

# Érettségire felkészítő feladatgyűjtemény – Informatika

## Az Adatbázis-kezelés fejezet feladatainak megoldása

A megoldások bemutatásánál feltételeztük az MS Access kezelésének alapszintű ismeretét. Az eszköztár ikonjai helyett a megfelelő menüparancsra utaltunk. Emlékeztetünk arra, hogy a gyorsmenüt a jobb egérgombbal történő kattintással hívhatjuk elő. A gyorsmenüben találjuk az egyes objektumok tulajdonságlapját megjelenítő parancsot (Tulajdonságok).

Az adatbázis-kezelés alapműveleteit (adattábla importálása, összesítő lekérdezés, jelentés létrehozása stb.) a legelső előfordulásnál részleteztük. Az összesítő lekérdezéseknél például a későbbiekben már csak zárójelben adtuk meg a mezőnév után a záradékot vagy az aggregáló függvényt. Azokat a műveleteket magyaráztuk el részletesebben, amelyek nem szerepelnek a tankönyvben.

Az ugyanolyan típusú feladatoknál esetenként más-más megoldás mutattunk be (például maximális érték kiválasztása a Csúcsérték tulajdonság beállításával vagy összesítő lekérdezés létrehozásával). A bonyolult feltételek, összetett függvények alkalmazása helyett előnyben részesítettük az egymásra épülő lekérdezéseket.

Az objektumok elnevezésében követtük a feladatgyűjtemény jelölésmódját (feladat sorszáma, aláhúzásjel, a tartalomra utaló rövidítés). Ha nem volt megadva, akkor a rövidítés helyett szemléletes kifejezést használtunk. Inkább a kifejező, mint a rövid táblanevekhez ragaszkodtunk. Általában a megoldás nevét beírtuk a feladat sorszámahoz, egyébként pedig félkövér betűkkel szedve beillesztettük a szövegbe. A segédlekérdezéseket két aláhúzásjellel jelöltük a feladat száma után (például 4\_\_xx).

A feladatgyűjtemény CD-jén lévő fájlok táblázatait általában néhány rekorddal kiegészítettük, hogy jobban illeszkedjenek a lekérdezésekhez.

Az MS Access néha átrendezi az elkészült tervezőrácsot, módosítja a kifejezéseket. Ezekben az esetekben a lekérdezések megnyitásakor mást látunk, mint ami az alábbi leírásokban olvasható. A módosítás természetesen nem érinti a megoldás helyességét.

A lekérdezések szűrőfeltételeiben szereplő mezőket az ellenőrzés céljából általában akkor is megjelenítettük, ha a feladat ezt nem kérte. A tábláknál, lekérdezéseknél, jelentéseknél a feladatgyűjteményben megszabottn túl csak elemi formázást alkalmaztunk.

Végül megjegyezzük, hogy egymáshoz hasonló lekérdezések készítésénél célszerű az elsőről másolatot létrehozni (*Fájl/Mentés másként*), és azt módosítani.

### Csoportcím

#### 1. feladat: Csoportcím tábla

Létrehozunk egy új adatbázist *Csoportcím* néven. A CD-ről importáljuk a *csopcim.txt* adatait (*Fájl/Külső adatok átvétele/Importálás*). A *Fájl típust* kiválasztjuk a *Szövegfájlokat*. Az importálásnál alkalmazott beállítások:

- A mezőket karakterek határolják. A *Speciális* gombbal megjelenített ablakban a 4. mező adattípusát is *Szövegre* állítjuk.
- A mezőket *Tabulátor* választja el egymástól. Az első sor nem tartalmazza a mezőneveket.
- Az adatokat egy új táblában szeretnénk tárolni.
- Mezőnevek: *Név, Neme, Város, IrSzám, Utca, Telefon*.
- Ne legyen elsődleges kulcs.
- Táblanév: *Csoportcím*

A táblát a következő adatokkal egészítjük ki:

Kana Péter	fiú	Sopron	9400	Győri út 504.	(99)365-989
Szussza Nóra	lány	Bp.	1143	Soproni u. 43.	(1)156-4865

Az új rekord adatait a tábla végén lévő üres sorba tudjuk beírni.

## 2. feladat: 2\_rend

Új jelentést készítünk (*Jelentések* objektumcsoport, *Jelentés létrehozása* varázsló segítségével). A varázsló egyes lépéseinél alkalmazott beállítások:

- Felvesszük a *Csoportcím* tábla összes mezőjét.
- Nem adunk hozzá csoportszintet.
- Kiválasztjuk a *Név* mezőt a *Növekvő* rendezéshez.
- Táblázatos szerkezetet jelölünk ki.
- Tetszőleges stílust választunk.
- Jelentéscím a mentésnél: *2\_rend*

A jelentés *Tervező* nézetében a *Jelentésfej* feliratát *Csoportcímre*, az *Oldalfej IrSzám* feliratát pedig *Irányítószámra* változtatjuk.

## 3. feladat: 3\_Létszám

*Tervező* nézetben létrehozunk egy új lekérdezést. Hozzáadjuk a *Csoportcím* táblát. A lekérdezést összesítővé alakítjuk (*Nézet/Összesítés*). Felvesszük a tervezőrácsba a *Név* mezőt, majd az *Összesítés* rovatban kiválasztjuk a *Count* függvényt. A mező *Cím* tulajdonságához (*Nézet/Tulajdonságok*) beírjuk „A csoport létszáma” szöveget. A mező szélességét megnöveljük, hogy látható legyen a teljes cím.

## 4. feladat

Létrehozunk egy új lekérdezést (**4\_FiúkSzáma**), és hozzáadjuk a *Csoportcím* táblát. A lekérdezést összesítővé alakítjuk. A tervezőrácsba felvesszük a *Neme* mezőt, és a *Count* függvényt választjuk az *Összesítés* sorban. Felvesszük ismét a *Neme* mezőt a *Where* záradékkal az *Összesítés* sorában, feltételnek pedig beírjuk a „fiú” szót. Az oszlop *Címe*: „A fiúk száma”.

A lányok létszámának megállapításához pontosan ugyanilyen lekérdezést készítünk (**4\_LányokSzáma**), csak a *Neme* mező feltételeként a „lány” szót írjuk be.

