

## **Informatika II. - Adatbázis-kezelés**

Szerző(k):

**Krankovits Melinda**

2007-07-04



---

## Tartalom

### Általános adatbázis-kezelési alapok

Bevezetés	1
Alapfogalmak	2
Adatbázis-kezelő rendszer felépítése	5
Modulzáró feladatok.	7

### Adatbázi tervezés, modellezés

Normálformák, relációs modell	9
Az adatmodellezés három szintje.	13
Modulzáró feladatok	17

### MS Access alapok

Adattípusok, kifejezések, operátorok.	19
Alapvető függvények.	22
Modulzáró feladatok	25

### Adatkezelés

Adatbázis létrehozása, megnyitása.	27
Táblák, indexek.	30
Kifejezés szerkesztő használata	38
Kapcsolatok, hivatkozási integritás.	43
Lekérdezés alapismeretek.	47
Adatbevitel a táblába	53
Modulzáró feladatok	58

### Űrlapok, Jelentések

Úrlapok elemei, létrehozása.	62
A parancsgomb varázsló használata.	70
Jelentések készítése, elemei	75
Modulzáró feladatsor	78

## Bevezetés.

Ez a tananyag az Informatika II. tantárgy Adatbáziskezelés moduljának jegyzetéhez készült, annak tanulási útmutatójának szerepét tölti be.

A jegyzet az informatika szakos hallgatók jegyzetének egy könnyebben érthető, leszűkített változata, melyet szintén Kiss Jenő egyetemi adjunktus írt, Adatbázis-kezelés (MS Access 97) címmel. (Novadat kiadó, 2006)

Minden lecke elején utalunk arra, hogy az adott témakörhöz tartozó tananyag a jegyzetben hányadik oldalon található. A jegyzetben szereplő tananyagot nem jelenítjük meg újra a képernyőn, azt a jegyzetből kell megtanulni.

Az egyes, nehezebben érthető részekhez további magyarázatokat talál ebben az elektronikus anyagban. A leckék végén elhelyezett ellenőrző kérdések segítségével ellenőrizheti tudását.

A jegyzetben megismerkedhetnek az adatmodell, és adatbázis alapfogalmakkal, a relációs adatmodell és SQL elemeivel, valamint bővebben az Access adatbázis-kezelő használatával.

A tananyagot modulokra osztottuk, az egyes modulok végén modulzáró feladatok segítik a tananyag rendszerezését, összefoglalását. A vizsgafeladatok a leckék végén, illetve a modulzáró feladatsorban található feladatokhoz hasonlóak lesznek.

Az elektronikus tananyagot letölthető feladatokkal bővítettük, melyek a tankönyvi példa alapján kidolgozott adatbázis egy-egy fázisát mutatják. A leckék végén utalást talál arra, melyik letölthető feladat tartalmazza az aktuális adatbázist. Ez nem jelenti azt, hogy Önállóan nem kell megoldani a tankönyvi példát, csupán mankóként szolgáljon, ha elakad a megvalósításban! Munkájához sok sikert kívánunk!

## Az adatmodell fogalma, elemei.

### Tanulási célok

A lecke tanulása után Ön képes lesz:

- az adatmodell, és az adatbázis fogalmának helyes értelmezésére
- az egyes adatmodell elemek jellemzőinek felidézésére

### Tevékenység:

Olvassa el a jegyzetben az 7. oldaltól a 9. oldalig tartó anyagot!

### Összefoglalás

*Definíció: Az adatmodell egyedek, tulajdonságok és kapcsolatok halmaza, amely absztrakt módon tükrözi a valós objektumoknak, azok jellemzőinek, és viszonyainak elvont kategóriáit.*

Az **adatmodell** olyan koncepciók gyűjteménye, amelyek egy adatbázis szerkezetét egyértelműen leírják. A szerkezet magában foglalja az adatok típusát, azok kapcsolatát, a korlátozó feltételeket, az adatkezelési műveleteket, és felhasználói igényeknek megfelelő összetett műveleteket.

Az **adatbázis** egy megvalósított adatmodell, amely az adatokon kívül tartalmazza azok típusait és kapcsolatait leíró **metadatokat**.

#### **Adatmodell elemek:**

**egyed:** minden olyan elem, ami más dologtól megkülönböztethető. Pl. személy, autó..

**tulajdonság:** az egyedeket tulajdonságokkal (attribútum) írjuk le. Minden olyan tulajdonság kulcs attribútum, amely egyértelműen meghatározza, hogy az egyed melyik értékéről van szó. Pl. autónál az alvászám, személynél a személyi szám.

**kapcsolat:** egyedek közötti viszonyok leírására szolgál.

Akkor mondjuk, hogy két vagy több egyedtípus kapcsolatban van egymással, ha létezik olyan rendező elv, amely az egyed típusok egyedeit egymáshoz rendeli.

**1. egy-egy kapcsolat,** jelölése 1:1

ebben az esetben egy szülő egyedhalmaz minden eleméhez egy gyerek egyedhalmazból csak egy elem tarozhat.

*Egy házas férfi - házastársa - egy házas nő* lehet. Itt a férfi egyethalmaz egy példány van kapcsolatban a nő egyedhalmaz egy példányával. Vegyük észre, hogy ez a példa csak monogám országokban és csakis a házastárs kapcsolatra állja meg a helyét, munkatársi viszonyra már nem.

**2. egy-többhöz kapcsolat**, jelölése 1:N

Ilyenkor a gyerek egyedhalmazból több egyed is kapcsolható a szülő halmazhoz, tipikusan a VAN-E kapcsolatok ilyenek.

A *személy - van\_e - autó* kapcsolatban egy személynek több autója is lehet, de egy autónak csak egy tulajdonosa.

**3. több-többhöz kapcsolat**, jelölése M:N

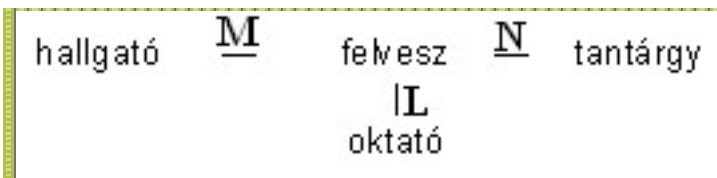
Mindkét halmazból több egyed is részt vehet a kapcsolatban.

színész - játszik - darab kapcsolat

hallgató - felvesz - tantárgy

**4. terner kapcsolat**, jelölése M:N:L

Speciális kapcsolat, amely több M:N kapcsolatot összevonása.



## Ellenőrző feladatok - Alapfogalmak

Válaszoljon a kérdésekre!



**1. Válassza ki az alábbi meghatározások közül az adatmodell elemeit!**

- Adattábla
- Tulajdonság
- Egyed típus



**2. Jelölje meg az alábbi meghatározások közül melyik típusba sorolná a következő kapcsolatot!**  
**zenekedvelő személy - kedvenc - együttes**

- 1:1
- 1:N
- M:N
- terner



## Adatbázis-kezelő rendszer felépítése.

### Tanulási célok

A lecke tanulása után Ön képes lesz:

- felsorolni az Adatbázis-kezelő rendszer részeit.
- felsorolni azok tulajdonságait.

### Tevékenység:

Olvassa el a jegyzet 14. oldalától a 24. oldalig tartó részt!

Olvassa el a kiegészítő anyagot!

### Összefoglalás, további információk a tananyaghoz.

#### Adatbázis-kezelő fogalmi:

DDL : (Data Definition Language) adatdefiníciós nyelv, feladata az adatbázisok létrehozása, logikai struktúra megadása.

SDL : (Structure Definition Language) fizikai szerkezet leírása

DML : (Data Manipulation Language) adatmanipulációs nyelv, feladata a tárolt adatok visszakeresése, lekérdezése, módosítása

DBMS: (DataBase Manage System) mindez együtt az adatbázis-kezelő rendszer

DBMS rendszerek: DB2, ORACLE, MS ACCESS, MS-SQL

SQL: (Sutctured Query Language) struktúrált lekérdező nyelv. Bővebben ad Access lekérdezés szerkesztésénél lesz szó róla.

#### Adatbázis-kezelő részei:

-lekérdezés feldolgozó (lekérdezés, módosítás, struktúra módosítás)

-tárkezelő (fájlkezelő, pufferkezelő)

-tranzakció-kezelő (egy vagy több lekérdezés vagy módosítás csoportosítása)

-tárolt adatok, és metadatok

## Ellenőrző feladatok - Adatbázis-kezelő rendszer felépítése.

Válaszoljon a kérdésekre!



**1. Válassza ki az alábbi meghatározások közül mit jelent az SDL rövidítés!**

- A logikai szerkezet leírása.
- A fizikai szerkezet leírása.
- Struktúrált lekérdező nyelv.



**2. Válassza ki az alábbi meghatározások közül a tranzakció fogalmát!**

- A tranzakció a lekérdezések optimalizálását szolgáló eljárás.
- A tranzakció olyan műveletek csoportja, amit egymás után egy egységként kell végrehajtani.
- A tranzakció közvetlenül felügyeli az adatok lemezen való tárolását.



**3. Mit jelent a "metaadat" fogalom?**

- Az adatbázis egységnyi adata
- Visszakeresett tárolt adat
- Módosított tárolt adat
- Adatok a hozzá tartozó tulajdonságokkal, kapcsolatokkal együtt

## Modulzáró feladatok.

Oldja meg a következő feladatokat!

 **1. Válassza ki az alábbi meghatározások közül az adatmodell elemeit! (1 pont)**

- Egyed típus
- Kapcsolat
- Tárolás

 **2. Jelölje be az alábbiak közül melyik ismert DBMS rendszer! (1 pont)**

- ORACLE
- JAVA
- LINUX
- MsSQL

 **3. Válassza ki az alábbi meghatározások közül az Adatbázis-kezelő rendszer főbb részeit! (2 pont)**

- Felhasználói felület
- Lekérdezés-feldolgozó
- Jogosultság-kezelő
- Tárkezelő

 **4. Válassza ki a következő állítások közül mi igaz a tárkezelőre! (2 pont)**

- A lemezek általában lemezblokkokra vannak osztva.
- A pufferkezelő olvassa be egy állomány blokkjait.
- A pufferkezelő a memóriát kezeli.
- A tárkezelő nincs kapcsolatban a tranzakció feldolgozással.



**5. Válassza ki az alábbi meghatározások közül azokat a technikákat, amelyekkel a tranzakció-kezelés megvalósítható! (1 pont)**

- Rekord zárolás
- Napi adatmentés
- Atomosság
- Naplózás

## Normálformák, relációs modell

### Tanulási célok

A lecke tanulása után Ön képes lesz:

- meghatározások közül kiválasztani az alábbi fogalmak jelentését:

reláció, rekord, attribútum, mező, elsődleges kulcs, idegen kulcs.

- meghatározások közül kiválasztani a normálformák jellemzőit:

1NF, 2NF, 3NF

### Tevékenység:

Olvassa el a jegyzet 25. oldalától a 27. oldalig tartó részt, majd a kiegészítő tananyagot!

### Összefoglalás, további információk a tananyaghoz.

#### Relációs adatmodell:

Matematikai alapja van, az adatokat táblázatokban tárolja.

Három alapvető komponens jellemzi:

**Tábla:** az adatok sorokba, és oszlopokba vannak szervezve

**Adatmanipuláció:** SQL vagy relációs algebrai műveletek

**Adatintegritás:** bizonyos valós életből vett szabályok betartását ellenőrizhetjük

#### Részei:

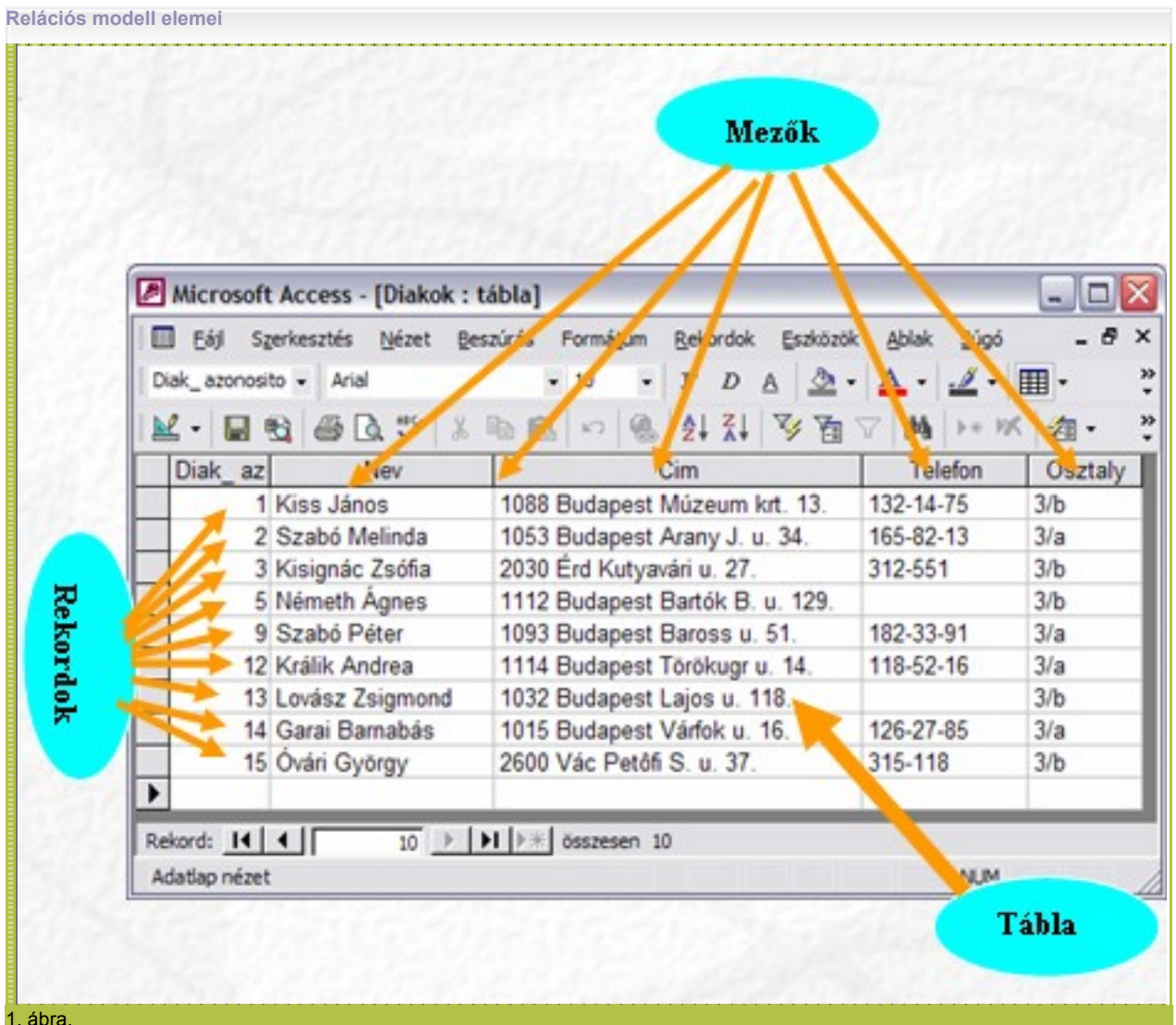
**Fájlok** vagy táblák, a közöttük lévő kapcsolatokkal.

A **tábla** rekordokat tartalmaz. (tábla helyett a reláció fogalom is használatos)

A **rekord** egy sor a táblában, amely összetartozó adatokat tartalmaz. (egyedek)

A **mező** egy oszlop a táblában, amely az rekordokban tárolt információk egy szelete.

(attribútumok, tulajdonságok leírására)



1. ábra.

A relációs modell egyik fontos fogalma a **kulcs**.

**Elsődleges kulcs** (primary key) a rekordok egyértelmű azonosítására szolgál, az **idegen kulccsal** (foreign key) hivatkozni tudunk egy másik táblázatra, amely a hivatkozóval logikai kapcsolatban van.

Az 1. ábrán feltüntetett adatokból a következőket vonhatjuk le:

- elsődleges kulcs: diák azonosító, mezőneve: Diak\_az
- mezőnevek, amik a diák tulajdonságait írják le: Nev, Cím, Telefon, Osztály.
- tábla: diákok adatait tartalmazza, tábla neve: Diakok
- érték: Kiss János, az első rekord Nev mezőjének tartalma.

**Normalizálás:** Táblázatsztémbontó relációs műveletek sorozata.

Előnyei:

- Csökken a tárolási igény.

- Megszűnnek a törlési, beszúrási, módosítási problémák.
- Logikailag áttekinthetőbb lesz.

Összesen 5 normálforma van, ebből az első 3 betartása kötelező érvényű.

### 1NF

- Oszlopok és sorok száma azonos
- Minden oszlop csak egy meghatározott értéket vehet fel.  
(Pl. név nem lehet MZ/X, születési nap nem lehet 7.5)
- Minden attribútum csak egyetlen értéket vehet fel a tartományból.  
(Pl. több kedvenc zene többértékű attribútum lenne, ekkor több külön rekordban tároljuk)
- Minden sorhoz egyedi kulcs tartozik, funkcionális függés  
(Pl. személy egyednél személyi szám)

### 2NF

a reláció 1NF - ben van

a nem kulcs attribútumok funkcionálisan *teljesen* függenek az elsődleges kulcstól.

