszlopok, mezők tulajdonságait (típus).

Tábla varázsló: a kezdő paraméterek megadása után a tábla varázsló elkészíti önállóan a táblázatot.

Tábla importálása: egy korábban elkészített adatbázisból tudunk adatokat, táblákat importálni az éppen folyó projektbe. Az importálás után a két tábla között nem lesz semmilyen reláció, tehát ha az egyikben módosítást hajtunk végre az a másikba nem fog átkerülni.

Tábla csatolása: hasonló az előzőhöz, itt azonban nem hozunk létre új táblát, csak hivatkozunk a másikra, ebben az esetben a végrehajtott módosítás mindkét táblában meg fog jelenni.

Tábla létrehozása: tábla létrehozásához a tervező nézetet használjuk, itt kialakíthatjuk a tábla mezőit, felépítését és megadhatjuk, hogy az egyes mezők milyen adattípusúak legyenek.

Adott táblát az őt alkotó mezők definiálásával állítjuk elő.

Az adattípus legtöbb esetben meghatározza az adott mező hosszát is.

A mezőnév az oszlop fejléce/megnevezése az oszlopban lévő adatokat és azok tulajdonságait azonosítja. A mezőnév maximum 64 karakter hosszúságú lehet.

Az ACCESS az alábbi mezőtípusokat használja:

Szöveg: tetszőleges karaktersorozat. maximum 255 karakter

Szám: numerikus adatok, melyekkel számításokat végezhetünk el 1, 2, 4, 8 vagy 12 bájt

Dátum: értéke 100 és a 9999 közti évek száma, mérete 8 bájt

Pénznem: numerikus adatok, melyeket pénznemként kezel, a tizedesjeltől balra 15, jobbra 4 számjegyet használhat 8 bájt

Számláló az ACCESS által létrehozott egymást követő vagy véletlen számok, egyedi azonosítására alkalmasak 4 vagy 16 bájt méretűek

Igen/Nem Logikai érték: amely két lehetséges érték közül csak az egyiket tartalmazza. 1 bit

OLE objektum: beágyazott objektum, más fájl formátumból (doc; xls; jpg; bmp; mp3)

Hiperhivatkozás: tetszőleges karaktersorozat, amely szöveggént kerül tárolásra, 2048 karaktert

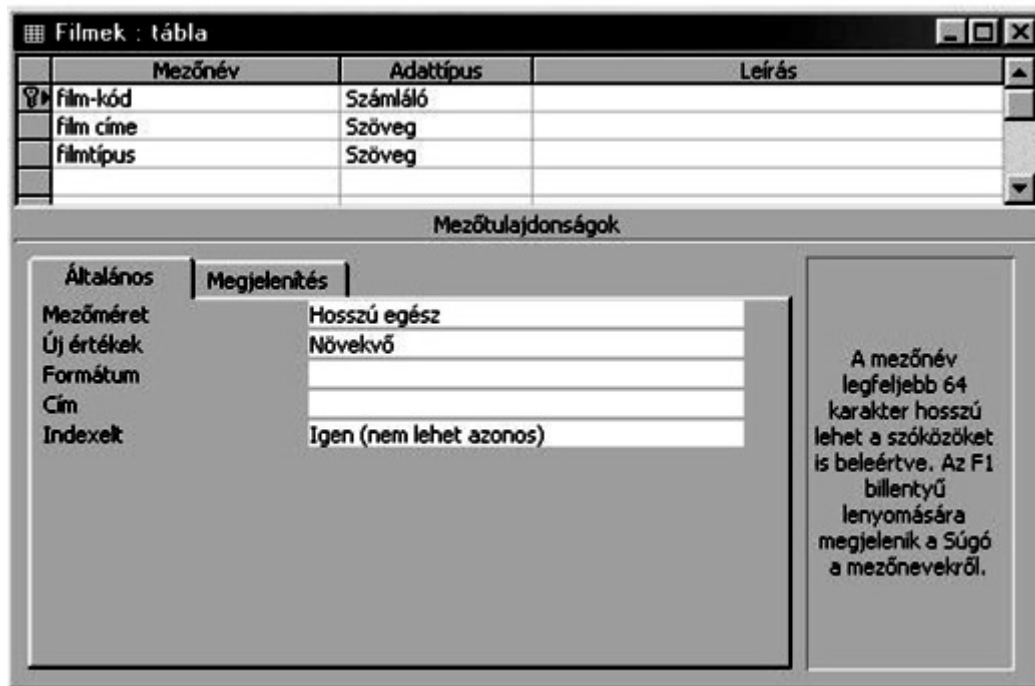
Keresés varázsló: olyan mezőt hozhatunk létre, melynek értékét egy legördülő lista segítségével lehet megadni. Ezek az értékek másik táblából származhatnak.

Mezőtulajdonságok

A mezők tulajdonságait a mezőtulajdonságok panelben tudjuk megadni. Ezekkel a beállításokkal meghatározhatjuk, hogy a mezőkben hogyan történjen az adatok tárolása, kezelése és a megjelenítése.

Táblák létrehozására a két legfontosabb, leggyakrabban használt nézet a tervező nézet és az adatlap nézet, nézzük ezeket kicsit részletesebben.

Tervező nézet (10. ábra)



10. ábra. Tervező nézet

A táblák létrehozásának leggyakoribb módja az, hogy először a feladat definiálása után megtervezzük a szükséges mezőket. Ehhez meghatározzuk és begépeljük a mezők neveit, majd az Adattípus oszlopban egy legördülő menüből kiválasztjuk a mező típusát.

A Leírás oszlopban megadhatjuk a mező adatainak rövid meghatározását, ez nem kötelező, csak egy lehetőség, hogy ezzel a későbbi felhasználó munkáját tudjuk segíteni.