A feladatot megoldhatjuk egyetlen lekérdezéssel is (**4\_FiúLány**). Készítsünk összesítő lekérdezést, amelybe felvesszük a *Csoportcím* tábla *Neme* mezőjét, és a *Group By* parancsot választjuk az *Összesítés* sorában. Felvesszük a *Név* mezőt a tervezőrácsba a *Count* függvénnnyel az *Összesítés* sorában. Ez utóbbi oszlop *Címe*: „Létszám”.

**Megjegyzés:** a fiúk számát például úgy is meghatározhatjuk, hogy egy új lekérdezés tervezőrácsának *Mező* sorába beírjuk a

```
DCount (" [Név] " ; "Csoportcím" ; " [Neme]='fiú' " )
```

kifejezést. A lekérdezéshez ne adjuk hozzá a *Csoportcím* táblát!

## 5. feladat: 5\_FiúSzázalék

Készítünk egy új lekérdezést, melyhez táblaként hozzáadjuk a *3\_Létszám* és a *4\_FiúkSzáma* lekérdezést. A tervezőrács első oszlopának *Mező* sorába beírjuk a

```
[ 4_FiúkSzáma ] ! [ CountOfNév ] / [ 3_Létszám ] ! [ CountOfNév ]
```

kifejezést (a szögletes zárójelbe tett tábla- és mezőnevet felkiáltójel választja el egymástól). A *Kifejezésszerkesztő* megkönnyíti a hosszú kifejezés összeállítását.

Oszlopcímnek megadjuk „A fiúk százalékos aránya” szöveget, az oszlop formátumát pedig *Százalékra* állítjuk. Az oszlopszélességet megnöveljük a cím hosszának megfelelő mértékben.

## 6. feladat: 6\_BejárókSzáma

Az új lekérdezéshez hozzáadjuk a *Csoportcím* táblát, majd a tervezőrácsba felvesszük a *Név* és a *Város* mezőt. A lekérdezést összesítővé alakítjuk. A *Név* mezőre a *Count* függvényt, a *Város* mezőre pedig a *Where* záradékot választjuk az *Összesítés* sorában. A *Where* feltételeként a *<>* „Bp.” kifejezést írjuk be. Az első oszlop címe: „A bejáró tanulók száma”, szélességét az *Adatlap* nézetben a cím hosszához igazítjuk.

## 7. feladat: 7\_TelefonokSzáma

Összesítő lekérdezést készítünk, melyhez hozzáadjuk a *Csoportcím* táblát, a tervezőrácsba pedig felvesszük a *Név* és a *Telefon* mezőt. Az *Összesítés* sorában kiválasztjuk a *Count* függvényt, illetve a *Where* záradékot. A *Where* feltétele az `Is Not Null` kifejezés. Az első oszlop címe: „A telefonok száma”. Ennek megfelelően állítjuk be az oszlopszélességet.

## 8. feladat: 8\_TelefonSzázalék

Az új lekérdezéshez hozzáadjuk a 3. és a 7. feladat lekérdezését. A tervezőrácsban egyetlen oszlopot használunk fel, melynek *Mező* sorába a következő kifejezést írjuk (alkalmazzuk a *Kifejezésszerkesztőt*):

```
[7_TelefonokSzáma]! [CountOfNév] / [3_Létszám]! [CountOfNév]
```

A *Formátumot Százalékra* állítjuk. Az oszlop címe: „A telefonon elérhető csoporttagok aránya”. Ennek hosszához állítjuk az oszlop szélességét.

## 9. feladat: 9\_AdatokJelentés

Először készítünk egy segédlekérdezést (**9\_Adatok**), amelyhez hozzáadjuk a következő lekérdezéseket: *Létszám*, *FiúkSzáma*, *LányokSzáma*, *BejárókSzáma*, *TelefonokSzáma*. Mindegyik lekérdezés egyetlen mezőt tartalmaz, ezeket felvesszük a tervezőrácsba.

A jelentést *Tervező* nézetben készítjük el a segédlekérdezés kiválasztásával. A jelentéstörzsbe beviteli mezőket helyezünk el a következő kifejezésekkel:

```
= "A csoport létszáma: " & [3_Létszám.CountOfNév] & " fő."
```

```
= "A fiúk száma: " & [4_FiúkSzáma.CountOfNév] & " fő."
```

```
= "A lányok száma: " & [4_LányokSzáma.CountOfNév] & " fő."
```

```
= "A csoport " & Kerek([4_FiúkSzáma.CountOfNév] / [3_Létszám.CountOfNév] * 100; 2) & "%-a fiú."
```

```
= "A bejáró tanulók száma: " & [6_BejárókSzáma.CountOfNév] & " fő."
```

```
= "A csoport " & [7_TelefonokSzáma.CountOfNév] & " tagjának van telefonja."
```

```
= "A csoporttagok " & Kerek([7_TelefonokSzáma.CountOfNév] / [3_Létszám.CountOfNév] * 100; 1) & "%-a érhető el telefonon."
```

Megnöveljük a karakterek méretét, hogy szövegük olvasható legyen. Elrendezzük és átméretezzük a beviteli mezőket is.

**Megjegyzés:** az 5. és a 8. feladat lekérdezéseit nem vehetjük fel a segédlekérdezésbe, mert a jelentés futtatásakor a kifejezésekben szereplő mezőnevek (*CountOfNév*) azonossága miatt hibaüzenetet kapunk.

## 10. feladat

A jelentéshez készítünk egy segédlekérdezést (**10\_Telefonszámok**), melynek tervezőrácsába felvesszük a *Csoportcím* tábla *Név* és *Telefon* mezőjét. A *Név* oszlopban *Növekvő* rendezést állítunk be, a *Telefon* oszlopba pedig beírjuk az `Is Not Null` feltételt. A jelentést (**10\_Telefonszámok**) a varázsló segítségével hozzuk létre a lekérdezés alapján. Felvesszük mindkét mezőt, itt már nem szükséges rendezést előírni. Táblázatos megjelenítést és tetszőleges stílust választunk.

## Barátok

### 1. feladat: Barátok tábla

Létrehozunk egy új adatbázist, melybe importáljuk a *barat.txt* tábláját (lásd a *Csoportcím* 1. feladatát). A tábla első sora tartalmazza a mezőneveket. Az importálás során az „Ir sz” mező típusát is *Szövegre* állítjuk. A táblához nem adunk elsődleges kulcsot. Az importálás után *Tervező* nézetben a feladatgyűjtemény mintájának megfelelően módosítjuk a mezőneveket (pontot nem írhatunk a mezőnévbe).