### 3NF

a reláció 2 NF - ban van

Funkcionális függés csak az elsődleges kulcstól van, megszüntettük a tranzitív függéseket.

---

A normalizálás lépéseinek szemléltetésére készítettem egy pps bemutatót, amelyet a letölthető dokumentumok között talál!  
(nomalformak.pps)

---

## Ellenőrző feladatok - Normálformák

Válaszoljon a kérdésekre!



**1. Válassza ki az alábbiak közül a relációs modell részeit!**

Tábla

SQL

- Mező
- Tulajdonság

**2. Folytassa a következő mondatot! Az elsődleges kulcs ....**

- A tábla azonosítására szolgál.
- A rekord azonosítására szolgál.
- Nem lehet összetett kulcs.

**3. Az alábbiak közül mi az előnye a normalizálás folyamatának?**

- Csökken a hibalehetőség a tárolásban.
- Csökken a tárolási igény.
- Megszűnnek a módosítási problémák.
- Növekszik a redundancia.



## Az adatmodellezés három szintje.

### Tanulási célok

A lecke tanulása után Ön képes lesz:

- Meghatározni és/vagy listából kiválasztani az egyes szintek feladatát.
- Meghatározni a logikai modell és fizikai modell kialakításának lépéseit.

### Tevékenység:

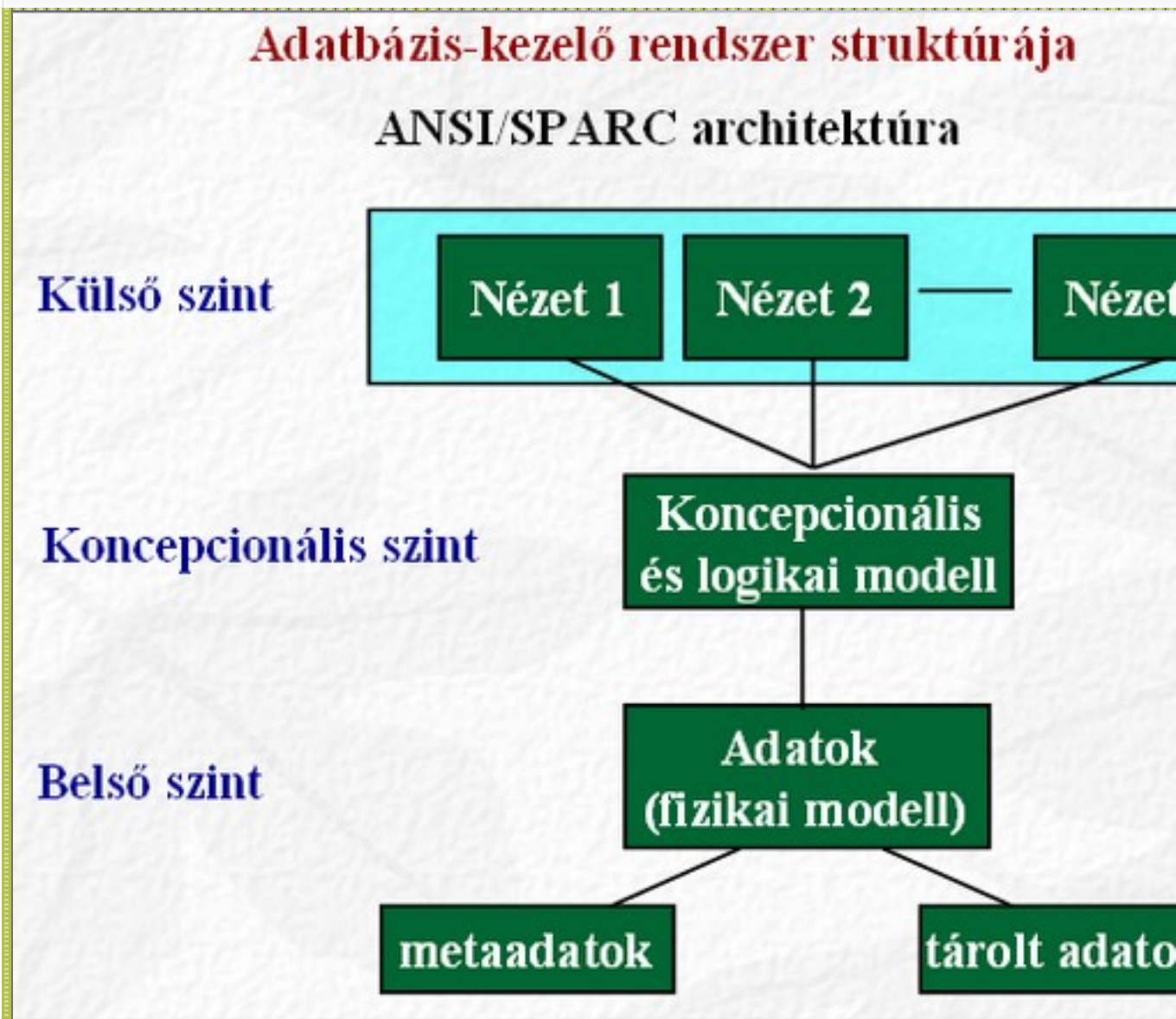
Olvassa el a jegyzet 27. oldalától a 39. oldalig tartó tervezési feladatokat!

### Összefoglalás, további információk a tananyaghoz.

Az adatmodell a következő 3 szintre bontható:

- 1. Külső szint:** azt írják le, hogyan látják a felhasználók az adatbázist.
- 2. Középső vagy koncepcionális szint:** teljes adatbázis szerkezete, és logikai modell.
- 3. Belső vagy fizikai szint:** adatok fizikai elhelyezkedése, és elérési módja.

Adatmodellezés szintjei



1. ábra.

**Logikai adatfüggetlenség:** adatbázis-kezelés legfontosabb követelménye, nemcsak az adatokat, hanem azok jellemzőit, és a kapcsolatokat leíró adatokat is tároljuk. (ezek a metaadatok)

Az adatszerkezet megváltoztatása csak a metaadatokban jelent változást, nem kell a felhasználói programot átírni.

**Fizikai adatfüggetlenség:** az adattárolási szerkezetek, és a hozzáférési módok változása nem vonja maga után a koncepcionális séma és a program változását.

A feladat megfogalmazása, nézetek összefésülése után a **logikai rendszerterv** elkészítése

következik:

- adatmodell és elemei (egyedtípusok táblázatos formában)
- statikus modell (egyedtípus szerkezeti lista lásd jegyzet 29. oldal)
- adatszótár (táblázat a tulajdonságtípusok előfordulásáról)
- kódjegyzék (azonosítók)
- adatmodell relációs elemzése:
  - \* funkcionális függés vizsgálata
  - \* egyedtípus mártix
  - \* adatmodell séma (lehet ER modell, EER modell)

**Fizikai rendszerterv:** Logikai modell leképezése, rekordtervek elkészítése, adattípusok, méretek meghatározása.

## Ellenőrző kérdések - Az adatmodellezés három szintje.

Válaszoljon a kérdésekre!



### 1. Válassza az alábbiak közül mit tartalmaz a külső szint!

- Azt, hogy hogyan látják a felhasználók az adatbázist.
- Adatok fizikai elhelyezkedését.
- Az adatok elérési módját.
- Logikai modellt, az adatbázis szerkezetét.



### 2. Mit jelent a fizikai adatfüggetlenség?

- Az adatbázis nem függ a programozótól.
- A tárolási szerkezetek módosítása nem vonja maga után a program módosítását.
- A kész adatbázis bárhová telepíthető.

**3. Melyek a logikai modell kialakításának helyes lépései?**

Helyes sorrend:

1. Egyed típusok meghatározása
2. Adatszótár
3. Statikus modell
4. Relációs elemzés
5. Kódjegyzék

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

## Modulzáró feladatsor

Válaszoljon a kérdésekre!



**1. Válassza az alábbiak közül mit tartalmaz a koncepcionális szint! (1 pont)**

- Azt, hogy hogyan látják a felhasználók az adatbázist.
- Adatok fizikai elhelyezkedését.
- Az adatok elérési módját.
- Logikai modellt, az adatbázis szerkezetét.



**2. Mit jelent a logikai adatfüggetlenség? (1 pont)**

- Az adatbázis nem függ a fizikai modelltől.
- Az adatszerkezetek módosítása csak a metaadatokban jelent változást.
- A kész adatbázis program bármikor átírható.



**3. Melyek a fizikai modell kialakításának helyes lépései? (2 pont)**

- |                                   | Helyes sorrend: |
|-----------------------------------|-----------------|
| 1. Logikai modell leképezése      | ___             |
| 2. Adattípusok meghatározása      | ___             |
| 3. Rekordtervek készítése         | ___             |
| 4. Tárolási méretek meghatározása | ___             |



**4. Válassza ki az alábbiak közül a relációs modell részeit! (1 pont)**

- Tábla
- Rekord

Tulajdonság Kapcsolat

**5. Folytassa a következő mondatot! Az idegen kulcs .... (1 pont)**

- A kapcsolódó tábla azonosítására szolgál.
- A kapcsolódó rekord azonosítására szolgál.
- ...ből egy táblában csak egy lehet.

## Adattípusok, kifejezések, operátorok.

### Tanulási célok

A lecke tanulása után Ön képes lesz:

- felsorolni az Access mező adattípusait,
- meghatározni azok értéktartományát,
- csoportosítani a kifejezéseket, operátorokat.

### Tevékenység:

Olvassa el a jegyzet 41.-44. oldalát, valamint a következő kiegészítő anyagot!

## Összefoglalás, kiegészítés a tananyaghoz:

Az Access adattípusait a következő ábra jól szemlélteti:



1. ábra.

A **szám** adattípus mezőméret szerint többféle lehet, tárolása 1 bájtól max. 17 bájtig történhet. Megválasztása mindig az adott feladattól függ, milyen adat tárolására fogjuk használni. Például ha tudjuk, hogy az adat lehetséges értékei pozitív egészek, és nem lehet több 255-nél, akkor elegendő a Bájt adattípus. ( Pl. általános iskolában osztály azonosítása (1-8), vagy magasság centiméterben megadva)

Általános szabály, hogy mindig a lehető legkisebb tárolási osztályt válasszuk, ne pazaroljuk a helyet.

A kifejezések, operátorok 4 csoportba oszthatóak:

- szöveges,
- aritmetikai,
- relációs,
- logikai.

A relációs kifejezések közül újdonságnak számítanak:

**Between** - intervallum meghatározása

**Like** - hasonlítás

**In** - előfordulás

**Is Not NULL** - NULL érték meghatározása

Ezekkel a kifejezésekkel bővebben foglalkozunk a beviteli ellenőrzéseknél (korlátozó feltételek, maszkolás), illetve a lekérdezés szerkesztésnél is.

A **Boole** algebrából jól ismert logikai műveleteket (AND, OR, NOT) egymásba is ágyazhatjuk, zárójelekkel szeparálva.

## Ellenőrző kérdések - Adattípusok

Válaszoljon a kérdésekre!



**1. Mekkora a szöveg adattípus karaktereinek maximális értéke? Írja be az üres helyre a megfelelő szám értéket!**

A szöveg adattípus max. \_\_\_\_\_ karakter lehet.



**2. Az alábbiak közül melyek MS Access adattípusok?**



- Szöveg típus, max. 64000 karakter.
- Logikai típus, igen/nem.
- Hiperhivatkozás



**3. Az alábbi műveletek közül melyik szolgál a szöveges adat összekapcsolására?**

- 
- &
- In
- \$



**4. Az alábbi relációs műveletek közül szintaktikailag melyik helyes?**

- 4 In 2,3,4
- 3 Between 2,5
- 3 Between 2 And 5
- 5 => 4

## Alapvető függvények.

### Tanulási célok

A lecke tanulása után Ön képes lesz:

- a következő függvényekhez hozzárendelni azok funkcióját és tulajdonságait: szövegkezelő függvények, dátum és idő függvények, matematikai, konverziós, és általános függvények.

### Tevékenység:

Olvassa el a jegyzet 45-55. oldalán található anyagot!

## Kiegészítés a tananyaghoz:

Az Access programozásához szükség van egy kódnyelvre, amit **VBA** szkript nyelvnek nevezünk, és a **MS Visual Basic** nyelv egy olyan változata, ami speciálisan az Access objektumaihoz illeszkedik.

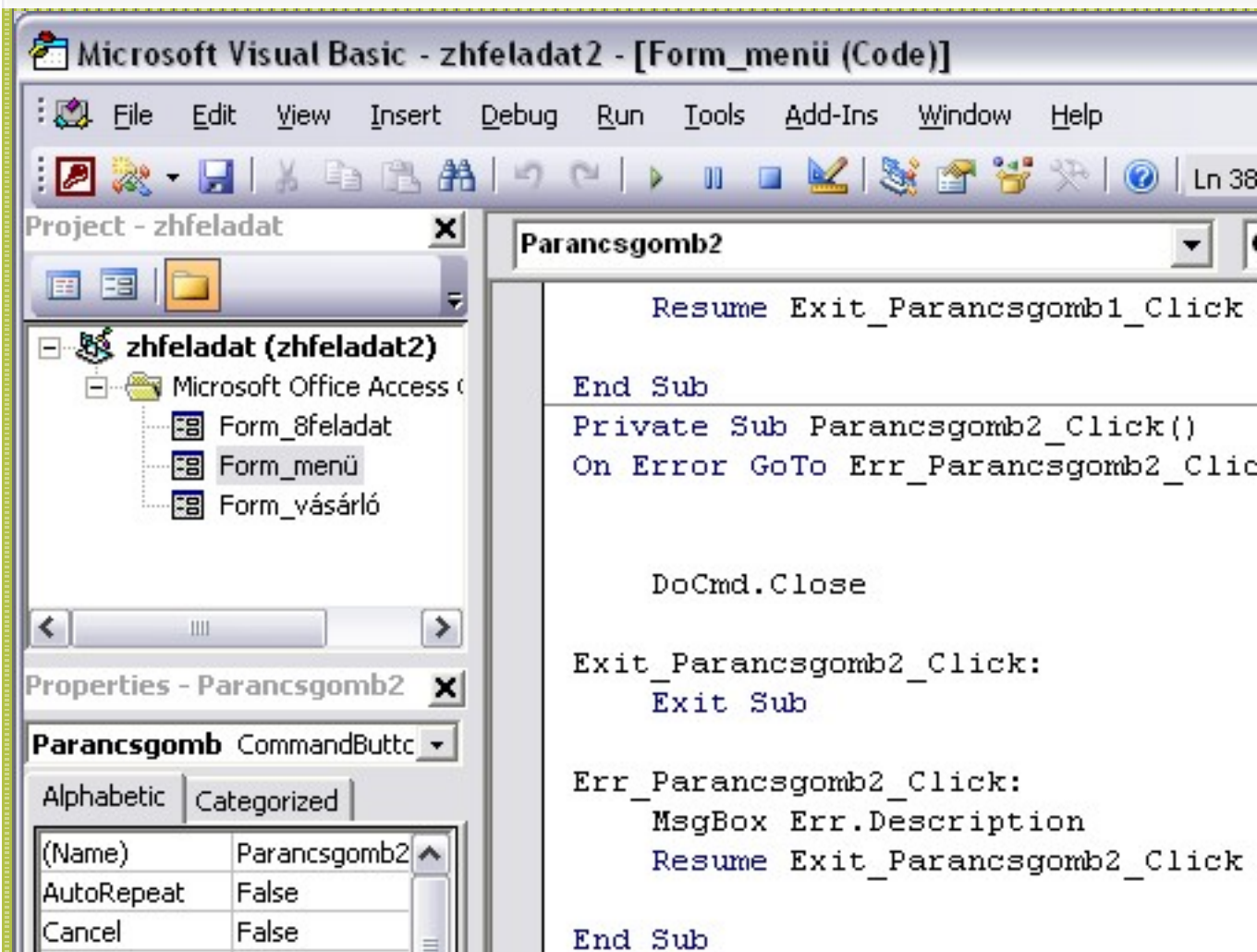
Miért kell az Accesst programozni?

Alapvetően nem kell, de néha szükségünk van olyan felhasználói funkciók előállítására, amit az Access önmagában nem tud lekezelni. Lehetnek ezek listákhoz kapcsolódó kódok, például a felhasználó egy beviteli mezőben elkezdi gépelni a "Balaton..." szót, és a legördülő menüből már csak a Balatoni települések listája látható. Ha tovább gépeli, és a "Balatonf..."-nél tart, akkor már csak néhány települést lát a szűkített listában. (Balatonfüred, stb.)

Ezt normál Access lekérdezéssel nem lehet megoldani, szükség van egy olyan kódra, ami figyeli a felhasználó által begépelte információt, frissíti a listát, *eseményvezérelt*, tehát ha újabb billentyű leütés történt, akkor újra frissít.

Ezen kívül még számos olyan eset van, amikor például egy űrlap nyomógombjához rendelünk valamilyen kódot. A következő ábra szemlélteti a nyomógomb (parancsgomb2) egérekattintásra (click) lefutó VBA kódját, Visual Basic felületen.

Access VBA kód egy parancsgombhoz.



1. ábra

Az ábrán látható, hogy a parancsgomb2 egy "menü" nevű úrlapon helyezhedik el. Ehhez hozzárendeltük a click metódust, ami akkor fut le ha a felhasználó a jobb egérgombbal egyszer kattint a nyomógombra. A különböző hibakezelő függvények hívása után a *DoCmd.Close* metódus fut le, ami nemes egyszerűséggel bazárja az úrlapot.