Ebben a nézetben (az ablak alsó részén) módosíthatjuk az egyes mezők tulajdonságait is.

Az egyes mezők között a TAB\_ENTER vagy kurzormozgató billentyűkkel mozoghatunk, de egér használatával is elérjük ezt az eredményt.

### Mező beszúrása

Tervező nézetben bármikor beilleszthetünk a tábla végére egy újabb mezőt úgy, hogy a meglévő mezőnevek alatti első üres soron kattintunk és begépeljük a mező nevét. Az új mező ilyenkor a legutolsó helyre kerül. (Adatlap nézetben ez egy új oszlopot jelent majd a tábla végén.)

Ha a meglévő mezők közé szeretnénk egy újabb mezőt beszúrni, akkor arra a sorra kell kattintanunk, ahová az új mezőt el akarjuk helyezni. Ezután válasszuk ki a Beszúrás/Sorok

menüpontot vagy a Sor beszúrása

ikon.

#### Mező törlése

Felesleges mező törléséhez álljunk arra a sorra, amelyik a törlendő mezőt tartalmazza, majd

kattintsunk a Sorok törlése  
Szerkesztés/Sorok törlése menüpontra.

ikonra vagy a

#### Adatlap nézet

Amikor a Tervező nézetben elvégeztük a tábla definiálását át kell váltani Adatlap nézetbe azért, hogy adatokkal töltsük fel a táblánkat. A két nézet közötti váltást az eszközsoron a

Nézet ikon legörgetésével és a nézet kiválasztásával tudjuk realizálni, vagy a Nézet menüpont segítségével hajthatjuk végre.

Adatlap nézetben látható táblázat tartalmazza a Tervező nézetben megadott mezőneveket, a rekordokat pedig soronként kell begépelnünk.

#### Adatbevitel

Adatlap nézetben a tábla cellái között a TAB, ENTER vagy a kurzormozgató billentyűkkel lehet mozogni, vagy egér használatával.

Fontos, hogy mindig a mezőtípusnak megfelelő adatot rögzítsünk a mezőben, ellenkező esetben a program nem tudja értelmezni az értéket, és a bejegyzés érvénytelen lesz.

Pl. ne vigyünk be szám típusú mezőbe szöveget.

A mező tartalmát bármikor módosíthatjuk (11. ábra).



	film-kód	film címe	filmtípus
	1	Ötödik elem	akció
	2	Mátrix	akció
	3	Micimackó	mese
	4	Millennium Bell	koncert
	5	Alul semmi	víg
*	Számláló)		

Rekord: 5 összesen 5

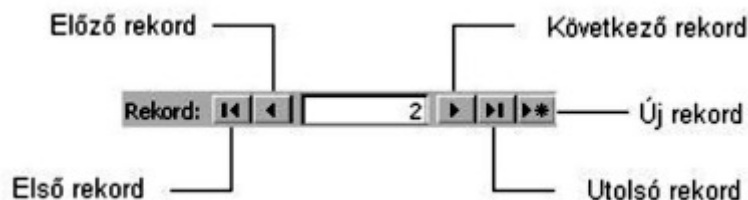
11. ábra. Adattábla

Az adatok elmentése külön nem szükséges, mivel a rekordok automatikusan mentésre kerülnek, amint egy másik rekordra váltunk. Ezt a sor előtti rekordjelzőben látható háromszög jel mutatja.

Új rekord rögzítése alatt egy ceruza szimbólum jelenik meg a sor elején. Ez azt jelenti, hogy a rekord aktív (szerkesztés alatt van), még nincs elmentve. Új sorba lépés esetén a ceruza szimbólum eltűnik és helyette megjelenik a háromszög szimbólum, jelezve, hogy az előző rekordot a program már elmentette.

#### Váltás a rekordok között

A rekordok között legegyszerűbben az egérrel tudunk váltani, lásd az alábbi ábrát, mely önmagáért beszél, további magyarázatot nem igényel (12. ábra).



12. ábra. Váltás rekordok között

### Rekord hozzáadása

Új rekord hozzáadásához az adattáblánk utolsó sorára kell kattintanunk. Új rekordot a Beszúrás/Új rekord menüponttal is létrehozhatunk.

### Rekord törlése

Rekord törléséhez először a törlendő rekord tetszőleges cellájára kell állnunk, majd válasszuk ki a Rekord törlése ikont az eszköztárról, vagy a Szerkesztés/Rekord törlése menüpontot, az új helyre.

Továbbá lehetőség adódik rekordok áthelyezésére, másolására, kivágására, rendezésére is, ez a már jól megismert Windows-os alkalmazások metódusai szerint működik.

### Több tábla

A gyakorlatban – általában – olyan összetett feladatokat kell megoldani, ahol szükséges több egymással kapcsolódó táblát rögzíteni.

Több táblából álló adatbázist a táblák megtervezése után a Tervező nézetben létesíthetünk, elkészítjük a táblák közti kapcsolatokat (relációkat). Ezt követően a tervező nézetben, határozzuk meg a táblák között kapcsolatot, töltjük fel adatokkal.