Új rekordokat *Adatlap* nézetben vehetünk fel a táblába. Az adatokat az *Új rekord* sorába írjuk.

## 2. feladat: 2\_Barátok

A jelentést a varázsló segítségével készítjük el. Felvesszük a *Barátok* tábla összes mezőjét, nem adunk hozzá csoportszintet, de *Növekvő* rendezést írunk elő a *Név* mező szerint. *Táblázatos* megjelenítést választunk, tetszőleges stílussal. A jelentés *Tervező* nézetében módosítjuk az *IrSz* mező feliratát és a jelentés címét.

## 3. feladat: 3\_Telefonszámok

Készítünk egy új lekérdezést, melyhez hozzáadjuk a *Barátok* táblát. A tervezőrácsba felvesszük a *Név* és a *Telefon* mezőt. A *Telefon* mező feltétele: `Is Not Null`.

## 4. feladat: 4\_E-mailCímek

A lekérdezést a 3. feladathoz hasonlóan készítjük el, csak a *Telefon* mező helyett az *E-mail* mező kerül a tervezőrácsba. Szükség esetén *Adatlap* nézetben módosítsuk az *E-mail* oszlop szélességét.

## 5. feladat

A feladat megoldásánál először feltételezzük, hogy Budapesten lakunk. Készítünk egy új lekérdezést (**5\_NemBudapestiekNeve**), melyhez hozzáadjuk a *Barátok* táblát. A tervezőrácsba felvesszük a *Név* és a *Város* mezőt. A *Város* mező megjelenítését kikapcsoljuk, feltételként pedig beírjuk a `<>"Budapest"` kifejezést.

A nem budapesti lakosok megszámlálásához készítünk egy újabb lekérdezést (**5\_NemBudapestiekSzama**), melyet összesítővé alakítunk át. Hozzáadjuk az *5\_NemBudapestiekNeve* lekérdezést, a tervezőrácsba felvesszük a *Név* mezőt, az *Összesítés* sorában pedig kiválasztjuk a *Count* függvényt. Az oszlop címe: „A nem budapesti lakosok száma”, ennek megfelelően módosítjuk a szélességét.

Ha az irányítószám alapján akarjuk összeállítani a nem budapesti lakosok listáját, akkor a `Not Like "1*"` kifejezést írjuk be a tervezőrácsba felvett *IrSz* mező feltételeként. (Az irányítószám *Szöveg* típusú mező!).

A feladat látványosabb megoldása paraméteres lekérdezést alkalmaz (**5\_Paraméteres**). Egy új lekérdezéshez hozzáadjuk a *Barátok* táblát. A tervezőrácsba felvesszük a *Név* és a *Város* mezőt. A *Város* mező megjelenítését kikapcsoljuk, feltételként pedig beírjuk a

```
<>[Melyik városban laksz?]
```

kifejezést. A lekérdezés futtatásakor az adatbázis-kezelő megjelenít egy párbeszédablakot, melyben rákérdez a lakóhely városára. Az eredménytábla mindazon barátok névsorát fogja tartalmazni, akik nem a megadott városban laktak. Ha a paraméteres lekérdezés alapján elkészítjük a rekordokat megszámláló lekérdezést (**5\_ParaméteresSzámlál**), akkor ennek a futtatásánál is megjelenik a párbeszédablak.

## 6. feladat: 6\_BarátokSzamaVárosonként

Készítünk egy új, összesítő lekérdezést, melyhez hozzáadjuk a *Barátok* táblát. A tervezőrácsba felvesszük a *Város* és a *Név* mezőt. Az elsőnél a *Group By* parancsot, a másodiknál a *Count* függvényt választjuk az *Összesítés* sorában. A második oszlop címe: „A barátok száma”.

## Gyűlés

### 1. feladat: Gyűlés tábla

Létrehozunk egy új adatbázist, melybe importáljuk a *gyules.txt* fájl táblázatát (lásd a Csoportcím 1. feladatát). Az első sor tartalmazza a mezőneveket, a felajánlott adattípusokat nem módosítjuk. Nem adunk elsődleges kulcsot a táblához. A mezőneveket a tábla *Tervező* nézetében a feladatgyűjteménynek megfelelően módosítjuk.

A *Lyukas-e* mezőt úgy értelmezzük, hogy az 1-es jelöli a lyukasórát.

A táblát kiegészítettük a feladatgyűjteményben látható táblázat El Emíliára vonatkozó sorával.

### 2. feladat: 2\_raer

Összesítő lekérdezést készítünk, melyben a *Nap* és *Óra* mezőknél a *Group By* parancsot, a *Lyukas-e* mezőnél pedig a *Sum* függvényt jelöljük meg a tervezőrácsban. (A *Név* mezőt nem vesszük fel a

tervezőrácsba!) A *Sum* értékeit tartalmazó oszlop címe: „A ráérő tanárok száma”. Az oszlopszélességet ennek megfelelően módosítjuk.

### 3. feladat

Először összesítő lekérdezést készítünk (**3\_\_LegtöbbLyukasóra**), melyhez hozzáadjuk a *2\_raer* lekérdezést. A tervezőrácsba felvesszük a *SumOfLyukas-e* mezőt, és kiválasztjuk a *Max* függvényt. A feladat megoldását meghatározó lekérdezéshez (**3\_\_legtobb**) hozzáadjuk a *2\_raer* és az előzőleg elkészített lekérdezést. A két táblát a *SumOfLyukas-e*, illetve a *MaxSumOfLyukas-e* mezők segítségével összekapcsoljuk. A kapcsolat tulajdonságainál (gyorsmenü) szoros illesztést állítunk be (ez az alapértelmezés). Ezek után nincs más dolgunk, mint a tervezőrácsba felvenni a *2\_raer* lekérdezés *Nap* és *Óra* mezőjét.

Ha nem készítünk jelentést vagy űrlapot a lekérdezéshez, akkor egy újabb oszlop felvételével jeleníthetünk meg magyarázatot az *Adatlap* nézetben. A tervezőrács *Mező* sorába beírjuk a „A legtöbb ráérő tanár:” karaktersorozatot, az oszlop *Címe* pedig: „Összesítés”.