A példa szemléltetés céljából került ismertetésre, ilyen kódokat ebben a kurzusban nem fognak írni.

Néhány VBA függvényt viszont a jegyzetből érdemes megnézni, mivel ezek előfordulhatnak lekérdezésekben is.

Ilyen lehet például a dátum függvények közül a *Date()*, vagy *Year()* függvény, ha egy rendelési dátumot akarunk összehasonlítani a mai dátummal, és csak azokat listázni, amik korábbiak.

A függvényeket a jegyzetből kell megtanulni.

## Ellenőrző kérdések - Alapvető függvények.

Válaszoljon a kérdésekre!



**1. Az alábbiak közül melyik függvény adja vissza a rendszerdátumot és rendszeridőt?**

- Date()
- Now()
- Day()
- Year()



**2. Hogyan nevezzük az Access programozásához szükséges kódnyelvet?**

- SQL
- VBA szkript
- Visual Basic

## Modulzáró feladatsor

Oldja meg a következő feladatokat!



**1. Mekkora a feljegyzés adattípus karaktereinek maximális értéke? Írja be az üres helyre a megfelelő szám értéket! (1 pont)**

A feljegyzés adattípus maximum kb. \_\_\_\_\_ karakter lehet.



**2. Az alábbiak közül melyek MS Access adattípusok? (1 pont)**

- Számláló
- Egész számok
- Hiperhivatkozás



**3. Az alábbi műveletek közül melyik szolgál a szöveges adat összekapcsolására? (1 pont)**

- SUM
- +
- ln
- \$



**4. Az alábbi relációs műveletek közül szintaktikailag melyik helyes? (2 pont)**

- 4 ln (2 3 4)
- 3 Between 2,5
- "Melinda" Like "M\*\*"
- 5 => 4

---



**5. Melyik műveleti jel szolgál a nem egyenlő kifejezésére? (1 pont)**

Nem egyenlő jele: \_\_\_\_



**6. Az alábbiak közül melyik függvény adja vissza csak a rendszer dátumot? (1 pont)**

- Date()
- Now()
- Day()
- Year()

## Adatbázis létrehozása, megnyitása.

### Tanulási célok

A lecke tanulása után Ön képes lesz:

- MS Access programot elindítani, létező adatbázist betölteni.
- Új üres adatbázist létrehozni.
- Keretrendszer objektumait felsorolni.

### Tevékenység:

Olvassa el a jegyzet 57-60. oldalát, majd a következő kiegészítő anyagot!

### Kiegészítés a tananyaghoz:

#### 1. Az Access indítása XP-ben.

A Start gombra kattintva a Windows XP-ben az Access leggyakrabban a *Minden program/Microsoft Office/Microsoft Office Access 2003* útvonalon érhető el. Azért leggyakrabban, mert ez a "szokásos" telepítés, ha ettől eltérő helyre telepítjük, akkor elképzelhető hogy más könyvtárba került. Ha az asztalon nincs ikonja és nem találjuk a programot, akkor használjuk a keresést, vagy nézzük meg a vezérlőpulton a telepített programok közt.

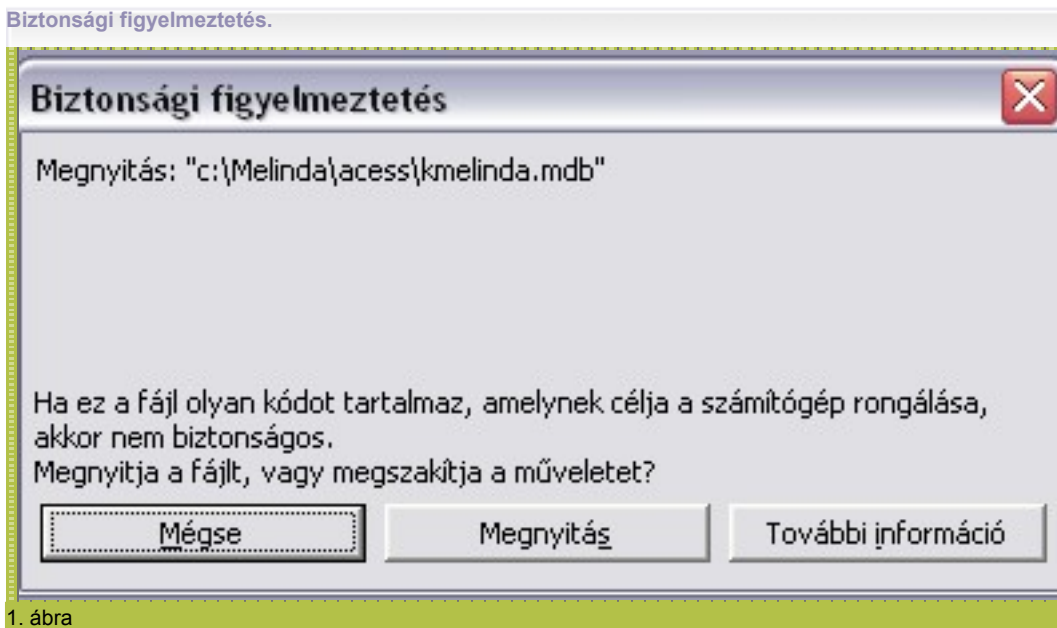
#### 2. Új üres adatbázis létrehozása

Ha a jegyzetben leírtakat követjük és létrehozunk egy új adatbázist, akkor meg kell mondanunk a helyét (C:/) és a nevét (BtgEllat), ezzel létrehoztuk a *BtgEllat.mdb* fájlt a meghajtón.

Az Access adatbázis-kezelő minden adatbázist egy **.mdb** kiterjesztésű fájlban tárol, beleértve a táblákat, kapcsolatokat, indexeket, lekérdezéseket, űrlapokat, jelentéseket is.

#### 3. Létező adatbázis megnyitása

Történhet úgy, hogy a kiválasztott .mdb fájlra kétszer kattintunk, ekkor az alapértelmezett program, az Access elindul és amennyiben az adatbázis már tartalmaz adatokat, kódokat a következő figyelmeztetést kapjuk megnyitás előtt:



Amennyiben a Megnyítást jóváhagyjuk, máris láthatóak az adatbázis objektumai.

A másik lehetőség, az Access indítása után a **Fájl menü/Megnyitás** paranccsal történik. Ekkor a kiválasztott adatbázist szintén ellenőrzi és megnyitja a rendszer.

#### 4. Adatbázis objektumok

- táblák
- lekérdezések
- űrlapok
- jelentések
- lapok
- makrók
- modulok

### Ellenőrző kérdések - Adatbázis létrehozása, megnyitása.

Válaszoljon a kérdésekre!



1. Válassza ki az alábbiak közül az Access adatbázis kiterjesztését!



.mdf



- .mdx
- .mdb
- .acs

**2. Jelölje meg az állításokat, igaz vagy hamis?**

A sablonokat azért alkalmazzuk, mert nem tudjuk milyen típusú adatbázist szeretnénk létrehozni.

igaz hamis

Az üres adatbázis létrehozásához meg kell adni a nevét és a helyét, hogy el tudjuk menteni.

igaz hamis

Az Access nem enged meg ékezetes neveket az adatbázishoz.

igaz hamis

**3. Az alábbiak közül jelölje meg melyek az Access adatbázis objektumai!**

- Mezők
- Fájlok
- Makrók
- Jelentések

## Táblák, indexek.

### Tanulási célok

A lecke tanulása után Ön képes lesz:

- új táblát létrehozni tervező nézetben,
- kiválasztani a megfelelő mezőnevet, adattípust rendelni hozzá,
- beviteli maszkot készíteni,
- mezőket indexelni,
- elsődleges kulcsot létrehozni egy táblán.

### Tevékenység:

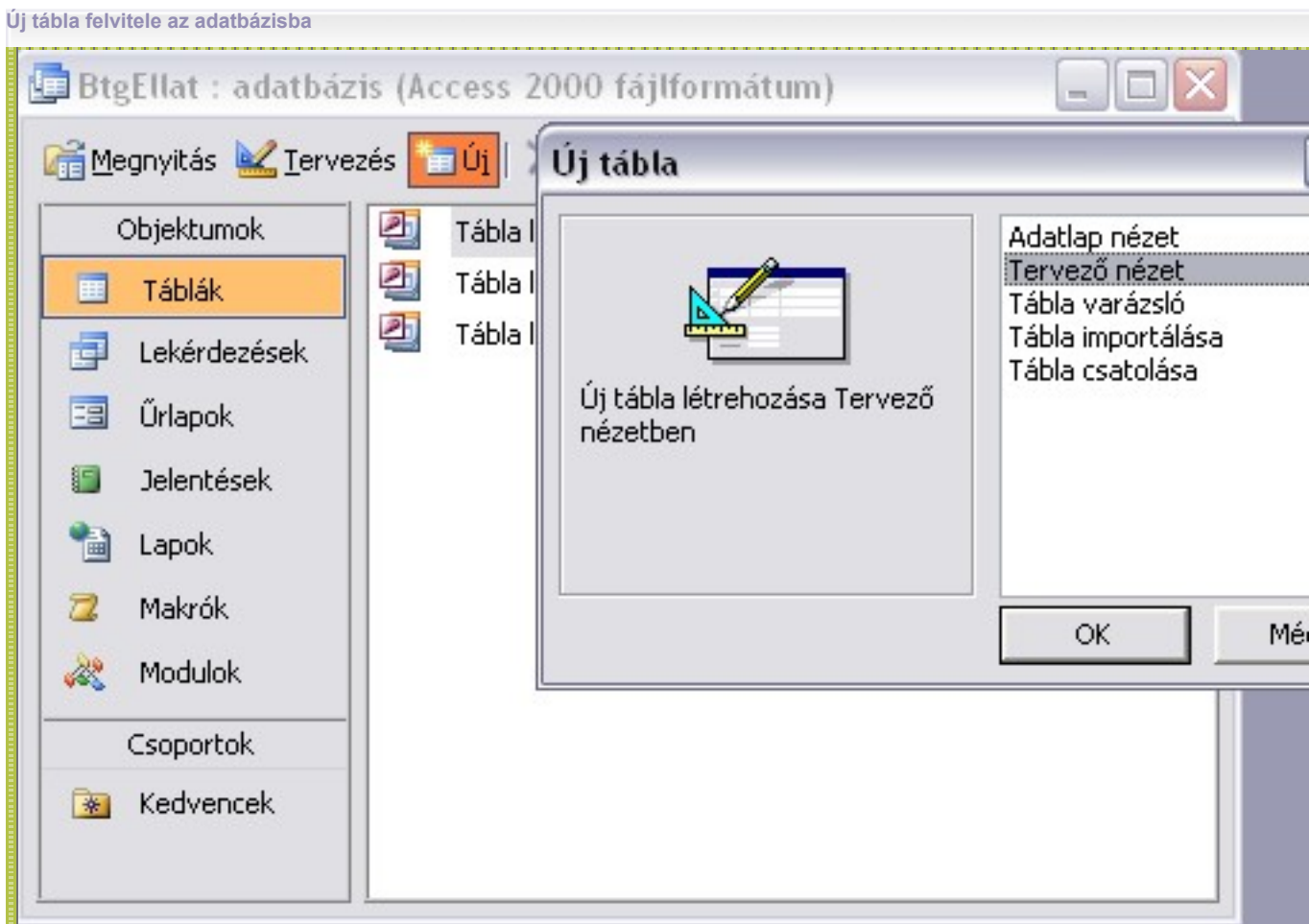
Olvassa el a jegyzet 61.-70. oldalát, majd a következő kiegészítő tananyagot!

A 71. oldaltól a 80. oldalig megtalálja az alappélda tábláinak leírását, ezeket hozza létre egy új adatbázisban a saját számítógépén!

### Kiegészítés a tananyaghoz:

A jegyzet elolvasása után most közösen létrehozzuk a Betegek táblát a BtgEllat adatbázisba.

1. Ha még nem tette volna meg, indítsa el az Accesst!
2. Az előző leckében tanult módon hozza létre a BtgEllat.mdb nevű üres adatbázist.
3. A megnyitott adatbázis-ablakban most az objektumok közül a "Táblák" legyen aktív, ekkor kattintson az alábbi képen narancs színű "Új" nyomógombra, és válassza az "Új tábla létrehozása tervező nézetben" parancsot.



1. ábra

Most az új táblába, (aminek majd akkor adunk nevet, amikor elmentjük, addig tábla1-nek hívják) rendre felvisszük a jegyzetben (70. oldal) lévő mezőneveket, és adattípust rendelünk hozzá.

Ezen kívül az "Általános kartonon" megadjuk a beviteli maszkot, és az indexet is. A megjelenítés beviteli mező lesz, a listával és a kombinált listával majd akkor foglalkozunk, ha a lekérdezéseket már elsajátítottuk!

### 1.lépés

Mezőnévhez írjuk be: TAJ, majd az adattípusa legyen: Szöveg.

*Megjegyzés:*

*Az egyes cellák között a TAB billentyűvel tudunk navigálni, vagy a kurzormozgató nyilakkal.*

Ha az adattípus mezőben vagyunk, automatikusan megjelenik egy lenyíló lista, ami segíti a választást. Ez látható a következő ábrán:



## 2. lépés

Vigyük fel a többi mezőnevet is, a 2. ábrán látható adattípussal!

## 3. lépés

Jelöljük ki újra a TAJ mezőnevet, és állítsuk be hozzá a könyvben található korlátozásokat!

Mezőmérethez gépeljük be a 9-et. (ez alapértelmezésben minden szövegre 50)

A beviteli maszk a következő lesz:000\000\000

Ez a tanultak alapján azt jelenti, hogy a 0 helyére számjegyet kell gépelni a bevitelkor, kötelező annyi darabot megadni, ahány 0 szerepel a maszkban, előjel nem megengedett. A - jeleknek is szerepelniük kell a bevitelkor, helyzetük kötött.

A beviteli maszk készítését varázsló is segíti, melyet a mező mellett lévő gomb segítségével lehet hívni:



A következő ábrán (3. ábra) látható pl. a TAJ szám bevitelére készített maszk, ezt lehet

módítani is, vagy kipróbálni a működését a próba soron.

Beviteli maszk varázsló

### Beviteli maszk varázsló

Melyik beviteli maszkot szeretné használni?

Használja a Próba mezőt, ha látni szeretné, hogyan működik a kijelölt maszk.  
A Beviteli maszk lista módosításához kattintson a Lista szerkesztése gombra.

Beviteli maszk:	Végeredmény:
Telefonszám	(12) 345-678 OR (1) 234-5678
<b>TAJ szám</b>	<b>123-456-789</b>
Személyi igazolvány száma	AH 212754
Irányítószám	1031 Budapest
Jelszó	*****
Hosszú idő	0:00:00

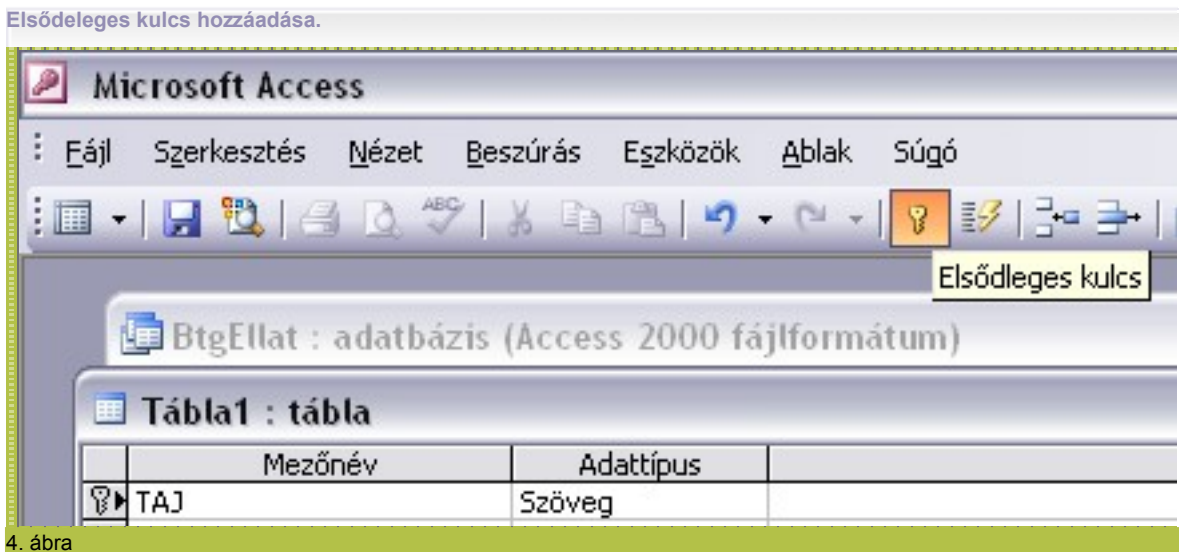
Próba:

3. ábra

#### 4. lépés

A táblát addig nem lehet elmenteni, amíg nincs elsődleges kulcs definiálva. A Betegek tábla kulcs a TAJ-szám lesz.

Kaatintsunk a következő ábrán látható kulcs jelű gombra az eszközsoron, miközben a TAJ szám az aktív a mezőnevek közül. (Magyarul a TAJ mezőnév előtt látható a kis háromszög.)