### Több tábla tervezése

Nézzünk egy ezzel kapcsolatos példát: telefonjegyzékünket számítógépen szeretnénk rögzíteni.

vezetéknév	keresztnev	irányítószám	Város	lakcím	telefonszám típusa	telefonszám
Kiss	Imre	4225	Debrecen	Alkotás út 12.	otthoni	(52) 345-217
Kiss	Imre	8000	Debrecen	Alkotás út 12.	munkahelyi	(52) 311-123
Kiss	Imre	8000	Debrecen	Alkotás út 12.	mobil	(30) 925-3527
Kiss	Imre	8000	Debrecen	Alkotás út 12.	fax	(22) 333-251
Tóth	Béla	8000	Székesfehérvár	Palotai út. 64.	otthoni	(22) 425-875
Kardos	Kinga	1204	Budapest	Epejes útca 28	mobil	(70) 937-8421
Kardos	Kinga	1204	Budapest	Epejes utca 28	otthoni	(1) 3447-249

13. ábra. Rekord felépítése

Probléma akkor keletkezik, ha egy személyhez több telefonszám, mobilszám, fax tartozik, mert ebben az esetben a 13. ábrán látható táblázatot úgy kellene kialakítani, hogy egy névhez annyi sor kapcsolódik, ahány telefonszámmal rendelkezik a konkrét személy. Így indokolatlanul nagy adatbázist kell létrehoznunk, üzemeltetnünk.

Követendő megoldás, ha a személyes adatok egyszer szerepelnek az adatbázisban, ezért bontsuk két külön táblára az adatbázist

Alkalmazzunk egy egyértelmű azonosítót, mely közös mind a két táblában és összerendeli az összetartozó információkat. Ezt az azonosítót nevezzük kulcsnak. A kulcs lesz az azonosító mező, használatával kiderül, hogy adott személyhez milyen telefonszámok tartoznak.

Ha ezt jól strukturáljuk, jól adtuk meg a kulcsot, akkor konkrét név lekérdezése esetén az összes kapcsolódó telefonszám látható lesz az azonosító segítségével.

### Kapcsolatok

Miután létrehoztuk adatbázisunk tábláit, kapcsolatokat kell megállapítani, meghatározni közöttük. Attól függően, hogy a táblák rekordjai milyen módon kapcsolódnak egymáshoz, három esetet különböztetünk meg: – mint már említettük – egy az egyhez, egy a többhöz, több a többhöz kapcsolat.

### Kapcsolat létesítése (14. ábra)

Filmesek : tábla		
	Mezőnév	Adattípus
	film-kód	Számláló
	film címe	Szöveg
	filmtípus	Szöveg

14. ábra. Kapcsolat létesítése

Általában minden táblának van olyan mezője, amelyik egyedileg azonosítja a táblában tárolt rekordokat. Vagyis meghatározza, hogy melyik előfordulásról van szó.

Ezt a mezőt nevezzük a tábla elsődleges kulcsának. Ezt a kulcs mezőt a Tervező nézetben

tudjuk beállítani (eszközsor Elsődleges kulcs ikon használatával) Ekkor a mező neve előtt megjelenik a kulcs szimbólum.

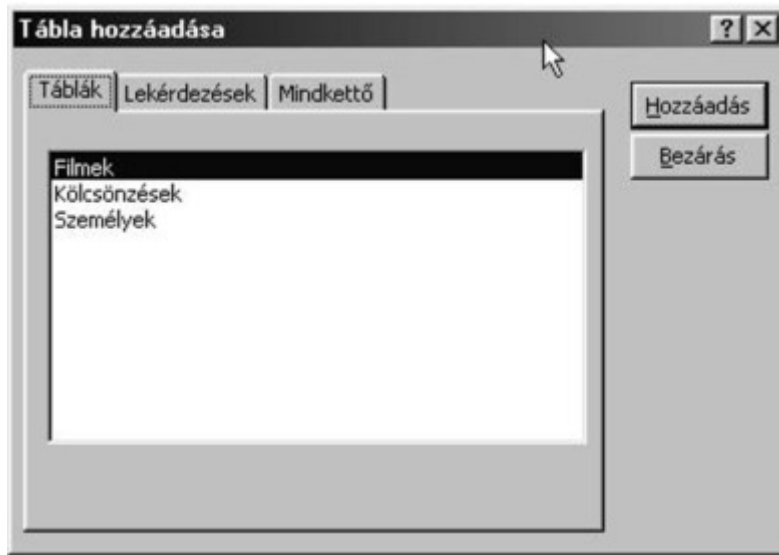
Az elsődleges kulcs segítségével van lehetőségünk a táblák között kapcsolatot teremteni. hogy a kulcsmezőt és a másik táblában a kulcsmezővel azonos értékű mezőket egymáshoz rendeljük. Pl. a videófilmek adatbázisunk Filmek táblájának elsődleges kulcsa a film-kód mező, ezt a mezőt kell kapcsolnunk a Kölcsönzések tábla film-kód mezőjéhez.

Kapcsolat létrehozásának első lépése, hogy az adatbázis ablakban az eszköztár

Kapcsolatok  
kiválasztjuk az Eszközök/Kapcsolatok menüpontot.

ikonjára kattintunk, vagy a

Ekkor megjelenik a Tábla hozzáadása ablak, amelyből a Hozzáadás gomb segítségével a kapcsolatok ablakba vegyük fel az összekapcsolandó táblákat (15. ábra).

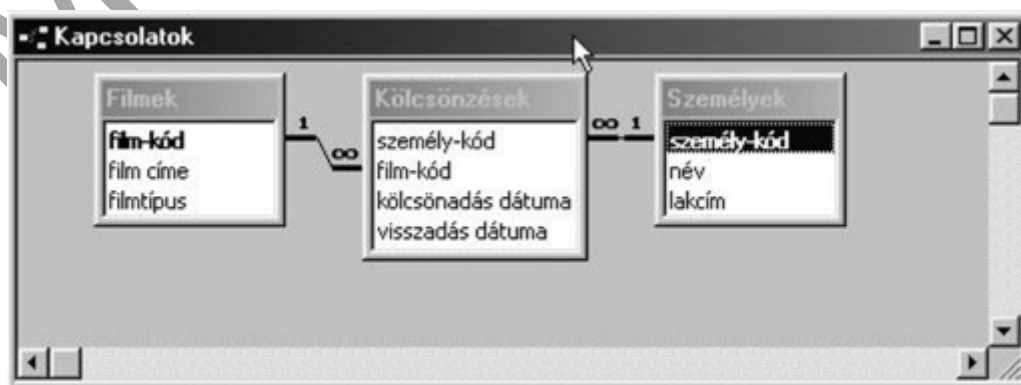


15. ábra. Táblák hozzáadása

(Az elsődleges kulcs mezőket félkövér betűk jelzik a kapcsolatok ablakban.) Az egyik tábla kulcsmezőjét az egér segítségével húzzuk át a másik tábla hasonló mezőjére. A kapcsolódó mezőknek nem kötelező azonos nevet adnunk, viszont ugyanolyan típusú információt kell tartalmazniuk, és azonos adattípusúaknak kell lenniük.