**Megjegyzés:** a lekérdezések összekapcsolása helyett megtehettük volna, hogy a tervezőrácsba hozzáadjuk a *SumOfLyukas-e* mezőt a

```
[ 3__LegtöbbLyukasóra ] ! [ MaxOfSumOfLyukas-e ]
```

feltétellel.

### 4. feladat: 4\_\_kik

Készítünk egy új lekérdezést, melyhez hozzáadjuk a *Gyűlés* táblát és a *3\_\_legtobb* lekérdezést. A két táblát a *Nap* és *Óra* mezőkkel összekapcsoljuk (szoros illesztés). A tervezőrácsba felvesszük a *Gyűlés* tábla *Nap*, *Óra*, *Név* és *Lyukas-e* mezőjét. Ez utóbbira beírjuk az "1" feltételt. A *Nap*, illetve *Óra* mezőkre növekvő rendezést írunk elő.

### 5. feladat

A feladat megoldásához két összesítő segédlekérdezésre lesz szükségünk. Az elsőben előállítjuk a tanárok névsorát (**5\_\_Névsor**). Felvesszük a *Gyűlés* táblát, a tervezőrácsban pedig csoportosítást (*Group By*) írunk elő a *Név* mezőre. A második lekérdezéshez hozzáadjuk az első lekérdezést. A tervezőrácsba felvesszük a *Név* mezőt, az összesítés sorában pedig kiválasztjuk a *Count* függvényt. Ezzel meghatároztuk az adattáblában szereplő tanárok számát (**5\_\_TanárokSzáma**).

A feladat megoldását eredményező lekérdezéshez (**5\_\_feleplusz**) hozzáadjuk a *2\_raer* és az *5\_\_TanárokSzáma* lekérdezést. A tervezőrácsba felvesszük a *2\_raer* eredménytábla *Nap*, *Óra* és *SumOfLyukas-e* mezőit. Ez utóbbira előírjuk a

```
>=[ 5__TanárokSzáma ] ! [ CountOfNév ] / 2
```

feltételt. A lekérdezés *Adatlap* nézete érthetőbb lesz, ha a tervezőrács első oszlopában a *Mező* sorba üres karaktersorozatot írunk (""), az oszlop *Címét* pedig a „Legalább a tanárok fele ráér” felirattal látjuk el.

**Megjegyzés:** a tanárok névsorát az összesítő lekérdezés helyett egy választó lekérdezés *Egyedi értékek* tulajdonságának *Igen*-re állításával is elkészíthetjük.

### 6. feladat

A jelentéseket a *Jelentés varázslóval* készítjük el.

A **2\_raer** jelentéshez felvesszük a lekérdezés összes mezőjét, előírjuk a *Nap* szerinti csoportosítást, de nem használunk rendezést. Táblázatos szerkezetet választunk, tetszőleges stílussal. A jelentés *Tervező* nézetében módosítjuk a *Jelentésfej* szövegét.

A **3\_\_legtobb** jelentéshez felvesszük a lekérdezés *Nap* és *Óra* mezőjét, nem adunk hozzá csoportszinteket, nem írunk elő rendezést, táblázatos elrendezést választunk tetszőleges stílussal. *Tervező* nézetben módosítjuk a *Jelentésfej* szövegét.

A **4\_\_kik** jelentéshez felvesszük a lekérdezés mezőit. Csoportosítást végzünk a *Nap* és az *Óra* mezők szerint. Előírjuk a *Név* mező szerinti rendezést. Tetszőleges elrendezést és stílust választhatunk. A jelentés *Tervező* nézetében csökkentjük a feliratok és beviteli mezők méretét, illetve módosítjuk a jelentésfej feliratát. Az *Óra* beviteli mezőt célszerű balra igazítani.

Az **5\_feleplusz** jelentéshez felvesszük a lekérdezés *Nap* és *Óra* mezőjét. A *Nap* mező szerint végzünk csoportosítást. Rendezést írunk elő az *Óra* mezőre. Tetszőleges elrendezést és stílust választhatunk. *Tervező* nézetben módosítjuk a *Jelentésfej* feliratát.

### Megjegyzés

A *Gyűlés* tábla adatait kereszttáblás elrendezésben szemléltethetjük (**1\_Keresztáblás**). Készítsünk egy új lekérdezést, melyhez hozzáadjuk a táblát. A tervezőrácsba sorra felvesszük a *Nap*, *Óra*, *Név* és *Lyukas-e* mezőket. A *Lekérdezés* menüből a *Keresztáblás* lekérdezés típust választjuk. A tervezőrács *Összesítés* sorában a *Nap*, *Óra* és *Név* mezőknél meghagyjuk a *Group By* parancsot, a *Lyukas-e* oszlopában pedig kiválasztjuk a *Max* függvényt. A *Keresztáblás* sorában a *Nap* és *Óra* mezőknél a *Sorfejléce*t, a *Név* mezőnél az *Oszlopfejléce*t, a *Lyukas-e* mezőnél pedig az *Értéke*t választjuk. Mivel a kereszttáblás lekérdezéseknél csak egyetlen mező lehet oszlopfejléc, ezért a feladatgyűjteményben látható elrendezést nem tudjuk létrehozni.

Szemléletesebb kereszttáblát kapunk, ha a *Lyukas-e* mező helyett a

```
IIf(Max([Lyukas-e]) = "1"; "lyukasóra")
```

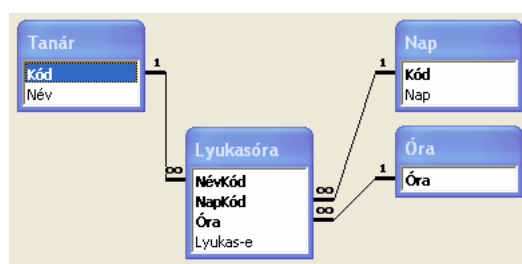
kifejezést írjuk az utolsó oszlop *Mező* sorába (**1\_Keresztáblás2**). Az *IIf* függvény hasonlóan működik az Excel *Ha* függvényéhez. Ha teljesül az első paraméterként megadott feltétel, akkor a pontosvessző utáni kifejezés értéke kerül a megfelelő cellába.

A *Gyűlésterv* kezdetű lekérdezésekre a következő feladat megoldásához van szükség.