A nyomógomb két állapotot eredményez, ha sikerült rátennünk a mezőre az elsődleges kulcsot, akkor a háromszög mellett megjelenik a kis kulcs is, ha még egyszer megnyomjuk az eszközsor ikonját, akkor eltűnik. (Így lehet javítani, törölni kulcsokat!)

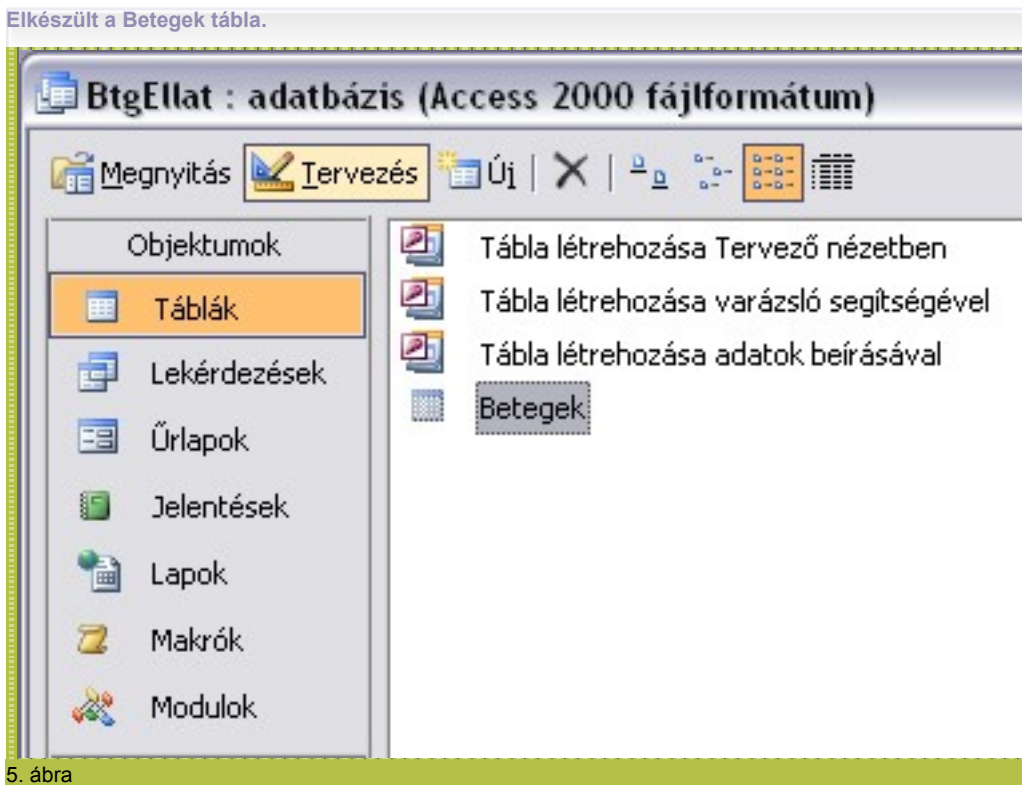
Ha több mezőből álló összetett elsődleges kulcs kell, például gyenge egyedtípusnál a kutya egyedét önmagában nem azonosítja a neve, csak a gazdája nevével együtt, akkor egérrel ki kell jelölni mind a két (vagy több) mezőt, azután kiválasztani az eszközsorról a kulcsot. Ilyen az alappéldában is lesz a Who nevű táblában!

## 5. lépés

Mentsük el a kész táblát, a **Fájl menü/Mentés** parancsával.

Ha egyszerűen mentés előtt bezárjuk az ablakot, akkor meg fogja kérdezni az Access, hogy mentjük-e a változásokat a tábla1-nél, tehát itt is tudunk menteni. Mentéshez most már csak a tábla neve kell: Betegek.

Ha végeztünk, akkor az adatbázisban megjelenik a Táblák objektumok között a Betegek tábla.



Ha a kész táblát módosítani szeretné, akkor a 5. ábrán látható Tervezés gombra kattintva visszajut az eredeti állapothoz, és akár mezőnevet, akár adattípust, akár indexet is módosíthat. Amíg nincsenek kapcsolatok a táblák közt, addig szabadon variálhat, utána már az indexeket és kulcsokat nehezebb változtatni, sokszor csak a kapcsolat megszüntetésével lehet. Utána persze újból felépíthető a kapcsolat az új kulcsokkal.

Amennyiben rendesen megterveztük az adatbázist, és itt már csak a program készítése folyik, ez nem is lehet probléma.

#### FELADAT

Ne felejtse el a jegyzetben lévő többi táblát is létrehozni, felvenni!

Táblák nevei:

Orvosok

Vizsgalat\* (segítség a kifejezőszerkesztő használata c. leckében)

VizsgTipus

Terkat

Bno

Who

Az egyes táblák indexeit szintén a mezők beállításait tartalmazó táblázatok után találja meg. Ennek egy része indexelt tulajdonság miatt automatikusan beállítódik, de az elsődleges kulcsot (primary key) ne felejtse el a táblára létrehozni!

#### További instrukciók:

1. A Vizsgalat nevű tábla, NAPLO mezője nem szöveg, hanem szám típusú.
2. A TAJ szám megjelenítése ugyanabban a táblában egy lekérdezés lesz, amit majd csak később tanulnak, ezért ezt sima beviteli mezőként vegyék fel, később majd módosítjuk.
3. Az alapértelmezett értékek és az érvényességi szabály beállításához olvassák el a következő leckét (kifejezésszerkesztő használata), segítség képpen.

A következő leckéhez ez lesz a kiindulás alapunk, jó munkát!

A mostani állapotot tükröző kész adatbázist a letölthető feladatok közé feltettem, ha valaki elakad, ott meg tudja nézni a beállításokat!  
(BtgEllat\_1verzio.mdb)

## Ellenőrző kérdések - Táblák, indexek.

Válaszoljon a kérdésekre!



1. Az alábbiak közül melyik karakter eredményezi a beviteli maszk készítésekor, hogy bármilyen karakter vagy szóköz következzen, kötelező bejegyzésként?

- L
- A
- &
- #



2. A következő beviteli maszk helyére milyen input érték kerülhet?  
A beviteli maszk: #999



- 20
- abc
- 2000
- 2000



**3. Ha azt szeretnénk elérni, hogy egy szám típusú mezőbe csak 1 és 10 közötti számot lehessen bevinni, akkor melyik mezőtulajdonságot kell beállítanunk?**

- érvényesítési szöveg
- alapértelmezett érték
- érvényességi szabály
- beviteli maszk



**4. Jelölje be a következő állításokat, igaz vagy hamis?**

- |                                                                            |      |       |
|----------------------------------------------------------------------------|------|-------|
| Az indexelt tulajdonság segítségével egy egymezőös indexet állíthatunk be. | igaz | hamis |
| A számláló típusú mező lehet kötelező kitöltésű.                           | igaz | hamis |
| A feljegyzés típusú mező nem indexelhető.                                  | igaz | hamis |

## A kifejezésszerkesztő használata

### Tanulási célok

A lecke tanulása után Ön képes lesz:

- Az alapértelmezett érték beállításához beépített függvényt rendelni.
- kifejezéseket szerkeszteni érvényességi szabályokhoz

### Tevékenység:

Olvassa el a következő kiegészítő tananyagot!

### Kiegészítés a tananyaghoz:

Az alappéldában először a 74. oldalon találkozunk azzal a problémával, hogy a DATUM nevű mező alapértékét be kell állítani a mai dátumra.

Ezt az alapvető függvények részénél már megtanultuk a Now() függvénnyel lehet lekérdezni.

Ennek a legegyszerűbb módja, ha felől tudjuk a megoldást, egyszerűen begépeljük az

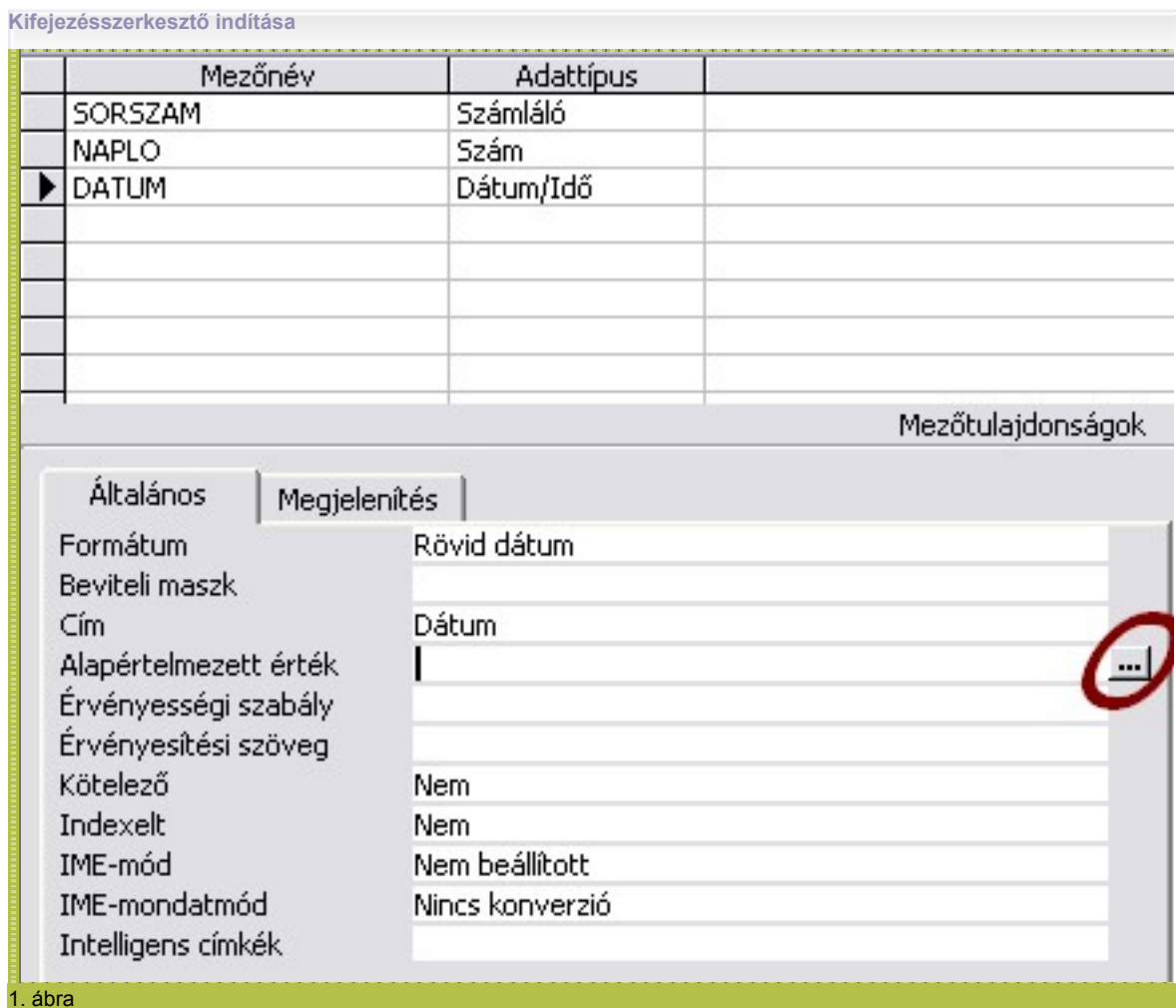
Általános karton Alapértelmezett érték mellé.

Van azonban az Accessnek egy beépített segítsége erre az esetre, amit

**kifejezésszerkesztőnek** hívnak.

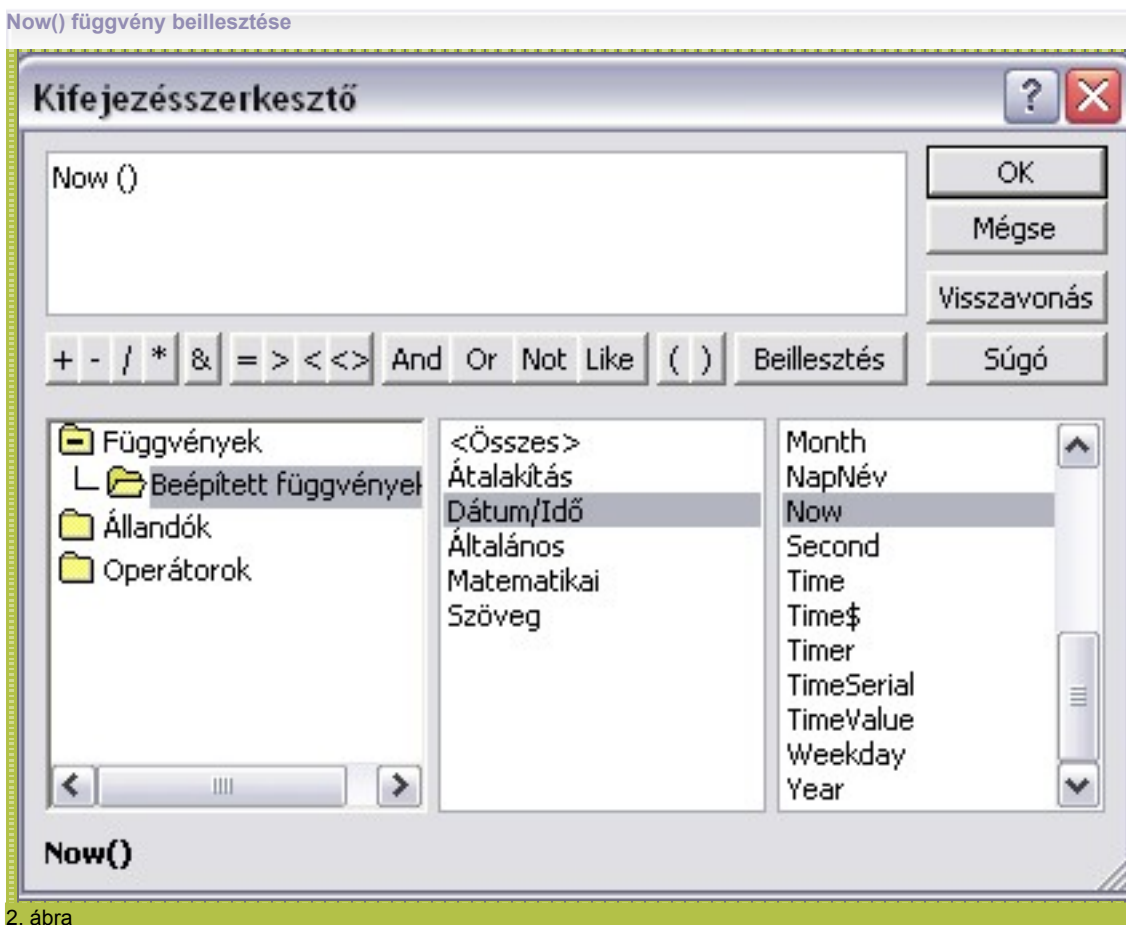
A következő ábrán látható, hogy a tábla tervezése közben amint az alapértelmezett érték mellé tettem a kurzort, megjelenik egy három pontot tartalmazó nyomógomb a beviteli mező mellett.

Erre kattintva érhetjük el a kifejezésszerkesztőt.



Az 1. ábrán a bekarikázott gombra kétszer kattintunk, és a következőket kell kiválasztani a mai dátum beállításához:

- először a 2. ábrán látható ablakban nyissa le a *Függvények* mappát, jelölje ki a *Beépített függvények* mappát.
- a második oszlopban a *Dátum/Idő* függvénycsoportot válassza ki
- a harmadik oszlopban pedig a megjelenő függvények közül a *Now()* függvényt válassza ki.
- kattintson a *Beillesztés* gombra, majd az *OK* gombbal térjen vissza a tábla tervezéshez.



2. ábra

A beépített függvényeken kívül találunk még itt úgynevezett *Állandókat*, ezek az:

"" - üres karakterlánc, hamis, igaz, null értékek.

és az *Operátorokat*, amelyek közül választhatunk a kifejezés szerkesztéséhez. A leggyakrabban használatos operátorokat a szerkesztő mező alatt nyomógombként is megtalálja.

Nemcsak az alapértelmezett értékhez tudjuk használni a kifejezésszerkesztőt, hanem az érvényességi szabály beállításához is. (jegyzet 65. oldal)

### Feladat

Egy szám típusú mező esetén állítsuk be, hogy 10 és 100 közé essen a bevitt adat!

### Megoldás 1.

Álljunk az érvényességi szabály cella mellé, és indítsuk el a kifejezésszerkesztőt!

Az operátorok közül válasszuk ki kettős kattintással a  $\geq$  operátort, majd adjuk meg a szerkesztő mezőben, hogy 10.

Ezután jön az AND kifejezés, mivel kettős feltételünk van, és mindkettőnek teljesülni kell.

Majd jöhet a <= operátor, majd újra begépeljük, a konstans értéket: 100

A végeredmény a következő lesz: "<<Kif>>"=10 And "<<Kif>>"<=100

Az operátorral együtt a szerkesztőbe került a "Kif" bejegyzés, ami azt jelenti, hogy a felhasználó által beírt kifejezést hasonlítja 10-hez, és 100-hoz. Ha a kettő közé esik, akkor elfogadja, ha nem akkor az **Érvényességi szöveget** fogja hibaüzenetként közölni.