A két kapcsolt tábla között ezután egy illesztő vonal jön létre, mint az a következő ábrán (16. ábra) látszik is, ebben az esetben egy filmnyilvántartó és rendező adatbázis esetében.

Miután a táblák között létrehoztuk a szükséges illesztő vonalakat, el kell mentenünk a kapcsolatok ablakot. Ezután következhet az adatbevitel a táblákba.



16. ábra. Kapcsolatok táblák közt



## Lekérdezések

Az adatbázisok elkészítésének, a táblák feltöltésének, a relációk meghatározásának legfontosabb célja, hogy ebből az adatbázisból a későbbiek során adatokat tudjunk legyűjteni, ezeket a legyűjtéseket nevezzük lekérdezéseknek. Az adatbázisban a lekérdezések végzik a tevékenység legjelentősebb részét.

Legáltalánosabb funkciójuk különböző adatok kigyűjtése a táblákból. Az összegyűjteni kívánt adatok általában több táblában helyezkednek el, a lekérdezések segítségével tudjuk az adott cél megfogalmazása után egy táblában megjeleníteni.

Általában nem cél az, hogy az összes rekordot egyszerre tekinthessük meg, ezért a lekérdezés konfigurálásában meg tudjuk adni azokat az adatokat, melyeket szerepeltetni szeretnénk. Ezt a tevékenységet nevezzük szűrésnek, így csak a szükséges rekord részletek kerülnek megjelenítésre. A lekérdezések gyakran lehetnek űrlapok és jelentések készítésének forrásai.

A lekérdezéseknek két alaptípusa van, ez a választó és módosító lekérdezés.

A választó lekérdezés egyszerűen összegyűjti a kért adatokat. Az eredményt megtekinthetjük a képernyőn, kinyomtathatjuk, archiválhatjuk, ezen felül felhasználhatjuk rekordforrásként egy űrlap vagy jelentés készítéséhez.

Módosító lekérdezés adott feladatot úgy hajtja végre, hogy módosított táblákat hoz létre, adatot adhat hozzá egy meglévő táblához, frissítheti stb.

E két fő lekérdezés fajtán kívül még az alábbiakat ismeri a szakirodalom:

### Táblakészítő lekérdezés

Megegyezik a választó lekérdezéssel, de a kiválasztott rekordok nem a képernyőn, hanem egy új táblában jelennek meg

### Törlő lekérdezés

Adott feltételeknek megfelelő adatok törlésére szolgál

### Hozzáfűző lekérdezés

Az adott feltételeknek megfelelő rekordokat hozzáfűzi egy már korábban létrehozott táblához, amely nem lehet azonos a forrás táblával

### Frissítő lekérdezés:

A megfelelő rekordok adatainak módosítására szolgál

### Paraméteres lekérdezés:

Egy bizonyos feltétel megadását a felhasználóra bizzuk

A felhasználó adja meg a hiányzó feltételt

### Keresztábrás lekérdezés:

Az adatok két szempont szerinti elemzését jelenti

### Lássunk egy példát a választó lekérdezésre:

A tartalom adta lehetőségek miatt most csak a választó lekérdezésre tudunk példát nézni, így a módosító lekérdezést kénytelenek vagyunk mellőzni.

Feltételezett adatbázisunk a FÖLD országainak adatait, paramétereit tartalmazza (országnev, beszélt nyelv, kontinens, államforma, pénznem, GDP, lakosság, éghajlat, tengeri kapcsolat, stb)

Létrehozunk egy választó lekérdezést, amelynek eredménye egy olyan tábla, melyben ABC rendben szerepelnek az ország nevek, a földrész, az államforma és a hivatalos fizetőeszköz neve is.

A feladat végrehajtásának első lépése az adattáblák hozzáadása, összerendelése, ezt követően a kapcsolatok az előzetes beállításoknak megfelelően megjelennek (17. ábra).



17. ábra. Választó lekérdezés 1. stáció

A második lépés a táblázat egyes oszlopaiban megjelenő adatok származási helyének megadása. Ki kell választanunk a szükséges adatmezőket az egyes adattáblákból, egy legördülő lista segítségével (18. ábra).

Mező:	Ország	Földrész	Pénznem	Államforma
Tábla:	Föld	Földrész	Pénznem	Államforma
Rendezés:	Növekvő			
Megjelenítés:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Feltétel:				
vagy:				

18. ábra. Választó lekérdezés 2. stáció

Ezt követően kiválasztjuk, hogy melyik oszlop szerint legyen rendezve a tábla, és ha jól paramétereztünk mindent, akkor megkapjuk a kívánt eredményt.

A lekérdezést eddig Tervező nézet készítettük, a szűrés eredményét az Adatlap nézetben láthatjuk, kapcsoljunk tehát át erre a nézetre.