### Gyűlésterv

A *Gyűlés* adatbázis táblájának több rekordja is tartalmazza ugyanazokat az értékeket (például egy tanár nevét). Ha megváltozik az érték (például egy másik tanár veszi át az órákat), akkor több helyen kell elvégezni a módosítást. Ezt az úgynevezett módosítási anomáliát – a szerkezet más hibáival együtt – úgy küszöböljük ki, hogy egyetlen tábla helyett több táblát készítünk. Külön táblákban helyezük el a tanárok nevét (és esetleg más adatait), a hét napjait, az órák sorszámát az egyes napokon és a lyukasórák idejét. A napokat, illetve órákat tartalmazó táblára itt nincs feltétlenül szükség, a rekordokra vonatkozó egyéb adatokat is tárolhatunk bennük (például a be- és kicsöngetés időpontját).

A táblákban azonosítóval látjuk el a rekordokat. (Az órák sorszámát tartalmazó táblában fölösleges külön azonosítót felvenni.) Az azonosító lesz a tábla kulcsa. A lyukasórákat tartalmazó táblába a *Lyukas-e* mező mellett a többi tábla kulcsát vesszük fel. A táblákat a kulcsok segítségével kapcsoljuk össze. Az adatbázis táblaterve tehát:



### 1. feladat: Nap, Óra, Tanár, Lyukasóra táblák

Készítünk egy új, üres adatbázist, melyet *Gyűlésterv* néven mentünk. Az adatbázis be is zárhatjuk. A táblákat táblakészítő lekérdezéssel hozzuk létre. Megnyitjuk a *Gyűlés* adatbázist (!), és készítünk egy új lekérdezést (**Gyűlésterv\_Tanár**). Hozzáadjuk a *Gyűlés* táblát, a tervezőrácsba pedig felvesszük a *Név* mezőt. A lekérdezést összesítővé alakítjuk, Összesítésként meghagyjuk a *Group By* záradékot.

A *Lekérdezés* menüben kiválasztjuk a *Táblakészítő* lekérdezést. A megjelenő párbeszédablakban másik adatbázisként a *Gyűléstervet* választjuk, a táblanévhez pedig beírjuk a *Tanár* elnevezést. Az *Adatlap* nézetben ellenőrizhetjük az új tábla rekordjait. Visszaváltunk *Tervező* nézetbe, majd a lekérdezés futtatásával (*Lekérdezés/Futtatás*) létrehozuk az új táblát.

A *Nap* és *Óra* táblákat hasonló módon hozzuk létre (**Gyűlésterv\_Nap**, **Gyűlésterv\_Óra** lekérdezések).

Bezárjuk a *Gyűlés* adatbázist, és megnyitjuk a *Gyűléstervet*. A benne található táblák megnyitásával ellenőrizhetjük a táblakészítő lekérdezések eredményét.

A *Tanár* és a *Nap* táblákba beillesztünk egy *Számláló* típusú mezőt, melynek a *Kód* nevet adjuk. A beillesztés a *Tervező* nézetben végezzük el (gyorsmenü: *Sorok beszúrása*). A mezőt kulcsként jelöljük meg (*Szerkesztés/Elsődleges kulcs*). Az *Óra* táblában magát az *Óra* mezőt jelöljük meg kulcsként.

Létrehozuk a *Lyukasóra* nevű táblát, amelybe felvesszük a *NévKód*, *NapKód*, *Óra* és *Lyukas-e* mezőket. A *NévKód*, *NapKód*, illetve *Óra* mezők típusa *Szám* (*Hosszú egész*), a *Lyukas-e* mező pedig *Szöveg*. A mezőtulajdonságoknál töröljük az *Alapértelmezett értéket*. A *Lyukas-e* mező *Érvényességi szabálya*: " " or "1". Kulcsot majd a 2. feladatban hozunk létre.

Megnyitjuk a *Kapcsolatok* ablakot, és hozzáadunk minden táblát. A hivatkozási integritás megőrzésével létrehozuk a következő kapcsolatokat:

*Főlérendelt tábla:*      *Alárendelt tábla:*

Tanár.Kód	→	Lyukasóra.NévKód
Nap.Kód	→	Lyukasóra.NapKód
Óra.Óra	→	Lyukasóra.Óra

A *Lyukasóra* táblát a 3. feladatban töltjük fel adatokkal.

## 2. feladat

A *Lyukasóra* tábla *Tervező* nézetében kijelöljük a *NévKód*, *NapKód* és *Óra* mezőket, majd kiválasztjuk a *Szerkesztés* menü *Elsődleges kulcs* parancsát.

## 3. feladat: 3\_Lyukasóra

Az űrlapot a varázsló segítségével hozzuk létre. Felvesszük a *Lyukasóra* tábla minden mezőjét. *Oszlopos* elrendezést választunk, tetszőleges stílussal. Az űrlap tervező nézetében a *NévKód*, *NapKód* és *Óra* beviteli mezőknél a következő módosításokat végezzük:

- A típust kombipanelre változtatjuk (gyorsmenü: *Típus megváltoztatása*)
- A *Tulajdonságok* (gyorsmenü: *Tulajdonságok*) *Adat* kartonján *Sorforrásként* megjelöljük a megfelelő táblát (*Tanár*, *Nap* illetve *Óra*).
- Rákattintunk a *Szerkesztő* gombra, és nyugtázzuk a *Lekérdezés-szerkesztő* indítását jelző üzenetet (*Igen*).
- A *Lekérdezés-szerkesztő* ablakban felvesszük a *Név* (*Nap*, *Óra*), majd (az *Óra* tábla kivételével) a *Kód* mezőt a tervezőrácsba. (A *Kód* legyen a második oszlop.) Az első oszlopra növekvő rendezést állítunk be.
- Bezárjuk a *Lekérdezés-szerkesztő* ablakot, mentjük a módosítást.
- A *Tulajdonságok* ablak *Sorforrás* rovatában megjelent egy SQL parancs. A *Kötött oszlop* tulajdonság adja meg, hogy melyik mező értékei kerülnek tárolásra. Ezt átírjuk 2-re (a lekérdezésben a második mező volt a kód). A *Csak listaelem* tulajdonság értékét *Igen*-re módosítjuk (ekkor csak a listában lévő értékek közül választhat a felhasználó).
- A *Formátum* panelen módosíthatjuk az *Oszlopszám* értékét. Ha átírjuk 2-re, akkor nem csak a lekérdezésben szereplő első oszlop (a *Név*) fog megjelenni a listában, hanem a hozzá tartozó kód is. (A megoldásban meghagytuk az eredeti értéket: 1 oszlopot.)
- Bezárjuk a *Tulajdonságok* ablakot.
- A módosításokat a *NapKód* és *Óra* beviteli mezőknél is végrehajtjuk. A kombipanelek létrehozása után módosíthatjuk az űrlapelemek elrendezését, beilleszthetünk űrlapfejet és más díszítőelemeket.