### Megoldás 2.

Hasonló az elsőhöz, csak most a BETWEEN összehasonlító operátort használjuk.

Alapeasteben a következő kerül a szerkesztőbe: Between Kif And Kif

Álljon a szerkesztőbe a Kif szövegre, és helyette gépelje be, hogy 10. Majd a másik helyett a 100 értéket.

Végeredmény: Between 10 And 100

## Ellenőrző kérdések - Kifejezésszerkesztő

Válaszoljon a kérdésekre!



1. Az alábbi érvényességi szabályok közül szintaktikailag melyik helyes?

- Between 10 or 100
- =>10
- <=10



2. Milyen kategóriákkal találkozhatunk, ha megnyitjuk a kifejezésszerkesztőt?

- aritmetikai operátorok
- kész sablonok
- beépített függvények
- beépített kifejezések



## Kapcsolatok, hivatkozási integritás.

### Tanulási célok

A lecke tanulása után Ön képes lesz:

- kapcsolatokat definiálni táblák között,
- megállapítani egy táblán az elsődleges, és idegen kulcsokat,
- listából kiválasztani a kaszkádolt törlés, kaszkádolt frissítés fogalmát

### Tevékenység:

Olvassa el a jegyzet 81-82. oldalát, majd a következő kiegészítő tananyagot!

### Összefoglalás, kiegészítés a tananyaghoz:

Miután az adatbázisban létrehoztuk a táblákat, meg kell adni a köztük lévő kapcsolatokat is.

*A kapcsolat a kulcsmezők értékeit rendeli egymáshoz.*

**1:1** kapcsolat jön létre, ha mindkét kapcsolt mező elsődleges kulcs, illetve ha egyedi index.

**1:N** kapcsolat jön létre, ha csak az egyik elsődleges kulcs, illetve ha csak az egyik egyedi index.

**N:N** kapcsolat jön létre, ha két 1:1 kapcsolat, egy harmadik tábla részvételével, itt összetett elsődleges kulcs van a két tábla idegen kulcsából.

**Határozatlan kapcsolat** jön létre, ha nem elsődleges kulcs és nem egyedi index a két kapcsolt mező.

**Hivatkozási integritás:** olyan szabályok rendszere, amely a táblában lévő rekordok közötti kapcsolatot érvényesíti, a kapcsolódó adatokat nem lehet véletlenül törölni, vagy módosítani.

Kapcsolt mezők **kaszkádolt frissítése** engedélyezve van: az elsődleges kulcs érték automatikusan frissíti az összes kapcsolódó rekordban lévő megfelelő értéket. (Pl. valakinek a neve megváltozik egy táblában, akkor a személyi szám mint kulcs alapján kapcsolt táblában lévő összes érintett rekordban is megváltozik a neve.)

Kapcsolt mezők **kaszkádolt törlése** engedélyezve van: az elsődleges táblában egy rekord törlése a kapcsolódó táblában a kapcsolódó rekordok törlését engedélyezi.

*Figyelem! Ez ellentmond a hivatkozási integritás megőrzésének, tehát csak akkor alkalmazzuk, ha biztosak vagyunk benne, hogy nem okoz hibát!* (Pl. nem lehet egy főkönyvi számot törölni a szülő táblából, ha valaki a gyerek táblában erre a számra könyvelési tételt dolgozott fel. Előbb törölni kell a gyerek rekordokat, majd a szülő tábla rekordját.)

A következőkben tekintsük át, a **btgellat\_1verzio.mdb** kiindulás adatbázisunkon (letölthető dokumentum!) hogyan is kell a kapcsolatokat létrehozni, a végeredmény a könyvben látható 4.17 ábra lesz.

### 1. Kapcsolat definiálása.

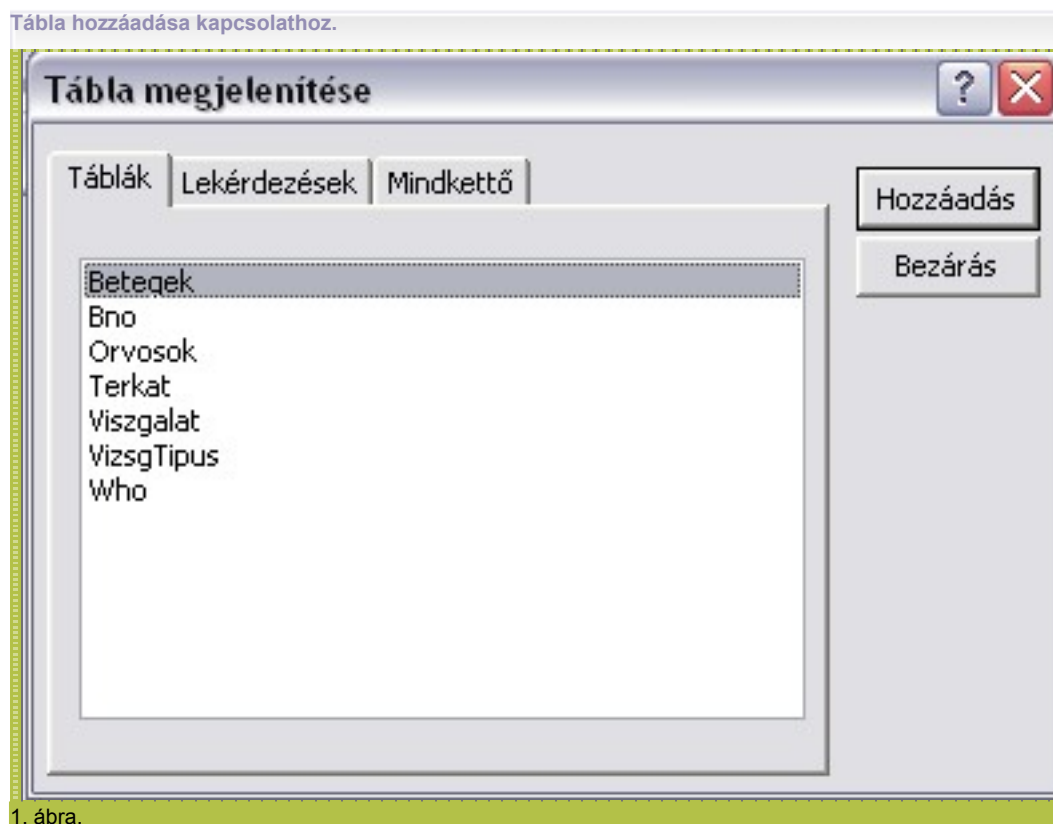
Válassza az **Eszközők menü/Kapcsolatok** menüpontot, vagy az eszközsor megfelelő ikonját:



A tábla megjelenítése ablakban a következőképpen lehet a táblákat a kapcsolatok ablakhoz hozzáadni.

Vagy egyenként a tábla nevére és utána a Hozzáad gombra kattint, vagy egérrel kijelöli az összes táblát egyszerre és a Hozzáad gombra kattint.

A következő ábra ezt az ablakot mutatja.



1. ábra.

Itt minden táblára szükség van, tehát az összeset adja hozzá.

A megjelenő kapcsolat ablakban a táblákat láthatja a könyv 4.17 ábrájának megfelelően, a



tábla neve a fejlécben szerepel, a mezőnevek a kis ablakban egymás alatt. Az elsődleges kulcsok félkövén látszanak. Tanulmányozza az ábrát!

A legtöbb kapcsolatban a **Vizsgalat** tábla szerepel, ezért célszerű ezt az egérrel középre helyezni, majd a jobb alsó sarkát megfogva méretezni akkorára, hogy ne legyen a szélén görgetősáv.

Ha elrendezte a táblákat jöhet a következő lépés, építsük fel a Betegek és a Vizsgalat tábla közötti kapcsolatot, a TAJ mező alapján.

Mindíg az elsődleges kulcs felől húzzuk a kapcsolatot a kapcsolat tábla felé, így nem lehet elrontani.

A következő bemutatóban nézze meg a művelet végrehajtását, majd oldja meg a saját tábláján is!

---

Az online tananyag ezen pontján interaktív, vagy multimédia objektumot talál. Kérem nézze meg az online tananyagot!

---

## Ellenőrző kérdések - Kapcsolatok, hivatkozási integritás.

Válaszoljon a kérdésekre!



**1. Állapítsa meg milyen kapcsolat hozható létre, ha csak az egyik mező elsődleges kulcs, illetve ha csak az egyik egyedi index.**

- 1:1
- 1:N
- N:N
- határozatlan



**2. Állapítsa meg igaz vagy hamis az állítás!**

- |                                                                      |      |       |
|----------------------------------------------------------------------|------|-------|
| A kapcsolódó mezőknek kivétel nélkül ugyanaz az adattípusuk.         | igaz | hamis |
| Mindkét kapcsolt táblának ugyanahhoz az adatbázishoz kell tartoznia. | igaz | hamis |

Számláló mező kapcsolatba hozható hosszú egész típusú mezővel.

igaz

hamis



### 3. Állapítsa meg a tagmondatok közötti összefüggést!

Ha a hivatkozási integritás be van kapcsolva, és a kaszkádolt törlés is engedélyezett, akkor az elsődleges táblában lehet rekordot törölni, a kapcsolódó táblában viszont nem törlődik.

A B C D E

## Lekérdezés alapismeretek.

### Tanulási célok

A lecke tanulása után Ön képes lesz:

- felsoroli a lekérdezések nézeteit,
- felsorolni a lekérdezések típusait,
- A SELECT SQL utasítás részeihez hozzárendelni a funkcióit,
- lekérdezést szerkeszteni az Access-ben,
- lekérdezést futtatni.

### Tevékenység:

Olvassa el a jegyzet 87.-114. oldalát, majd a következő kiegészítő tananyagot!

A jegyzetben feladatok is vannak, ezeket oldja meg!

### Összefoglalás, kiegészítés a tananyaghoz:

#### Lekérdezések nézetei:

- tervező nézet,
- SQL nézet,
- adatlap nézet.

#### Lekérdezések típusai:

- választó
- keresztábrás
- táblakészítő
- frissítő
- hozzáfűző
- törlő.

#### SELECT SQL utasítás általános szerkezete leegyszerűsítve:

SELECT mezők FROM tábla

[WHERE..] - feltétel a rekordok szűrésére

[GROUP BY..] - csoportosítás

[HAVING..] - csoportosítás szűrése

[ORDER BY..] - rendezés, növekvő, csökkenő

### 1. Lekérdezés szerkesztés

Az Access lehetővé teszi a felhasználó elől az SQL utasítások elrejtését, vagyis készíthetünk úgy lekérdezést tervező nézetben, hogy mit sem sejtünk az SQL utasítások szintaktikájáról. Mégis fontos eleme ez az adatbázis-kezelésnek, hiszen számos már rendszer is használja az SQL lekérdező nyelvet.

#### FELADAT

Kezjük egy egyszerű lekérdezéssel, kérdezzük le a Betegek táblából a beteg nevét, és születési dátumát, névsor szerint növekvő sorrendben!

#### MEGOLDÁS

1. Lekérdezés objektumtípus kiválasztása után kattintson az ÚJ feliratú gombra.
2. Tervező nézetben adja hozzá a Betegek táblát. (hasonlóan ahogy a kapcsolatokhoz is hozzáadta korábban!)
3. Adjuk meg az eredmény tábla oszlopait: BNEV, SZDAT
4. Megjelenítés legyen bejelölve mindkettőnél.
5. Rendezés a BNEV alatt legyen növekvő.
6. Mentsük el a lekérdezést első néven. Kattintson a mentés ikonra az eszközsoron és adja meg a nevet, vagy egyszerűen zárja be az ablakot, és ekkor adja meg a nevet.
7. Futtassa a lekérdezést: Lekérdezés objektumtípus kiválasztása után az első nevű lekérdezésen kettőt kattintson.

Nézze meg a következő videót, ha egyedül nem ment!

---

Az online tananyag ezen pontján interaktív, vagy multimédia objektumot talál. Kérem nézze meg az online tananyagot!

#### FELADAT

Nézzük meg az iménti lekérdezést tervező nézetben, SQL nézetben, és adatlap nézetben!

#### MEGOLDÁS

1. Az első nevű lekérdezést jelölje ki, majd kattintson a Tervezés gombra.
2. Ekkor a lekérdezés beállításait láthatja tervező nézetben.
3. Válassza a **Nézet menü/SQL nézet** parancsát, akkor az SQL utasítást látja:

```
SELECT Betegek.BNEV, Betegek.SZDAT  
FROM Betegek  
ORDER BY Betegek.BNEV;
```

4. Válassza a **Nézet menü/Adatlap nézet** parancsát, akkor a lekérdezés eredményét látja.  
(Ez nem módosítható)

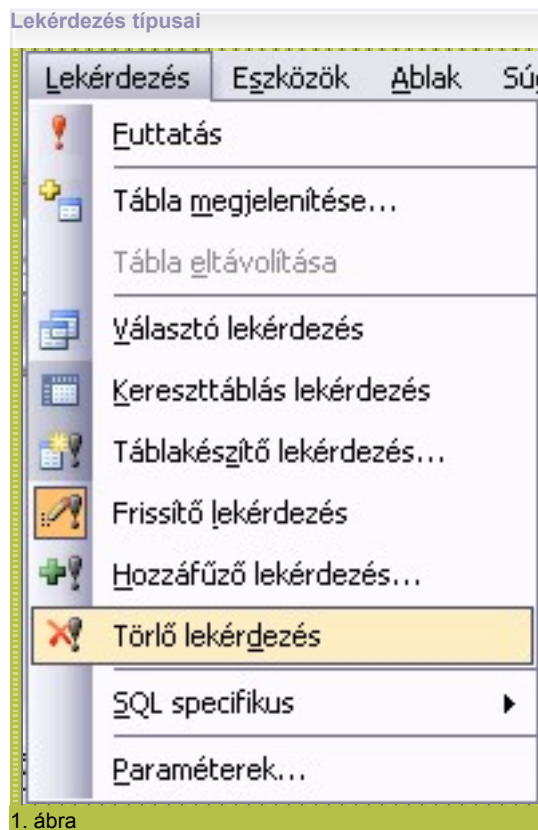
### FELADAT

Készítse el a jegyzet alapján a 4.2.2.8 fejezetben leírt, és megoldott lekérdezéseket is!

A leggyakrabban a választó, a paraméteres választó lekérdezést, és a törlő lekérdezést fogjuk használni.

A lekérdezések típusát szintén a tervező nézeten keresztül, a **Lekérdezés menüben** tudja változtatni, ezt mutatja a következő ábra:

A jegyzetben található lekérdezéseket a btgellat\_3verzio.mdb fájl tartalmazza.



1. ábra

## 2. Mező megjelenítésének szerkesztése Tábla/Lekérdezés-sel

Térjünk egy kicsit vissza az alapfeladathoz. Mivel a lekérdezéseket most sajátítottuk el, a Vizsgalat tábla adatbevitelkor kérhetünk segítséget a már meglévő tábláktól. Nem tudjuk fejből

a betegek TAJ számát, de a Betegek tábla fel van töltve, így a beviteli mező helyett kérjük kombinált listát, és ki lehet választani az adatokat!

Most el tudja készíteni a jegyzet 74. oldalán a TAJ mező megjelenítéséhez szükséges lekérdezést.

1. Válassza a táblák objektumok közül a Vizsgalat táblát, majd a Tervezés gombra kattintson.

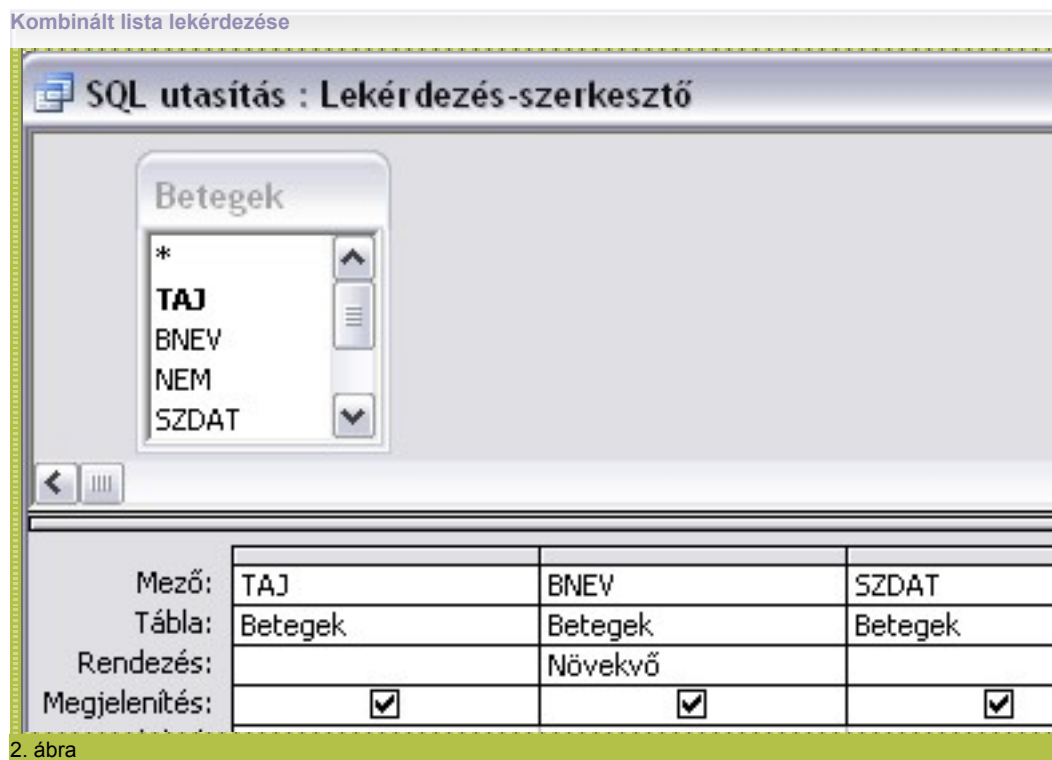
2. A TAJ mező megjelenítés kátróján állítsa be a következőket:

- A vezérlő elem: Kombinált lista

- A sorforrás típusa Tábla/lekérdezés

- A sorforrásnál a pontozott gombra kattintva elindul a lekérdezés tervező nézetben. Itt a következő ábrán látható mezőnevek szerepelnek:

TAJ, BNEV, SZDAT. A BNEV szerint rendezett.