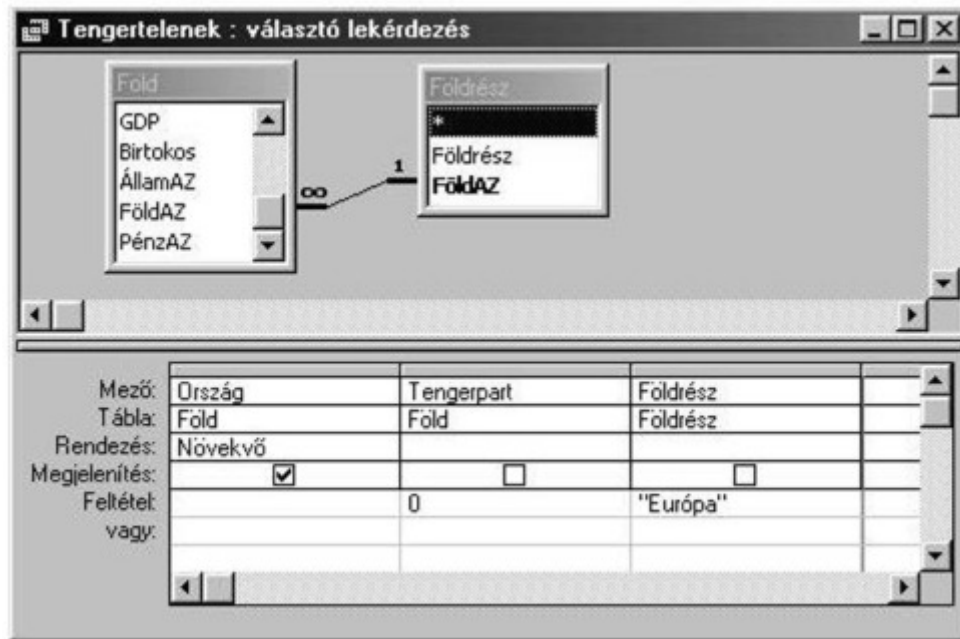
A példa egy következő lépése lehet:

Készítsünk listát az európai államok neveivel, területükkel és fővárosuk nevével. Ehhez a rendelkezésre álló – a feltételezés szerint – a Föld/Földrész adattáblát használjuk. Itt meg kell adnunk egy feltételt, miszerint csak azok a rekordok szükségesek, melyeknél a rekord földrész mezőjében az Európa megnevezés szerepel. Ezek után már megkaphatjuk azt a listát, melyben csak és kizárólag az európai országok adatai fognak szerepelni.



19. ábra. Választó lekérdezés 3. stáció

Zárásként kíváncsiak vagyunk a tengerrel nem rendelkező európai államok neveire:



20. ábra. Választó lekérdezés 4. stáció

Ebből a képből a szűrési feltételek megadása olvasható ki.



21. ábra. Választó lekérdezés 5. stáció

Ebből pedig a kapott eredmény.

### Űrlapok

Az ACCESSben űrlapokat használunk az adatokkal végzett munkákhoz, könnyen és egyszerűen kezelhető felülete az adatok rögzítésének, felhasználó barát megjelenésével kiváló eszköze az adatok rögzítése mellett a gyors adat keresésnek, segítségével kiküszöbölhetjük a rögzítési, adatbeviteli hibákat is. Az űrlapokon megtalálhatóak a kapcsolódó műveletek végrehajtó gombjai is.

Az űrlap öt, egymástól jól elkülöníthető résszel rendelkezik, ezek az:

az űrlapfej tartalmazza az űrlap címét, a használati utasításokat és a művelete végrehajtását segítő parancsgombokat. (képernyőn és nyomtatásban is megjelenik)

az oldalfej nyomtatás alkalmával minden olyan oldal tetején megjelenik, ahol az adatbázis ismétlődő adatokat tartalmaz.

az űrlaptörzs az űrlap legfontosabb alkotórésze, itt jelennek meg a rekordok.

az oldalláb minden oldal alján ismétlődő információk megjelenítésére szolgál ( oldalszám, dátum, magyarázatok).

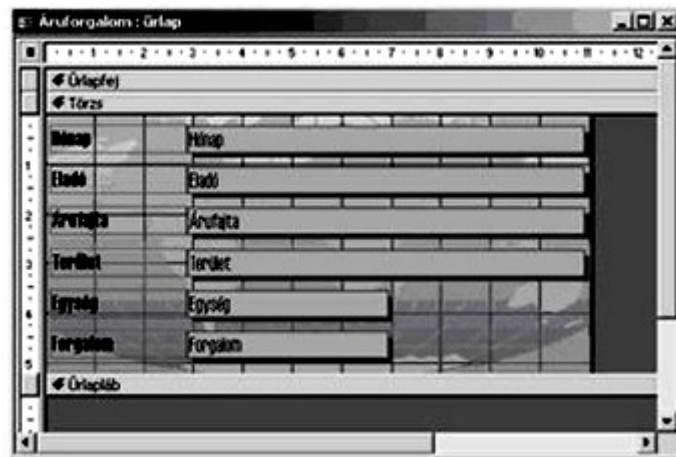
az űrlapláb az űrlapfejhez hasonló funkcióval rendelkezik. Az űrlapláb nyomtatásban az utolsó oldal alján jelenik meg.

Űrlapok segítségével azt is meghatározhatjuk, hogy az adatbázisban más felhasználók milyen jogosultsági szintekkel rendelkezhetnek. Lehetőség van például olyan űrlap létrehozására, amely csak bizonyos mezőket jelenít meg, csak meghatározott műveleteket hajthat végre rajta az adott felhasználó, a jogosultsági szintjének megfelelően.