Az űrlap segítségével feltöltjük a *Lyukasóra* táblát. Mivel csak a lyukasórákat visszük be az adatbázisba, a *Lyukas-e* beviteli mező *Alapértelmezett érték* tulajdonságához 1-et írunk a *Tervező* nézetben. (Erre a mezőre tulajdonképpen nincs szükségünk.)

#### 4. feladat

**4.2.** Készítünk egy összesítő lekérdezést (**4\_raer**), melyhez hozzáadjuk a *Nap*, *Óra* és *Lyukasóra* táblát. A tervezőrácsba felvesszük a *Nap.Nap*, *Óra.Óra* és *Lyukasóra.Lyukas-e* mezőt. Az előző kettőnél meghagyjuk a *Group By* záradékot, az utóbbinál pedig a *Count* függvényt választjuk.

**4.3.** Először egy összesítő lekérdezés (**4\_LegtöbbLyukasóra**) tervezőrácsába felvesszük a *4\_raer* lekérdezés *CountOfLyukas-e* mezőjét. Az *Összesítés* sorában kiválasztjuk a *Max* függvényt. Ezzel meghatároztuk az egy adott órában ráérő tanárok számának maximumát. Készítünk egy újabb lekérdezést (**4\_legtobb**), melyhez hozzáadjuk a *4\_raer* és a *4\_LegtöbbLyukasóra* lekérdezést. A két táblát a *CountOfLyukas-e*, illetve a *MaxOfCountOfLyukas-e* mezők segítségével összekapcsoljuk (szoros illesztés). A tervezőrácsba felvesszük a *4\_raer* lekérdezés *Nap* és *Óra* mezőjét. A címek kialakítását, a tábla formázását a *Gyűlés* feladatnak megfelelően végezzük el.

**4.4.** A lekérdezéshez (**4\_kik**) hozzáadjuk a *Tanár*, *Nap*, *Óra*, *Lyukasóra* és *4\_legtobb* táblát. A *4\_legtobb* tábla *4\_raer.Nap* mezőjét összekapcsoljuk a *Nap* tábla *Nap* mezőjével, a *4\_raer.Óra* mezőjét pedig az *Óra* tábla *Óra* mezőjével (mindkettő szoros illesztés). A tervezőrácsba felvesszük a *Nap.Nap*, *Óra.Óra*, *Tanár.Név* és *Lyukasóra.Lyukas-e* mezőt. Ez utóbbi feltétele: "1", megjelenítését ki is kapcsolhatjuk. A *Nap* és *Óra* mezőkre növekvő rendezést írunk elő.

**4.5.** Először összesítő lekérdezést készítünk, amellyel meghatározzuk a tanárok számát (**4\_TanárokSzáma**). Az időpontokat meghatározó összesítő lekérdezéshez (**4\_feleplusz**) adjuk hozzá a *Nap*, *Óra* és *Lyukasóra* táblát, illetve a *4\_TanárokSzáma* lekérdezést. A tervezőrácsba vegyük fel a *Nap.Nap*, *Óra.Óra*, *Lyukasóra.Lyukas-e* és *4\_TanárokSzáma.CountOfNév* mezőket. A *Nap*, *Óra* és *CountOfNév* mezők szerint összesítünk (*Group By*), a *Lyukas-e* mezőnél pedig a *Count* függvényt választjuk ki az *Összesítés* sorában. A *Lyukas-e* mező feltétele:

```
>=[ 4_TanárokSzáma ] ! [ CountOfNév ] / 2
```

A *Lyukas-e* és a *CountOfNév* mezők megjelenítését kikapcsoljuk.

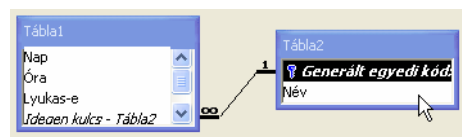
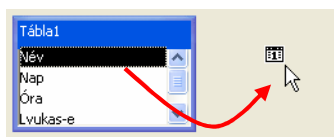
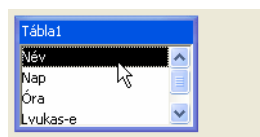
A tervezőrácsba azért kellett felvenni a *CountOfNév* mezőt, mert a feltételként megadott kifejezésbe csak olyan mezőket írhatunk, melyek szerepelnek valamelyik *Group By* záradékban. Ezzel természetesen nem hoztunk létre újabb csoportosítást.

#### Megjegyzés (Gyűlésterv2 adatbázis)

A *Gyűlés* feladat táblájának a szétválasztását (normalizálását) elvégezhetjük az adatbázis-kezelő *Táblaanalizáló* varázslójával is (*Eszközök/Analizálás/Tábla*).

Új adatbázist készítünk, és importáljuk a *Gyűlés* adatbázis *Gyűlés* tábláját. Elindítjuk a *Táblaanalizáló* varázslót. A folyamat lépései:

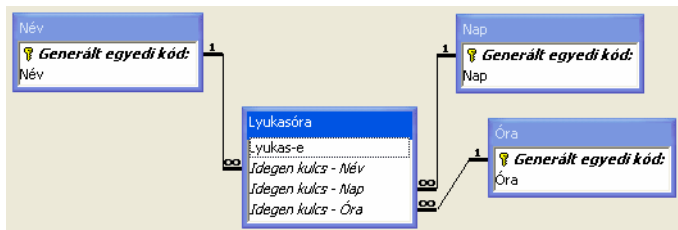
- A bevezető elolvasása után kijelöljük a *Gyűlés* táblát.
- Mi magunk szeretnénk eldönteni, hogy melyik mező melyik táblába kerüljön. (Egyébként – nagyon helyesen – csak a *Név* mező kerülne új táblába, de ez nem felelne meg a *Gyűlésterv* 1. feladatának.)
- A következő ablakban fogjuk meg az egérrel a *Név* mezőt és tegyük a tábla mellé! Megjelenik egy új tábla a *Generált egyedi kód* és a *Név* mezőkkel. Nevezzük el a táblát! Megfigyelhetjük, hogy az eredeti táblában a *Név* mező helyett létrejött egy *Idegen kulcs* mező. Menet közben a *Tippek* ikonra (💡) kattintva részletes magyarázatot kapunk a tennivalóról.