Bezáráskor nem kell nevet adni a lekérdezésnek, a megfelelő SQL utasítás bekerül a sorforrás sorába.

Állítsa be a 74. oldalon lévő többi megfelelő értéket is, majd mentse a módosításokat.

Ezután könnyebb lesz feltölteni a táblát. Ezt mutatja a következő ábra.

Tábla feltöltése kombinált lista panellel.

Viszgalat : tábla							
	Sorszám	Napló	Dátum	Idő	TAJ szám	Vizsg. típusa	Té
▶	(Számológó)	00000	2007.07.12.	12:56		0	
					TAJ-szám	Név	Sz
					111222333	Kiss Anita	200
					123111222	Kiss Bea	195
					123456789	Krankovits Melinda	197
					123456111	Lusta Béla	195
					111333444	Lusta Béláné	195
					111222444	Nagy Miska	194

3. ábra

## Ellenőrző kérdések - Lekérdezések

Válszoljon a kérdésekre!



1. Az alábbi lekérdező utasítás végrehajtásakor mi lesz az eredmény?

```
SELECT * FROM Betegek
WHERE Betegek.BNEV LIKE "A**"
```

- A betegek tábla összes rekordját listázza.
- A betegek táblából a betegek neveit listázza.
- A betegek táblából minden olyan beteget listáz akinek a neve A-betűvel kezdődik.
- A betegek táblából minden olyan beteg nevét listázza akinek a neve A-betűvel kezdődik.



2. A lekérdezés melyik záradékában lehet megadni a tábla nevét?

záradék: \_\_\_\_\_



3. Az alábbiak közül milyen lekérdezés típusokat ismer?

- kijelölő
- választó
- keresztirányú
- törlő



**4. Melyik nézetben nem módosítható a lekérdezés?**

- SQL nézet
- adatlap nézet
- tervező nézet



**5. Milyen kifejezést kell beírni a kipontozott helyre, ha nem a lekérdezett rekordokat, hanem csak a rekordok darabszámát szeretnénk látni?**

**SELECT ..... FROM Betegek;**

**(A választ csupa nagybetűvel adja meg!)**

kifejezés: \_\_\_\_\_



## Adatbevitel a táblába.

### Tanulási célok

A lecke tanulása után Ön képes lesz:

- táblát feltölteni rekordokkal,
- adattípusnak és korlátozásoknak megfelelő adatot bevinni a mezőkbe,
- kapcsolt tábla adatait felhasználni a bevitelhez.

### Tevékenység:

Olvassa el a következő kiegészítő tananyagot!

### Kiegészítés a tananyaghoz:

Az adatbázisunk felépítésének következő lépése, miután a kapcsolatokat beállítottuk, az adatok felvitele a táblákba.

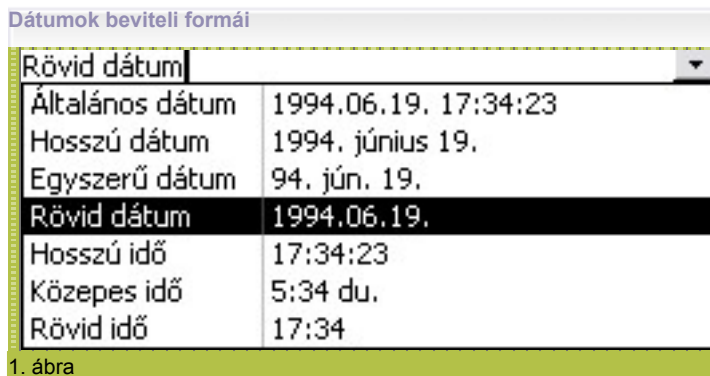
Lekérdezni adatokat csak azután tudunk, ha a táblák nem üresek.

Ehhez nem árt tudni, hogy az egyes mezők milyen értékeket vehetnek fel, például szöveges mezőbe szöveget kell írni, dátum mezőbe dátumot, beviteli maszkkal korlátozott adatbevitelnél csak adott karaktereket.

Nyilván nem emlékszünk fejből mindenre, ezt tartalmazza a rendszerterv is, jelen esetben a könyv. A jól felépített, felhasználóbarát bevített megkönnyítik a megfelelő *hibaüzenetek* is. ( pl. az érvényesítési szöveg ilyen)

Legegyszerűbb dolgunk akkor van ha a mezőnevek megfelelően "beszédese", a **Beteg** táblában a beteg nevét a **Név** mező alá kell írni, stb. A dátumokkal már meggyűlhet a bajunk, de a tábla tervező nézetében bármikor ellenőrizhető a beállítás, lásd *Táblák, indexek c. lecke*.

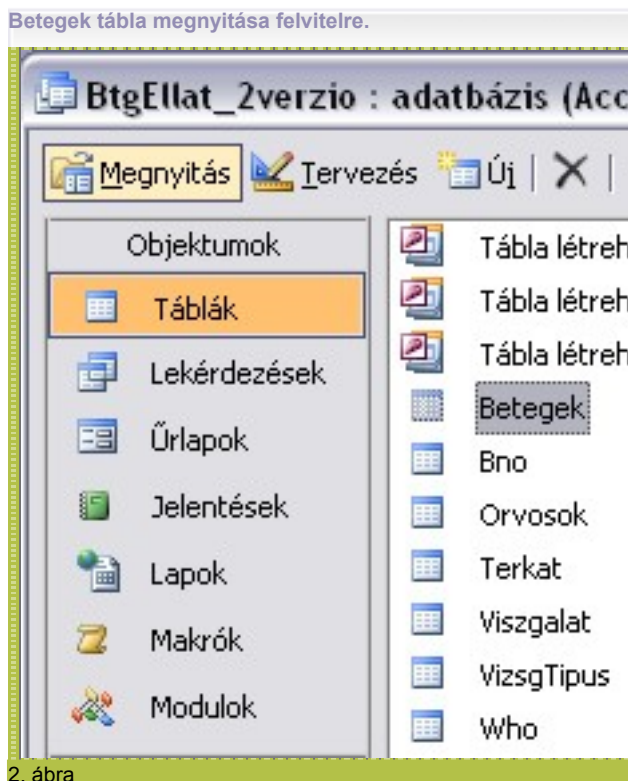
A következő ábra mutatja a dátumok beviteli formáit, amennyiben az adat megjelenítése beviteli mező, ezeket kell követni:



### 1. Betegek tábla feltöltése

Nyissa meg az adatbázist, ha sajátja nincs kész, akkor a *btgellat\_2verzio.mdb* állománnyal dolgozzon!

Ezután a táblák közül legyen aktív a Betegek tábla (egérrel egyszer kattint), majd a **Megnyitás** gombra kattintva táblázatosan fel lehet vinni az adatokat. (A tábla nevén a dupla kattintás ugyanígy megnyitja a táblát.)



Miután a táblát megnyitottuk a kurzormozgató billentyűkkel tudunk mozogni a cellák között, a megfelelő adatokat be kell gépelni.

A következő ábrán látható, hogy a **TAJ** szám megadásánál segít a beviteli maszk, a **NEM**

megadása jelölőnégyzettel törénik (ez igen/nem típusú adatnál alapértelmezés) így a nőket igen-re állítjuk, a férfiakat nem-re. Ezekon kívül a születési dátum rövid dátum formátumú, ezért az 1. ábra alapján tudjuk, hogy ponttal elválasztott magyar formátumú dátum kell ide.

Vajon a rendszer magától észreveszi, hogy nincs 13. hónap, vagy 32. nap? A válasz természetesen igen, próbálja ki!

#### Betegek rekordjai

Betegek : tábla							
		TAJ-szám	Név	Nem	Szül. dátum	Irányítószám	Település
▶	+	111\222\333	Kiss Anita	<input checked="" type="checkbox"/>	2002.10.14.	9023	Győr
	+	111\222\444	Nagy Miska	<input type="checkbox"/>	1948.02.03.	8700	Sehonnán
	+	111\333\444	Lusta Béláné	<input checked="" type="checkbox"/>	1956.05.18.	9002	Móvár
	+	123\111\222	Kiss Bea	<input checked="" type="checkbox"/>	1950.12.12.	9023	Győr
	+	123\456\111	Lusta Béla	<input type="checkbox"/>	1955.05.12.	9023	Győr
	+	123\456\789	Krankovits Meli	<input checked="" type="checkbox"/>	1975.05.29.	9178	Hédervár
*				<input checked="" type="checkbox"/>			

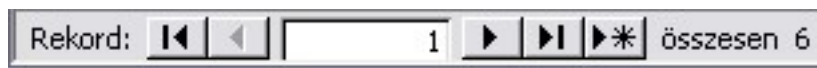
3. ábra

#### FELADAT

Vegye fel a 3. ábrán látható rekordokat!

A rekord előtti kis háromszög mutatja most is az aktuális rekordot, a \* pedig az új üres rekord felvitelére szolgál.

Az ablak alján látható navigációs sávon látható, hogy összesen 6 rekordunk van.



#### 2. Kapcsolódó tábla rekor felvitele

A táblák kapcsolatainál ellenőrizhetjük, hogy a **Betegek** tábla a TAJ mező alapján kapcsolatban van a **Vizsgalat** táblával. (könyv, 4.17 ábra)

Ennek megfelelően, a bevitelkor észrevehették, hogy a rendszer már most felkínálja az adott beteg rekorje mellé a vizsgálatait is rögzíteni lehet. Ezt mutatja a következő ábra, amely ablak rekord mellett lévő + jelet lenyitva jelenik meg. Mivel 1:N kapcsolat van, egy betegnek több vizsgálata is lehet!

## Kapcsolt tábla adatfelvitele

Betegek : tábla							
	TAJ-szám	Név	Nem	Szül. dátum	Irányítószám	Település	
+	123\456\789	Krankovits Meli	<input checked="" type="checkbox"/>	1975.05.29.	9178	Hédervár	
+	123\456\111	Lusta Béla	<input type="checkbox"/>	1955.05.12.	9023	Győr	
+	123\111\222	Kiss Bea	<input checked="" type="checkbox"/>	1950.12.12.	9023	Győr	
+	111\222\333	Kiss Anita	<input checked="" type="checkbox"/>	2002.10.14.	9023	Győr	
+	111\333\444	Lusta Béláné	<input checked="" type="checkbox"/>	1956.05.18.	9002	Móvár	
▶ -	111\222\444	Nagy Miska	<input type="checkbox"/>	1948.02.03.	8700	Sehonnán	
	Sorszám	Napló	Dátum	Idő	Vizsg. típusa	Térítési kat	
ℓ	1	00001	2007.07.12.	10:42	1 02		
*	(Számláló)	00000	2007.07.12.	10:42	0		
*			<input checked="" type="checkbox"/>				

4. ábra

A kapcsolódó rekordot, mivel ez gyerek rekord a **Vizsgtípus** táblához viszonyítva, csak akkor lehet elmenteni, ha ott már szerepel szülő rekord.

Magyarul, amíg nincs a **Vizsgtípus** táblában szülő rekord, addig itt nem szerepelhet érték.

Ugyanez igaz az **Orvosok** táblára a **PECSETSZAM** mező értékeire.

Tehát ez a fejta adatbevitel ekkor hasznos, ha az alaptáblákban már vannak rekordok.

## FELADAT

Töltse fel a többi táblát is alapértékekkel, 1-5 értelmes rekorddal.

A lekérdezésekhez hasznos lesz.

## Ellenőrző kérdések - Adatbevitel a táblába

Oldja meg a feladatokat!



1. Milyen formátumú dátumot kell megadnia, ha rövid dátum van beállítva?

- 2007. június 19.
- 2007.06.19.
- 2007. jun. 19.



2. Az igen/nem típusú adatnál milyen lesz az alapértelmezett megjelenítés?

- beviteli mező
- lista
- jelölőnégyzet

## Modulzáró feladatsor

Oldja meg a feladatokat!



### 1. Jelölje meg az állításokat, igaz vagy hamis? (1 pont)

A sablonokat akkor alkalmazzuk, ha tanulmányozni akarunk egy adatbázist.

igaz hamis

Az üres adatbázis létrehozásához meg kell adni a nevét és a helyét, hogy el tudjuk menteni.

igaz hamis

Az Access nem enged meg több szóból álló neveket az adatbázishoz.

igaz hamis



### 2. Az alábbiak közül jelölje meg melyek az Access adatbázis objektumai! (1 pont)

Jogosultságok

Táblák

Makrók

Nyomtatványok



### 3. Az alábbiak közül melyik karakter eredményezi a beviteli maszk készítésekor, hogy számjegy vagy szóköz következzen, nem kötelező bejegyzésként? (1 pont)

L

A

&

#

**4. Jelölje be a következő állításokat, igaz vagy hamis? (1 pont)**

- A többmezős indexelt tulajdonság csak menüből állítható be. igaz hamis
- A számláló típusú mező nem lehet kötelező kitöltésű. igaz hamis
- A hiperhivatkozás típusú mező nem indexelhető. igaz hamis

**5. Az alábbi érvényességi szabályok közül szintaktikailag melyik helyes? (1 pont)**


- Between 10 or 100
- Between 10 and 100
- =>10

**6. Milyen kategóriákkal találkozhatunk, ha megnyitjuk a kifejezésszerkesztőt? (2 pont)**

- logikai kifejezések
- kész sablonok
- dátum függvények
- beépített kifejezések

**7. Állapítsa meg milyen kapcsolat hozható létre, ha mindkét mező elsődleges kulcs, illetve egyedi index. (1 pont)**

- 1:1
- 1:N
- N:N
- határozatlan

 **8. Állapítsa meg igaz vagy hamis az állítás! (1 pont)**

A kapcsolódó mezőknek kivétel nélkül ugyanaz az adattípusuk.	igaz	hamis
Minden táblában van elsődleges kulcs.	igaz	hamis
Hosszú egész típusú mező kapcsolatba hozható számláló típusú mezővel.	igaz	hamis

 **9. Állapítsa meg a tagmondatok közötti összefüggést! (2 pont)**


Ha a hivatkozási integritás be van kapcsolva, és a kaszkádolt frissítés nem engedélyezett, akkor az elsődleges táblában lehet rekordot módosítani, a kapcsolódó táblában viszont nem módosul.

A B C D E

 **10. Az alábbi lekérdező utasítás végrehajtásakor mi lesz az eredmény?**

```
SELECT BNEV FROM Betegek  
WHERE Betegek.BNEV LIKE "A%"
```

- A betegek tábla összes rekordját listázza.
- A betegek táblából a betegek neveit listázza.
- A betegek táblából minden olyan beteget listáz akinek a neve A-betűvel kezdődik.
- A betegek táblából minden olyan beteg nevét listázza akinek a neve A-betűvel kezdődik.

 **11. A lekérdezés melyik záradékában lehet megadni a rendezettséget? (1 pont)**

záradék: \_\_\_\_\_





12. Az alábbiak közül milyen lekérdezés típusokat ismer? (2 pont)

- kijelölő
- frissítő
- táblamódosító
- törlő



13. Melyik nézetben módosítható a lekérdezés? (1 pont)

- SQL nézet
- adatlap nézet
- tervező nézet



14. Milyen kifejezést kell beírni a kipontozott helyre, ha nem szeretnénk felsorolni a mezőneveket, és az összesre kíváncsiak vagyunk?

**SELECT ..... FROM Betegek;**

**(A választ csupa nagybetűvel adja meg!) (2 pont)**

kifejezés: \_\_\_\_\_

## Űrlapok elemei, létrehozása.

### Tanulási célok

A lecke tanulása után Ön képes lesz:

- új autoűrlapot létrehozni,
- adott űrlapelemet elhelyezni az űrlapon,
- űrlapvezérlő elemeinek tulajdonságait felsorolni.

### Tevékenység:

Olvassa el a jegyzet 117-132. oldalán lévő tananyagot!

Oldaj meg a fejezetben található példákat is!