### Az űrlapok nézetei

Az ACCESSben elkészített űrlapokat ötfajta nézetben tekinthetjük meg, ezek a következők:

A Tervező nézetet (22. ábra) az űrlapok elkészítésekor vagy módosításakor használjuk. Ebben a nézetben az űrlapelemek tulajdonságait, formátumait és elrendezését módosíthatjuk.



22. ábra. Úrlapok nézetei/tervező nézet

Az Úrlap nézetben az űrlapot a beállításától függően egyszeres űrlap vagy folyamatos űrlap formátumban tekinthetjük meg. Az egyszeres űrlap nézetre példa a 23. ábra.



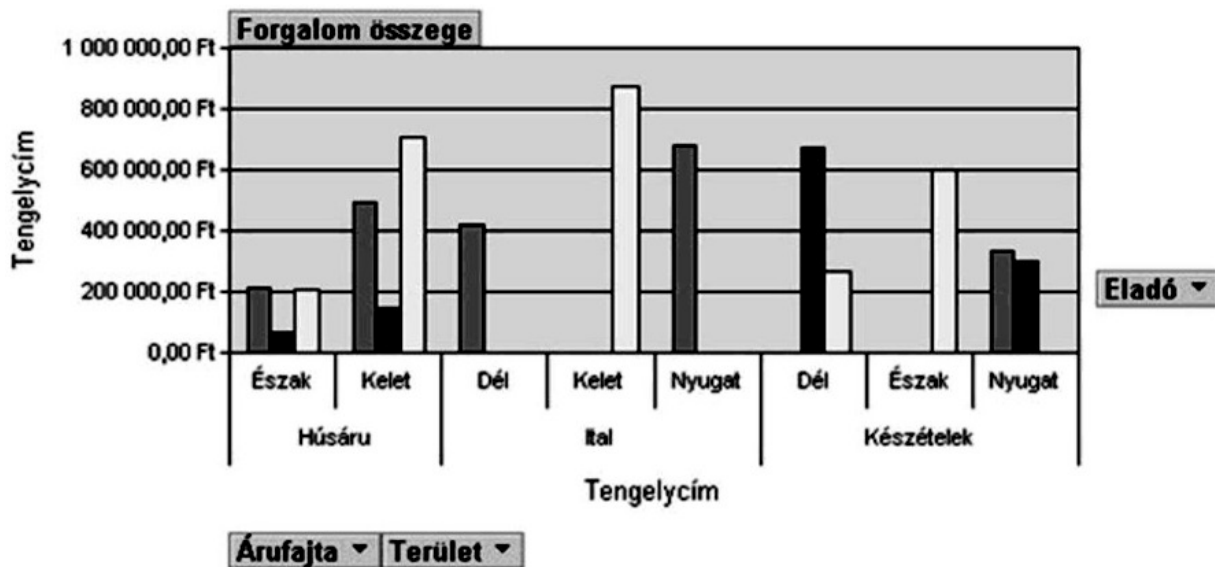
23. ábra. Úrlapok nézetei/űrlap nézet

Az Adatlap nézetben az űrlapon található mezőket a táblák adatlap nézetéhez hasonló táblázatos formában tudjuk megjeleníteni.

A Kimutatás nézetben az űrlapban szereplő mezők tartalmát összegző. Táblázatot készíthetünk.

A Kimutatás és Kimutatás diagram (25. ábra) nézetekben az összesítő táblázat illetve diagram tartalmát a Fogd és vidd módszerrel kell kialakítanunk.

A kimutatásdiagram (24. ábra) nézetben az űrlapon szereplő mezők tartalmát tudjuk diagram formában megjeleníteni.



24. ábra. Kimutatás diagram

### Jelentések

A jelentéseket a táblákban szereplő adatok összegzésére és bemutatására használják. Egy jelentés általában egy meghatározott kérdést válaszol meg, például: „Mennyi pénzt kaptam az egyes vevőktől ebben az évben?” Vagy: „Mely városokban élnek a vevőim?” Minden jelentésnek olyan formát adhat, hogy a lehető legjobban áttekinthető módon mutassa be az információikat.

A jelentéseket bármikor futtathatja, mindig az adatbázis aktuális adatait tükrözik. A jelentéseket általában nyomtatható formában tervezik meg, de a kész jelentést megtekintheti a képernyőn, exportálhatja más programba, vagy elküldheti e-mailben is.

**Összefoglaló:** jelen tananyagban megismertük az adatbázisok fogalmát, az adatbázisok felépítését, a relációkat. Megismertük az egyik legismertebb adatbázis kezelő programot, az MS Access-t. Az adatbázis tágabb értelemben egy olyan adathalmaz, melynek elemei egy meghatározott paraméter alapján összetartozóak. A programnak tudnia kell kezelni az adatok összerendezését, kapcsolatok nyilvántartását, lehetőséget kell kínálnon különféle szűrős lekérdezések futtatására, az adat integritást, adatvédelmet biztosítani kell. Az adatbázis az adatok és a köztük lévő összefüggések rendszere.



## TANULÁSIRÁNYÍTÓ

Most nézzük át, hogy miről is tanultunk? Milyen elvek szerint volt felépítve a tananyag?

Készítsünk az olvasottak alapján vázlatot, írjunk jegyzeteket a nyomtatott szöveg mellé.

Ha már megvan a vázlat és ismerjük a tananyag fő elemeit, akkor ezt követően összefüggéseket keressünk az egyes fejezetek között. Készítsünk listát azon fogalmakról, melyek – álláspontunk szerint – elengedhetetlenül fontosak a jó szakmai munka elvégzéséhez.

A tananyag információ tartalmának elsajátításával elérhető, hogy a tanuló elméleti tudáshoz juthasson az adatbázis kezelő szoftverek működése tekintetében.