A *Név* mező kiemelése a táblából

- Végezzük el ezt a műveletet a *Nap* és *Óra* mezőkre is!
- Az eredeti táblának a *Tábla átnevezése* ikon segítségével a *Lyukasóra* nevet adhatjuk.





### Az új táblák és kapcsolatok

- f) A következő ablakban válasszuk a lekérdezés létrehozását és a *Súgó* megjelenítését, majd a *Befejezés* gombot! Olvassuk el figyelmesen a *Súgóban* látható információt!

## Tanárértékelés

A feladatok megoldásához egy új adatbázisba importáljuk a *tanaert.txt* táblát. Az első sor tartalmazza a mezőneveket, a típusokat nem módosítjuk, és nem hozunk létre elsődleges kulcsot. A táblát a feladatgyűjteménynek megfelelően kiegészítettük El Emília adataival.

### 1. feladat: 1\_sz\_atlag

Összesítő lekérdezést készítünk, melyhez hozzáadjuk a *Tanárértékelés* táblát. A tervezőrácsba felvesszük a *Szempont* (*Group By*) és a *Pont* (*Avg*) mezőt. Ez utóbbi tulajdonságai közül a *Formátumot Rögzített*re módosítjuk 2 tizedessel. Az oszlop címe: „A pontszámok átlaga”.

### 2. feladat: 2\_t\_sz\_atlag, 2\_Keresztábrás

Az előző feladat megoldását kiegészítjük a *Tanár* mező felvételével, melynél szintén a *Group By* záradékot választjuk az *Összesítés* sorában.

Az átlagok megjelenítése sokkal szemléletesebb egy keresztábrás lekérdezésben (lásd a Gyűlés 6. feladatánál szereplő megjegyzést).

### 3. feladat

Egy új lekérdezéshez hozzáadjuk az *1\_sz\_atlag* lekérdezést. A tervezőrácsba felvesszük a *Szempont* és az *AvgOfPont* mezőt. Ez utóbbira csökkenő rendezést írunk elő. A lekérdezés tulajdonságainál a *Csúcsértéket* 1-re állítjuk. Így kapjuk meg a legjobb átlaggal rendelkező szempontot (**3\_sz\_max**). A leggyöngébb eredmény kimutatásához ugyanilyen lekérdezést készítünk, csak növekvő rendezést írunk elő (**3\_sz\_min**).

**Megjegyzés:** nagyméretű adatbázisok esetén a legnagyobb vagy legkisebb érték kiválasztása sokkal rövidebb ideig tart, mint az eredménytábla rendezése. Ezért a csúcsérték beállítása helyett célszerű egy összesítő lekérdezésben meghatározni a szélsőértéket, majd ennek felhasználásával megjeleníteni a rekord adatait (lásd: **3\_sz\_MaxÁtlag**, **3\_sz\_LegjobbSzempont**).

### 4. feladat: 4\_osszpont

Összesítő lekérdezést hozunk létre, melybe felvesszük a *Tanárértékelés* tábla *Tanár* (*Group By*), *Diák* (*Group By*) és *Pont* (*Sum*) mezőjét. A megjelenítés áttekinthetőbb, ha keresztábrás lekérdezést használunk (**4\_Keresztábrás**).

### 5. feladat: 5\_sorrend

Létrehozunk egy új összesítő lekérdezést, melynek tervezőrácsába felvesszük a *Tanár* tábla *Tanár* (*Group By*) és *Pont* (*Sum*) mezőjét. Ez utóbbinál csökkenő rendezést írunk elő.

### 6. feladat

Először létrehozunk egy összesítő lekérdezést, melyben meghatározzuk a Kis Irma által adott pontok összegét az egyes tanároknál (**6\_irmaaranya**). A lekérdezéshez hozzáadjuk a *Tanárértékelés* táblát, a tervezőrácsba pedig felvesszük a *Tanár* (*Group By*), a *Pont* (*Sum*) és a *Diák* (*Where*) mezőket. Ez utóbbi *Feltétel* sorába beírjuk Kis Irmát.

A következő lekérdezéshez (**6\_irmaaranya**) hozzáadjuk az összpontszámokat tartalmazó *5\_sorrend* és a *6\_irmaaranya* táblát. A két táblát a *Tanár* mezőn keresztül szoros illesztéssel

összekapcsoljuk (különben a lekérdezésben a két tábla Descartes-szorzatát kapnánk meg, az összes rekordot az összes rekorddal párosítva). A tervezőrácsba felvesszük a *6\_\_irmaanya* tábla *Tanár* és *SumOfPont* mezőjét. A következő oszlop *Mező* sorába beírjuk a

```
[5_sorrend]![SumOfPont]/[6__irmaanya]![SumOfPont]
```

kifejezést. Ez utóbbi oszlop címe: „Szorzótényező”. A számformátumot *Rögzített*re állítjuk, 2 tizedeshellyel.

## 7. feladat

Készítünk egy új lekérdezést (**7\_irmamax**), melyben megjelenítjük a *6\_irmaanya* lekérdezése *Tanár* és *Kifl* mezőjét. A *Kifl* mezőnél csökkenő rendezést állítunk be. A lekérdezés tulajdonságainál a *Csúcsérték*hez 1-et írunk. Így kapjuk meg a maximális arány értékét. A legkisebb arány meghatározását hasonló módon végezzük, csak növekvő rendezést választunk (**7\_irmamin**). A megoldással kapcsolatban lásd a 3. feladat megjegyzését.

## 8. feladat

**1\_sz\_atlag, 2\_Keresztábrás, 2\_t\_sz\_atlag, 3\_sz\_max, 3\_sz\_min, 4\_Keresztábrás, 4\_osszpont, 5\_sorrend, 6\_irmaanya, 7\_irmamax, 7\_irmamin**

A jelentéseket a *Jelentés varázslóval* állítjuk elő az egyes eredménytáblák alapján. A *Jelentésfej* feliratait a feladatnak megfelelően módosítjuk. A *2\_t\_sz\_atlag* és a *4\_osszpont* jelentéseket a *Tanár* mező szerint csoportosítjuk.