### Összefoglalás, kiegészítés a tananyaghoz:

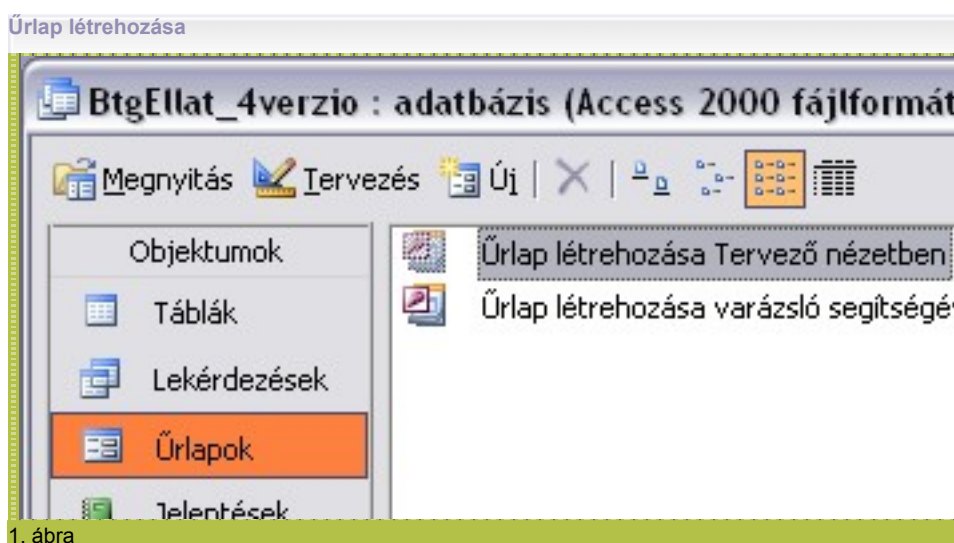
Az **űrlap** elsősorban a táblák karbantartási funkcióinak elvégzésére szolgál. Adatot felvinni, törölni természetesen adatlap nézetben is lehet, de a felhasználó számára szakszerűen az űrlapok biztosítják az adatokhoz való hozzáférést.

Az űrlapon megjelenő információk jelentős része egy **rekordforrás** (Tábla/Lekérdezés, azaz SELECT SQL művelet), más részük **vezérlőelem**.

Az űrlap készítésének módjai közül gyakran használjuk az **AutoŰrlap** funkciót, mivel ez jól használható, az adatbevitelnél megszokott űrlapot generál.

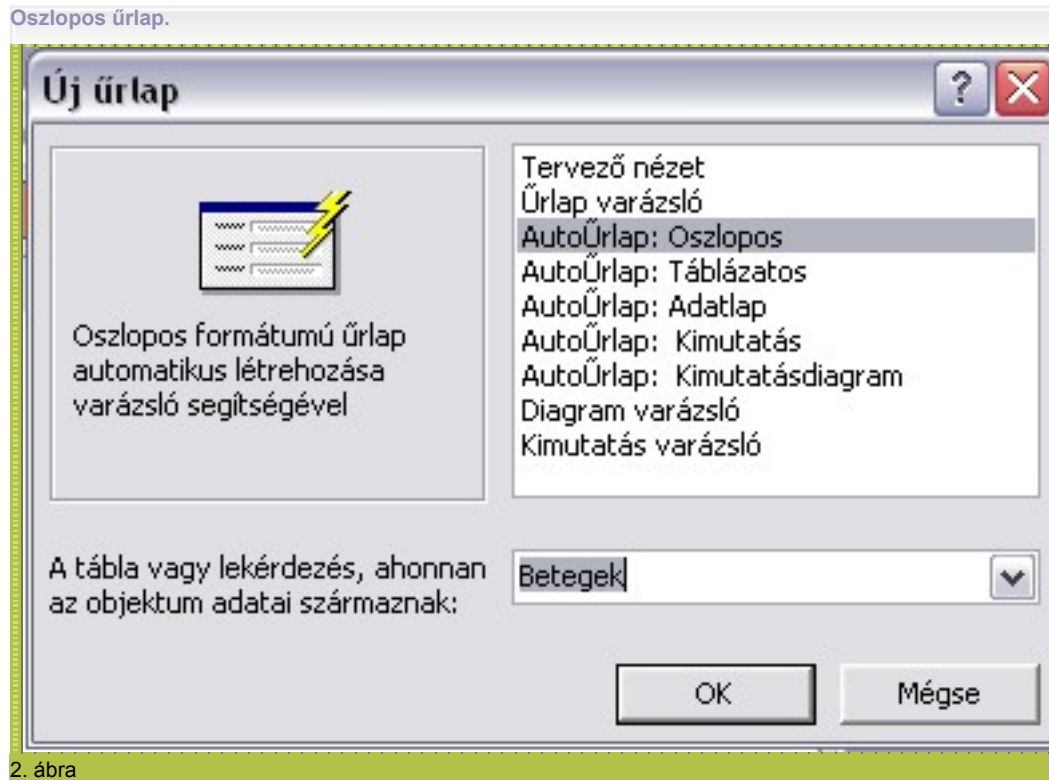
### AutoŰrlap létrehozása

Az objektum típusok közül először ki kell választanunk az Űrlap elemet, majd a Tervező nézetben szeretnénk létrehozni, így az Új gombra kell kattintani. Ezt mutatja az 1. ábra.



1. ábra

A következő lépésben egy AutoŰrlapot szeretnénk létrehozni a Betegek tábla karbantartására. Itt az Űrlap típusát választjuk ki, és a tábla (vagy lekérdezés) nevét. Az AutoŰrlap lehet oszlopos elrendezésű, táblázatos, vagy adatlap. A 2. ábra. szemlélteti a kiválasztott lépést.

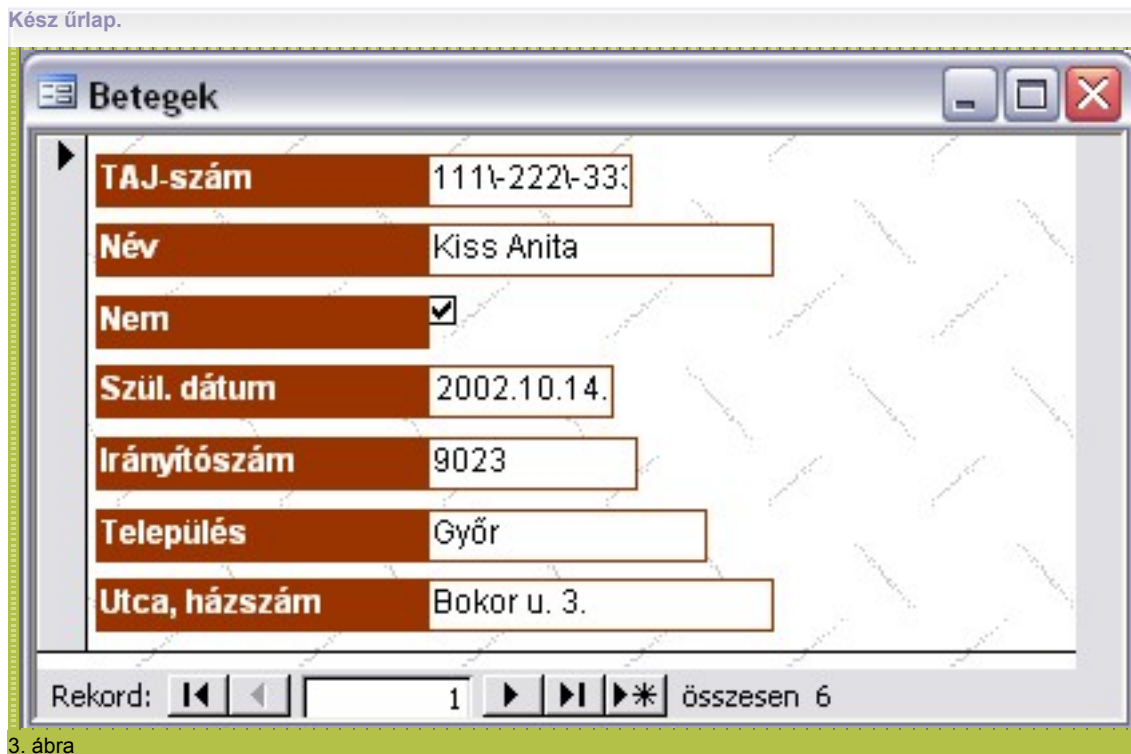


2. ábra

Az OK gombra kattintva máris elkészül az űrlapunk, űrlap1 néven amit a szokott módon a bezáráskor nevezhetünk és menthetünk el.

Az űrlapunkon az mezőneveknél megadott megjelenítési formának megfelelően generálódtak a beviteli mezők, így a beteg neménél jelölőnégyzet van, az összes többi helyen beviteli mező.

Az adatbeviteli korlátozások, maszkok itt is érvényesek.



A kész űrlap alján látható a **rekord navigációs sáv**, ami azonos azzal, amit az adatbevitelnél adatlap nézetben már tárgyaltunk.

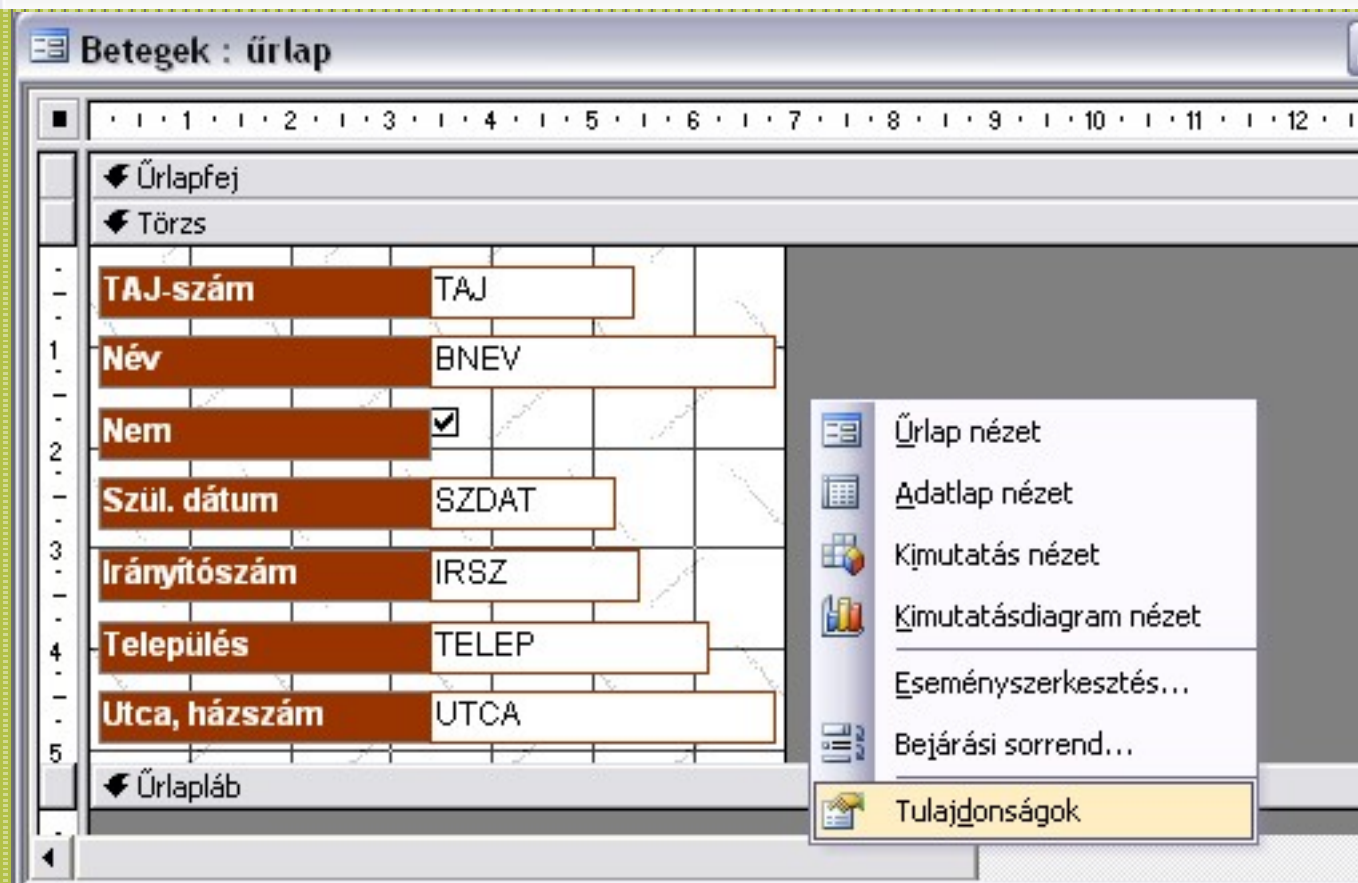
Az űrlap tulajdonságainál még számtalan dolgot lehet beállítani, ezeket az űrlap tervező nézetében (a Betegek nevű űrlap aktív, ekkor kattintunk a Tervezés gombra, csakúgy mint a lekérdezésnél már használta!) lehet megtenni.

A tervező nézetben az űrlap 3 része jól látható: **Űrlapfej, Törzs, Űrlapláb**. A lábrészbe kerülő rekord navigáció itt nem látszik. Ezt mutatja a 4. ábra.

A Törzs részbe kerültek a címkék, beviteli mezők, jelölőnégyzet.

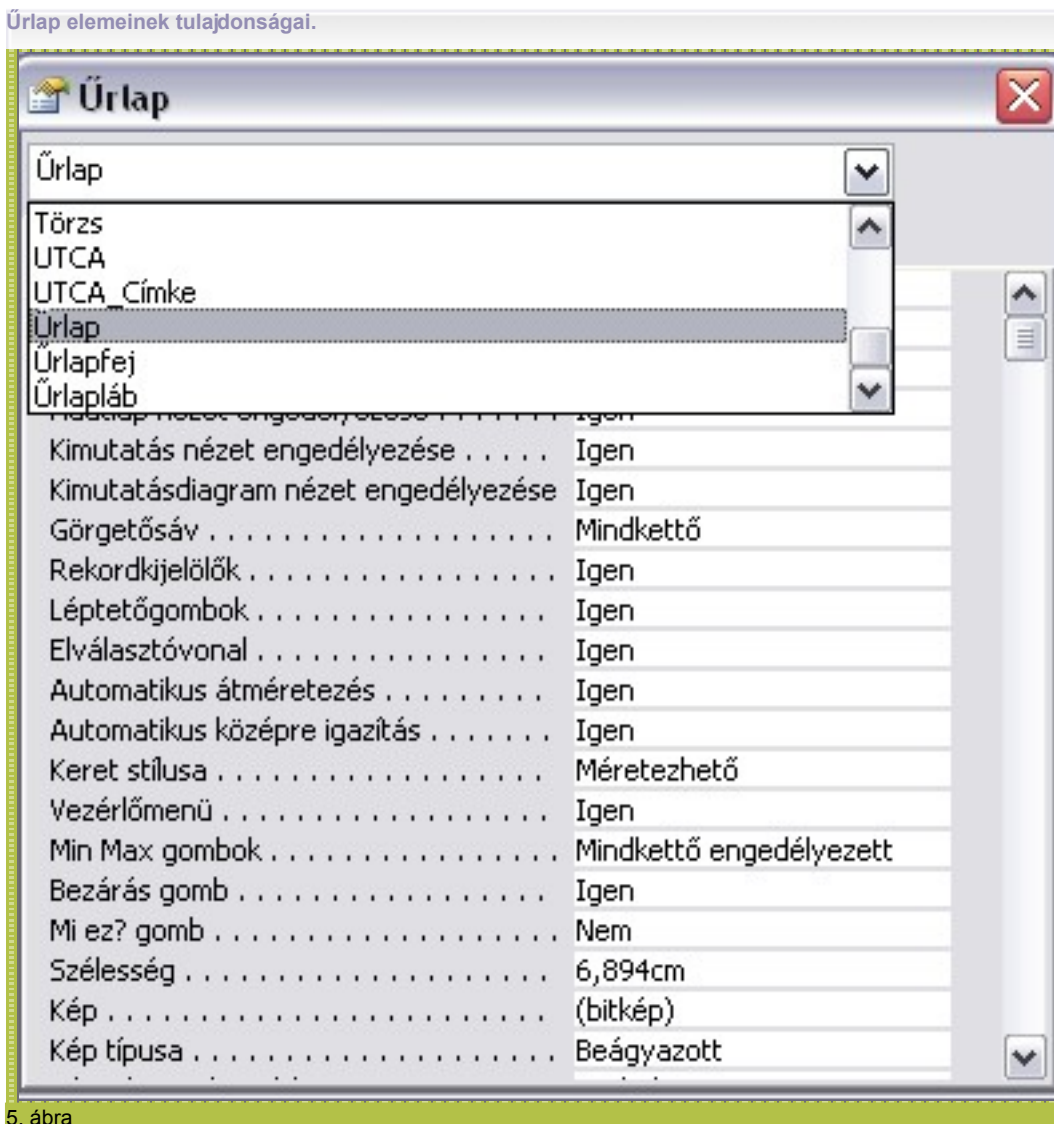
Ezek méretét, helyét az egér segítségével módosíthatja.

Űrlap módosítása.



4. ábra

A tulajdonságok gombra kattintva jeleníthetjük meg az űrlap részeinek beállításait úgy, hogy előtte ki kell választani az elemet aminek a tulajdonságait lekérjük. Így a lenyíló listából (vagy az elemre kattintva a tervező ablakban) lehet megtudni, hogy az elem, a hozzá tartozó címke, az űrlap, a fejléc, vagy a lábrész tulajdonságait nézzük.



5. ábra

Nézzük át röviden mit érdemes egy AutoŰrlapon változtatni.

Ha a Törzs részt nézzük, akkor a címkék feliratával és a mezőkkel nem sok dolgunk van.