Az elsajátított ismeretanyag értelmezése tekintetében szükség van a tanuló jó áttekinthető képességére, lássa az összefüggést a különféle szoftver elemek között, meg tudjon határozni a közös pontokat.

Az adott feladathoz képes legyen optimális táblakapcsolatokat felépíteni, találja meg a legjobb megoldást., értse a relációkat.

Az elméletben megtanult tudásanyagot, célszerű gyakorlati foglalkozásokon begyakorlással elsajátítani, konkrét példák megoldásával rutin szerűvé tenni.

## ÖNELLENŐRZŐ FELADATOK

### 1. feladat

Nevezze meg az adatbázis kezelés alapfogalmait.

---

### 2. feladat

Táblák közti kapcsolatok, relációk.

---

### 3. feladat

Az adatbázis tervezés lépései.

---

### 4. feladat

Ismertessük az Access mezőtípusai.

---

### 5. feladat

Ismertesse az Űrlap nézeteit.

**6. feladat**

Határozza meg, hogy mit nevezünk rekordnak.

\_\_\_\_\_

\_\_\_\_\_

MUNKANYAG

## MEGOLDÁSOK

## 1. feladat

Az adatbázis fogalma:

Az adatbázis információk gyűjtésére, rendszerezésére szolgáló eszköz, mely arra is alkalmas, hogy különféle szempontok szerint rendezéssel tudjunk adatokat, információkra szert tenni. Adatok tárolását, rendezését oldhatjuk meg vele, ez szólhat személyekről, árukészletről, filmek-, zenék-, számlázási- és készlet nyilvántartási adatokról, stb nyilvántartásáról, rendezéséről.

Az adatbázis tágabb értelemben egy olyan adathalmaz, melynek elemei egy meghatározott tulajdonságuk révén összetartozóak.

Az adatbázis kezelés alapfogalmai:

Az adatbázis a felhasznált adatok és a közöttük fennálló kapcsolatok, rendszere, nagyon fontos, hogy az adatbázisunk szerkezetét jól megtervezzük, mivel ennek következménye lesz az, hogy a későbbiekben milyen legyűjtéseket, szűréseket tudunk elvégezni benne.

A tábla logikailag összetartozó adatokat tartalmaz, tábla (táblázat) oszlopokból és sorokból épül fel, amely mezőknek, illetve rekordoknak nevezünk.

A rekord az adatbázis egy adatlánc mentén összefüggő sora, itt és így tároljuk az egymással összefüggő adatokat (pl. személy neve, neme, lakcíme, születési helye és ideje stb).

A személyt esetenként egyenek is nevezünk. Az egyed az, amelynek az adatait tároljuk és gyűjtjük az adatbázisban.

A mező az adatbázis egy oszlopa, amelyben az egyedek tulajdonságértékeit tárolja.

Az attribútum vagy tulajdonság az egyed valamelyik jellemző tulajdonsága. (Egy személy egy jellemző tulajdonsága lehet például a neve is.)

Egy egyedre vonatkozó tulajdonságok összességét egyedtípusnak nevezünk.

( például a név, életkor, szeme színe együtt).

Elsődleges kulcs: a tábla rekordjainak egyértelmű és egyben legfontosabb azonosítója, értéke egyedi.

Idegen kulcs: egy olyan azonosító vagy hivatkozás, melynek segítségével egy másik tábla másik, jellemző kulcsára hivatkozhatunk

Index: Egy előre megadott paraméter sor, mely a táblákban való keresés és a sorba rendezés gyorsítására alkalmas eszköz.

## 2. feladat

Egy-egy (1:1) kapcsolat: az egyik tábla egy eleméhez a másik tábla pontosan egy és mindig csak eleme kapcsolódik. Ezt a kapcsolattípust használjuk többek között házastársak nyilvántartására. Két adóévre: az adóévet megelőző és az azt megelőző adóévre.

Egy-több (1:N) kapcsolat: az egyik tábla egy eleméhez a másik tábla több eleme is tartozhat, egy egyed előfordulása egy vagy több egyed előforduláshoz van rendelve. Az egyik tábla egy rekordjához, egy másik tábla több rekordja tartozik, ezt a kapcsolattípust alkalmazzuk pl. boltok és szállítók közti adatok nyilvántartásakor.

Több-több (N:M) kapcsolat: bármely tábla elemeihez a másik tábla tetszőleges számú eleme tartozhat. Adott egyed egy vagy több előfordulása kapcsolatban áll egy vagy több előfordulásával. Ebben az esetben a kapcsolatot egy harmadik tábla révén tudjuk képezni, ezt nevezzük illesztő táblának. Az illesztő táblába fel kell venni mindkét tábla elsődleges kulcs mezőjét, így megteremtve a kapcsolat leképezésének lehetőségét. Ezt az a kapcsolattípust alkalmazzuk pl. egy cég ügyfelei és az ügyfélnek nyújtott szolgáltatások összerendelésére.

## 3. feladat

### 1. lépés: Követelményelemzés

Ebben a lépésben határozzuk meg a megoldandó feladatot, definiáljuk az adatbázis elkészítésének paramétereit. A, mit és hogyan problémakör meghatározása itt történik, egyeztetés a megbízóval és a leendő felhasználókkal.

### 2. lépés: Egyedek, táblák meghatározása

Az összegyűjtött adatokat rendszerezése, azon információk keretrendszerének meghatározása, ahol ezeket használni fogjuk.

### 3. lépés: Attribútumok, mezők meghatározása

Megtervezzük a táblákat és a táblákban kötelezően szereplő mezőket, ezzel konkrétan definiáljuk az egyedtípusokat.

### 4. lépés: Az azonosítók meghatározása

A felvett táblák közötti relációk kialakítása, a táblákban tárolt információk egyértelmű definiálása.