Érdemes az ablak gombjait, a görgetősávokat megfigyelni, engedélyezni, valamint a színeket átállítani.

Az **űrlap** elem **formátum** fülén lévő lényeges adatokat és beállításait itt felsorolom:

Görgetősáv: egyik sem/csak vízszintes/csak függőleges/mindkettő

Rekorkijelölő: ezek a navigációs sáv gombjai, igen/nem

Léptetőgomb: ezek a navigációs sáv gombjai, igen/nem

Min/Max gomb: ablak jobb felső sarkának gombjai

Bezárás gomb: ablak bezárása gomb

Kép: háttérképként magadott kép

**Adat** fülön lehet engedélyezni:


szerkesztés: igen/nem

törlés: igen/nem

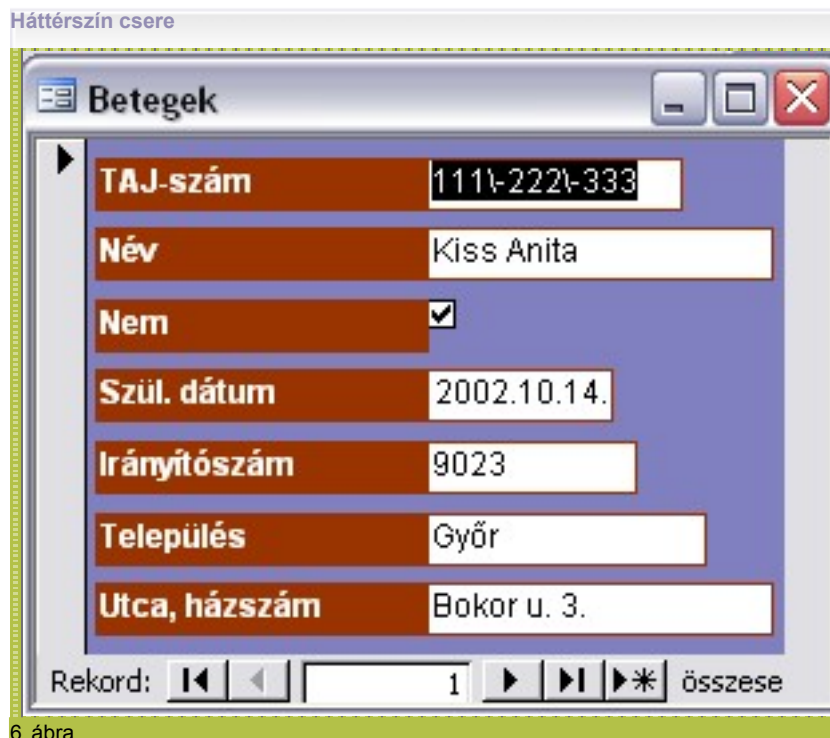
adatbevitel: igen/nem

rekordzárolás: nincs/összes/*szerkesztett rekord*.

A rekordok zárolására több felhasználós hozzáférés esetén van szükség, amíg valaki szerkeszti a rekordot, addig más ne módosíthassa, vagy törölhesse.

A **Törzs** elem **formátum** fülén lehet a színeket módosítani. Ha egyszínű hátteret szeretnénk az űrlapunknak, először az Űrlaphoz tartozó Kép objektumot ki kell törölni, majd itt a Hátterszín tulajdonságot lehet a mellette lévő gombra  kattintva beállítani.

A következő ábra már a megváltozott egyéni űrlapunkat mutatja.



6. ábra

**Feladat:**

Végezze el a jegyzetben lévő feladatokat!

## Ellenőrző kérdések - Űrlapok elemei, létrehozása.

Válaszoljon a kérdésekre alábbi ábra alapján!



### 1. Jelölje meg az alábbi állítások közül mi igaz az Autoűrlapra vonatkozóan!

- Autoűrlapot csak táblákhoz lehet létrehozni.
- Autoűrlap készülhet meglévő lekérdezéshez is.
- Az űrlap beviteli mezői a Törzs részbe kerülnek.
- A fejrészbe automatikusan bekerül a rekord navigációs sáv.



### 2. Melyek azok az adatkarbantartási funkciók, amiket egy űrlap egyszerű felhasználónak biztosíthat?

- Tábla törlése
- Rekord módosítása.
- Rekord felvitele a táblába.
- Kapcsolatok szerkesztése.



### 3. Mi történik, ha egy adatbázist 2 felhasználó is használhat egyidőben egy űrlapon keresztül, és a rekorzárolás nincs engedélyezve?

- Nincs probléma, aki először végez, az marad érvényben.
- Mindkét módosítást elmenti a rendszer csak más időbejegyzéssel.
- Az egyik felhasználó kitörölheti a másik által módosítani kívánt rekordot.
- Ugyanazt a rekordot nem tudják lekérdezni egyszerre.





## A parancsgomb varázsló használata.

### Tanulási célok

A lecke tanulása után Ön képes lesz:

- a parancsgomb űrlapvezérlő elem használatával űrlapot megnyitni, bezárni, nyomtatni.

### Tevékenység:

Olvassa el a következő kiegészítő tananyagot!

### Kiegészítés a tananyaghoz:

Az adatbázisunkhoz az előző lecke alapján tudunk felhasználói felületeket, űrlapokat készíteni különböző karbantartási funkciókhoz.

Elengedhetetlen azonban, hogy olyan felhasználó is tudja a végső programot kezelni, aki az Access-hez nem feltétlenül ért.

A programunknak tehát kell olyan induló felülettel rendelkeznie (menü), ahonnan a fontos funkciókat, űrlapokat, jelentéseket meg lehet nyitni, bezárni, nyomtatni.

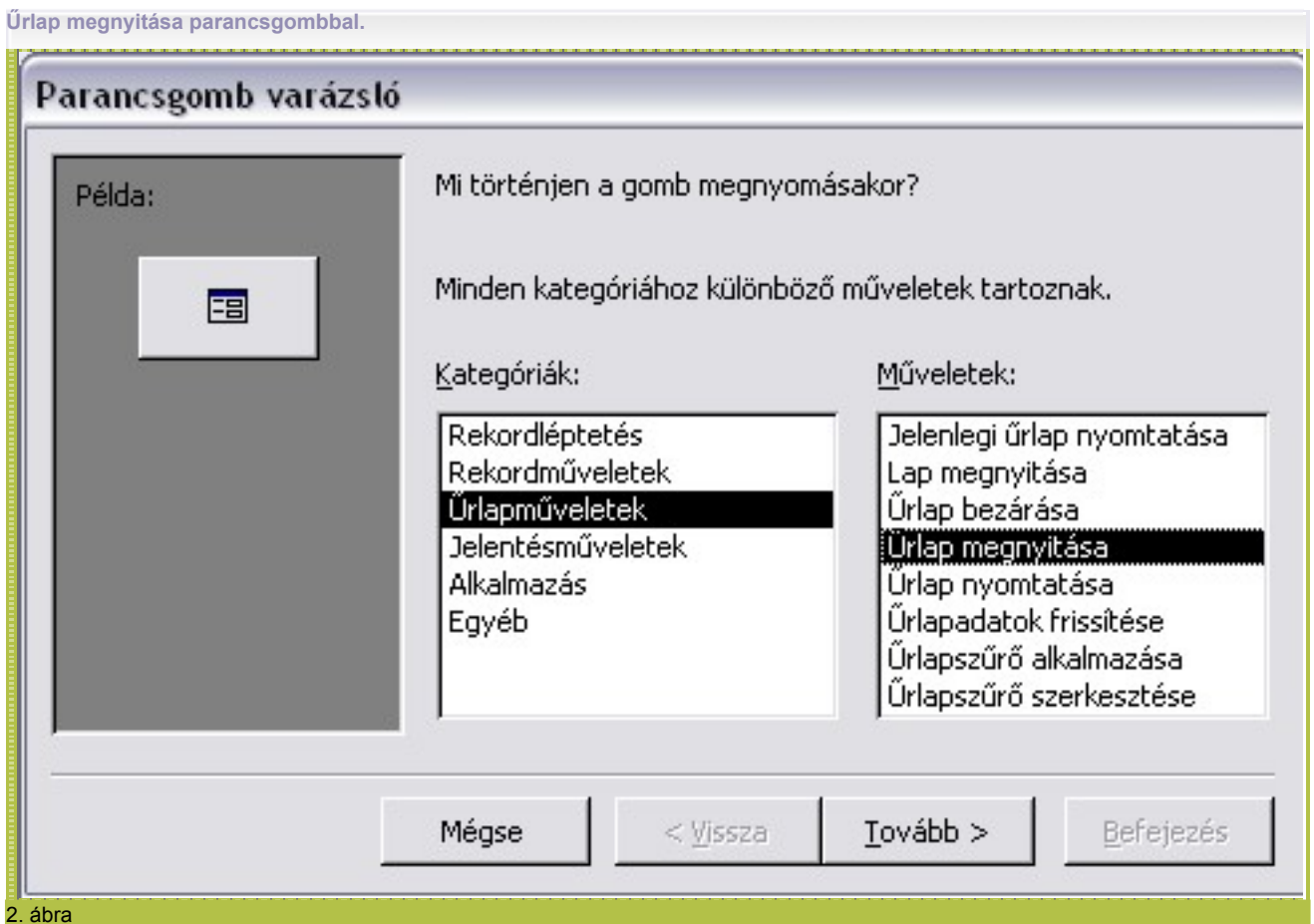
Ezekhez a műveletekhez az Access ad támogatást, programozás nélkül is elvégezhetőek egy űrlap vezérlőelem, a **parancsgomb** segítségével.

Ezt fogjuk most szemléltetni egy olyan példával, ahol az előző lecke **Betegek** űrlapját megnyitjuk egy újonnan létrehozott **Betegmegnyit** űrlap segítségével.

### Feladatmegoldás lépései:

1. Az Űrlapok objektumtípust válassza ki, majd az Új nyomógommbal hozzon létre egy üres űrlapot tervező nézetben.
2. Az űrlap törzs részébe helyezzen el egy képet ami a témához kapcsolódik. (Keressen az interneten!)
3. Az eszköztáron (ha nincs bekapcsolva, akkor a **Nézet menü/Eszköztárak** útvonalon megtalálja) keresse meg a parancsgomb ikonját.





2. ábra

5. A **Tovább** gombra kattintva ki kell választani a listából melyik űrlapot nyissa meg a gomb (Betegek), felirat tartozzon hozzá vagy kép (itt válassza a képet), végül a parancsgomb hivatkozási nevét is.
6. Ha a gomb elkészült, akkor az űrlapot is el kell menteni, legyen a neve **betegmegnyit**.
7. Futtsassa az űrlapot!

#### FELADAT

Az elészült űrlapon a tanult tulajdonságok beállításával (előző lecke!) módosítsa az űrlap kinézetét, tegyen rá zöld hátteret, feliratot, majd szedje le a görgetősávot, és a rekord navigációs sávot!

Az elkészült űrlap a következő lesz:



A következő videó bemutatja a kész feladatot, ha nehezen érthető, mielőtt nekilát, nézze végig!  
A feladatot a **btgellat\_4verzio.mdb** már tartalmazza, letölthető!

---

Az online tananyag ezen pontján interaktív, vagy multimédia objektumot talál. Kérem nézze meg az online tananyagot!

## Ellenőrző kérdések - Parancsgomb.

Válaszoljon a kérdésekre!



**1. Az alábbiak közül milyen műveletekhez lehet parancsgombot létrehozni varázslóval?**

- Tábla létrehozás.
- Űrlap nyomtatás.
- Jelentés megnyitás.
- Tábla szerkezet módosítás.



**2. Az alábbiak közül melyek igazak a parancsgombokra?**

- Csak VBA programozással lehet létrehozni.

- Különböző eseményhez kötött a végrehajtása.
- Csak egy lehet belőle egy űrlapon.
- Lehet rajra felirat, vagy kép is.

## Jelentések készítése, elemei.

### Tanulási célok

A lecke tanulása után Ön képes lesz:

- Jelentést készíteni varázsló segítségével.

### Tevékenység:

Olvassa el a jegyzet 133-140. oldalát!

Oldja meg a jegyzetben található feladatokat!

### Összefoglalás, kiegészítés a tananyaghoz.

A jelentések készülhetnek megjelenítésre, vagy nyomtatási szándékkal is.

#### Részei:

- törzsszakasz
- jelentésfej, jelentésláb: csak egyszer jelenik meg
- oldalfej, oldalláb: minden oldalon
- csoportfej, csoportláb: ha volt csoportosítás a jelentésben

A jegyzetben található feladat végigvezet egy varázslóval elkészített jelentésen, aminek a kezelése nagyon hasonlít az űrlap elkészítéséhez.

Amit viszont érdemes kiemelni, hogy ezek mögött a műveletek mögött is SQL utasítások állnak, így emlékezzünk vissza, hogy egy rendezés elkészítése a varázslóval **ORDER BY** záradék használatát jelenti a **SELECT** utasításban.

Megjelenik még a **GROUP BY** záradék a csoportképzésnél, valamint a **HAVING** záradék a csoportok szűrésénél. Ezen kívül több táblából való jelentést úgy készítünk, hogy a táblákat a **JOIN** műveletek valamelyikével kötjük össze!

Hivatalos kinézetű, név szerinti rendezett jelentés a betegekről.

Betegek						
Név	TA.J-szám	Nem	Szül. dátum	Irányítószám	Település	Utca, házszám
Kis Anikó	111\222\333	<input checked="" type="checkbox"/>	2002.10.14.	9023	Győr	Bolero 3.
Kis Bea	123\111\222	<input checked="" type="checkbox"/>	1950.12.12.	9023	Győr	Bolero 3.
Kovács Mária	123\454\789	<input checked="" type="checkbox"/>	1975.05.29.	9178	Budapest	Mező u. 1.
Lukács Béla	123\454\111	<input type="checkbox"/>	1955.05.12.	9023	Győr	Új u. 2.
Lukács Beáta	111\333\444	<input checked="" type="checkbox"/>	1954.05.18.	9002	Ménfőcsanak	Fűfő u. 8.
Nagy Mária	111\222\444	<input type="checkbox"/>	1948.02.03.	8700	Székesfehérvár	Új u. 4.

1. ábra


## FELADAT

A parancsgomb használata c. leckeiben tanultak alapján hozzon létre olyan űrlapot, ahonnan ki lehet nyomtatni ezt a jelentést!

A jegyzet jelentéseit a **btgellat\_4verzio.mdb** fájlban megtalálja a letölthető dokumentumok között!

## Ellenőrző kérdések - Jelentések

Válaszoljon a kérdésekre!

 **1. Válassza ki azt az állítást amely igaz a jelentésekre!**

- A jelentés csak egy tábla adatait tartalmazza, rendezett sorrendben.
- A jelentés adatait lehet csoportosítani, de ekkor a csoporton belül nem lehet rendezni.
- Az oldalfej és oldalláb minden oldalon látszik.
- Ha volt csoportosítás a jelentésben, akkor az mindig külön oldalra kerül.



**2. Folytassa a mondatot! A csoportosítás használatát a jelentésben...**

- Az ORDER BY záradékot jelenti a SELECT utasításban.
- Az WHERE záradékot jelenti a SELECT utasításban.
- Az GROUP BY záradékot jelenti a SELECT utasításban.
- A JOIN záradékot jelenti a SELECT utasításban.

## Modulzáró feladatsor.

Válaszoljon a kérdésekre!



**1. Az alábbi állítások közül melyek igazak a jelentésekre? (1 pont)**

- A jelentésfej és a jelentésláb csak egyszer jelenik meg a jelentésben.
- A jelentés adatainak csoportosítása nem zárja ki a rendezettséget.
- A törzsszakasz nem tartalmazhatja lekérdezés eredményét.



**2. Ha több táblából szeretnénk jelentést készíteni, az alábbiak közül melyik záradékot alkalmazza az Access a lekérdezésben? (1 pont)**

- GROUP BY
- ORDER BY
- JOIN
- WHERE



**3. Az alábbiak közül milyen műveletekhez lehet parancsgombot létrehozni varázslóval? (1 pont)**

- Tábla törlése.
- Űrlap megnyitása.
- Alkalmazás futtatása.
- Tábla szerkezet módosítás.



**4. Az alábbiak közül melyik állítás igaz általánosságban a vezérlőelemekre? (1 pont)**

- A kötetlen vezérlőelemnek nincs adatforrása.
- A számított vezérlőelem adatforrása egy összeadás eredménye.
- A kötött vezérlőelem egy rekordhoz van kötve.

 **5. Mi az űrlapok elsődleges funkciója? (1 pont)**

- Táblaadatok karbantartása.
- Csak az adatfelvitel megkönnyítése.
- Átláthatóbbá tenni az adatbázis lekérdezéseit.

 **6. Jelölje meg az alábbi állítások közül mi igaz az Autoűrlapra vonatkozóan! (2 pont)**

- Autoűrlapot csak táblákhoz lehet létrehozni.
- Autoűrlapra csatolt címkék kerülnek.
- Az Autoűrlap beviteli mezői mindig kötöttek.
- Az Autoűrlap később nem módosítható tervező nézetben sem.

 **7. Folytassa a mondatot! A segédűrlap..... (1 pont)**

- a Help-ben lévő magyarázó űrlap.
- segítség az űrlap kitöltéséhez.
- az űrlapon lévő másik űrlap.
- az űrlapba beszúrt jelentést tartalmazza.