### 5. lépés: A kapcsolatok meghatározása

Kapcsolatok meghatározása a táblák kialakítása során. Itt kell megadni az elsődleges azonosítókat, a kiemelten fontos mezők meghatározása, a táblák relációiban álló rekordjainak összekapcsolása egymással,

### 6. lépés: Ellenőrzés

A táblák, mezők, kapcsolatok definiálása és elkészítése után ellenőrzést végzünk, hogy minden fontos szempontot, adatot, relációt szerepeltetünk-e?

### 7. lépés: Adatbevitel és további objektumok létrehozása

A 6. lépésben megadott ellenőrzés és korrekció után felvihetjük a valós adatokat, az adatbázist feltölthetjük a már létező táblákba.

#### 4. feladat

Az ACCESS az alábbi mezőtípusokat használja:

Szöveg: tetszőleges karaktersorozat. maximum 255 karakter

Szám: numerikus adatok, melyekkel számításokat végezhetünk el 1, 2, 4, 8 vagy 12 bájttal

Dátum: értéke 100 és a 9999 közti évek száma, mérete 8 bájttal

Pénznem: numerikus adatok, melyeket pénznemként kezel, a tizedesjeltől balra 15, jobbra 4 számjegyet használhat 8 bájttal

Számláló az ACCESS által létrehozott egymást követő vagy véletlen számok, egyedi azonosítására alkalmasak 4 vagy 16 bájttal méretűek

Igen/Nem Logikai érték: amely két lehetséges érték közül csak az egyiket tartalmazza. 1 bit

OLE objektum: beágyazott objektum, más fájl formátumból (doc; xls; jpg; bmp; mp3)

Hiperhivatkozás: tetszőleges karaktersorozat, amely szöveggént

kerül tárolásra, 2048 karaktert

Keresés varázsló: olyan mezőt hozhatunk létre, melynek értékét egy legördülő lista segítségével lehet megadni. Ezek az értékek másik táblából is származhatnak.

#### Mezőtulajdonságok

A mezők tulajdonságait a mezőtulajdonságok panelben tudjuk megadni. Ezekkel a beállításokkal meghatározhatjuk, hogy a mezőkben hogyan történjen az adatok tárolása, kezelése és a megjelenítése.

## 5. feadat

Az ACCESSben elkészített űrlapokat ötfajta nézetben tekinthetjük meg, ezek a következők:

A Tervező nézetet az űrlapok elkészítésekor vagy módosításakor használjuk. Ebben a nézetben az űrlapelemek tulajdonságait, formátumait és elrendezését módosíthatjuk.

Az Űrlap nézetben az űrlapot a beállításától függően egyszeres űrlap vagy folyamatos űrlap formátumban tekinthetjük meg.

Az Adatlap nézetben az űrlapon található mezőket a táblák adatlap nézetéhez hasonló táblázatos formában tudjuk megjeleníteni.

A Kimutatás nézetben az űrlapban szereplő mezők tartalmát összegző. Táblázatot készíthetünk.

A Kimutatás és Kimutatás diagram nézetekben az összesítő táblázat illetve diagram tartalmát a Fogd és vidd módszerrel kell kialakítanunk.

A kimutatásdiagram nézetben az űrlapon szereplő mezők tartalmát tudjuk diagram formában megjeleníteni.

---

## 6. feladat

A rekord az adatbázis egy adatlánc mentén összefüggő sora, itt és így tároljuk az egymással összefüggő adatokat (pl. személy neve, neme, lakcíme, születési helye és ideje stb).



## IRODALOMJEGYZÉK

## FELHASZNÁLT IRODALOM

Békéssy–Demetrovics: Adatbázis–szerkezetek, Akadémia, 2005

Gajdos Sándor: Adatbázisok, Műegyetemi, 2004.

Bártfai Barnabás– Budavári Oszkár: Adatbázis–kezelés, BBS–E, Budapest, 1998.

Békefi Zoltán: Táblázatkezelés, Talentum, Budapest, 1996.

Bódy Bence – Németh J. András: A táblázatkezelés furfangjai, Novorg, Budapest, 1996.

Access 2000 kezdőknek – elektronikus tananyag

Informatikai és Hírközlési Minisztérium – Adatbázis kezelés Windows Access 2000–rel

WIKIPEDIA a szabad enciklopédia

A(z) 0061-06 modul 010-es szakmai tankönyvi tartalomeleme felhasználható az alábbi szakképesítésekhez:

A szakképesítés OKJ azonosító száma:	A szakképesítés megnevezése
55 345 02 0010 55 01	Logisztikai műszaki menedzserasszisztens
55 345 02 0010 55 02	Terméktervező műszaki menedzserasszisztens
52 341 04 1000 00 00	Kereskedelmi ügyintéző
54 345 02 0000 00 00	Logisztikai ügyintéző
52 342 01 0000 00 00	Marketing- és reklámügyintéző
52 342 01 0100 52 01	Hirdetési ügyintéző
52 342 02 0000 00 00	PR ügyintéző
54 341 01 0000 00 00	Külkereskedelmi üzletkötő

A szakmai tankönyvi tartalomelem feldolgozásához ajánlott óraszám:

22 óra

MUNKANYELVI

MUNKANYAG

A kiadvány az Új Magyarország Fejlesztési Terv  
TÁMOP 2.2.1 08/1-2008-0002 „A képzés minőségének és tartalmának  
fejlesztése” keretében készült.

A projekt az Európai Unió támogatásával, az Európai Szociális Alap  
társfinanszírozásával valósul meg.

Kiadja a Nemzeti Szakképzési és Felnőttképzési Intézet

1085 Budapest, Baross u. 52.

Telefon: (1) 210-1065, Fax: (1) 210-1063

Felelős kiadó:

Nagy László főigazgató