Az értékelések átlagát bemutató keresztábrás jelentés *Tervező* nézetében jelöljük ki az átlagokat tartalmazó beviteli mezőket, majd a *Tulajdonságoknál* állítsunk be rögzített számformátumot 2 tizedeshellyel!

## A tanárértékelés terve

### 1. Diák, Értékelés, Szempont, Tanár táblák

A táblákat létrehozhatjuk táblakészítő lekérdezéssel és *Táblaanalizáló* varázslóval is a *Gyűlésterv* adatbázis táblája alapján (lásd a *Gyűlésterv* 1. feladatát és megjegyzését). A *Kód* mezők alapján a hivatkozási integritás megőrzésével összekapcsoljuk a táblákat.

### 2. feladat

Kijelöljük az *Értékelés* tábla *Tervező* nézetében a *TanárKód*, *DiákKód* és *SzempontKód* mezőket, majd kiválasztjuk a *Szerkesztés* menü *Elsődleges kulcs* parancsát.

### 3. 3\_Értékelés

A megoldás részletes leírását a *Gyűlésterv* 3. feladatánál olvashatjuk. A *Pont* mezőhöz beírjuk a  $\leq 10$  érvényességi szabályt.

### 4. feladat

**4.1.** Az összesítő lekérdezéshez (**1\_sz\_atlag**) hozzáadjuk a *Szempont* és az *Értékelés* táblát, a tervezőrácsba felvesszük a *Szempont.Szempont* (*Group By*), illetve az *Értékelés.Pont* (*Avg*) mezőt. A mező címét és a *Számformátumot* a Tanárértékelés 1. feladatának megfelelően állítjuk be.

**4.2.** Az összesítő lekérdezéshez (**2\_t\_sz\_atlag**) hozzáadjuk a *Tanár*, *Szempont* és *Értékelés* táblát. Töröljük a *Tanár* és a *Szempont* táblák közti kapcsolatot. A tervezőrácsba felvesszük a *Tanár.Tanár* (*Group By*), *Szempont.Szempont* (*Group By*) és *Értékelés.Pont* (*Avg*) mezőt. A *Pont* oszlop tulajdonságait a Tanárértékelés 2. feladatának megfelelően állítjuk be. A lekérdezést keresztábrás formájában is megjeleníthetjük (**2\_Keresztábrás**).

**4.3.** A legjobb átlagot megjelenítő lekérdezéshez (**3\_sz\_max**) hozzáadjuk az *1\_sz\_atlag* táblát. A tervezőrácsba felvesszük a tábla mezőit. Az *AvgOfPont* mezőre csökkenő rendezést írunk elő. A lekérdezés tulajdonságainál a *Csúcsérték*hez 1-et írunk. Az első oszlop címét a Tanárértékelés 3. feladatának megfelelően módosítjuk.

A leggyengébb átlagot megjelenítő lekérdezést hasonló módon készítjük el, csak növekvő rendezést írunk elő (**3\_sz\_min**).

**4.4.** Az összesítő lekérdezéshez (**4\_összpont**) hozzáadjuk a *Tanár*, *Diák* és *Értékelés* táblát. Töröljük a *Tanár* és a *Diák* tábla közötti kapcsolatot. A tervezőrácsba felvesszük a *Tanár.Tanár* (*Group By*), *Diák.Diák* (*Group By*) és *Értékelés.Pont* (*Sum*) mezőt. A *Pont* mező címe: „Összpontszám”.

A lekérdezést kereszttáblás formában is megjeleníthetjük (**4\_Kereszttáblás**).

**4.5.** Az összesítő lekérdezéshez (**5\_sorrend**) hozzáadjuk a *Tanár* és az *Értékelés* táblát. A tervezőrácsba felvesszük a *Tanár.Tanár* (*Group By*) és az *Értékelés.Pont* (*Sum*) mezőt. Ez utóbbira csökkenő rendezést írunk elő. Az oszlop címe: „Összpontszám”.

**4.6.** Először elkészítjük azt az összesítő lekérdezést, amely a Kis Irma által adott pontokat összegezi az egyes tanároknál (**6\_irmaaranya**). Hozzáadjuk a *Tanár*, *Diák*, illetve *Értékelés* táblát, töröljük a *Tanár* és *Diák* tábla közötti kapcsolatot, a tervezőrácsba pedig felvesszük a *Tanár.Tanár* (*Group By*), *Értékelés.Pont* (*Sum*) és *Diák.Diák* (*Where*) mezőt. Ez utóbbinál a feltétel: "Kis Irma".

A feladat megoldását szolgáltató lekérdezéshez (**6\_irmaaranya**) hozzáadjuk az *5\_sorrend* és a *6\_irmaaranya* lekérdezést. A *Tanár* mezőn keresztül kapcsolatot hozunk létre a két lekérdezés között. A tervezőrácsba felvesszük a *6\_irmaaranya* tábla *Tanár* és *SumOfPont* mezőjét. A következő oszlop *Mező* sorába beírjuk a

```
[5_sorrend]![SumOfPont]/[6_irmaaranya]![SumOfPont]
```

kifejezést. Az oszlopokat a Tanárértékelés 6. feladatának megfelelő címmel látjuk el.

**4.7.** A lekérdezésekhez (**7\_irmamax**, **7\_irmamin**) hozzáadjuk a *6\_irmaaranya* lekérdezést, a tervezőrácsokba felvesszük a *Tanár* és a *Kifl* mezőt. A maximum meghatározásánál csökkenő, a minimumnál pedig növekvő rendezést állítunk be a *Kifl* mezőre. A lekérdezés *Csúcsérték* tulajdonságait 1-re állítjuk.

## Európa

A feladatok megoldása előtt egy üres adatbázisba importáljuk az *europa.txt* táblát (lásd a Csoportcím 1. feladatát). Az első sor tartalmazza a mezőneveket, nem módosítjuk a felajánlott mezőtípusokat, és nem adunk meg elsődleges kulcsot. A tábla tervező nézetében a *Terület* és a *Lakosság* mező címéhez az elnevezés mellett beírjuk a mértékegységet is.

### 1. feladat

A feladatot először lekérdezéssel oldjuk meg (**01\_suru**), mert nem célszerű adattáblában olyan mezőt felvenni, amely a többi mező alapján kiszámítható adatokat tartalmaz. A lekérdezéshez hozzáadjuk az *Európa* táblát. A *Tervezőrácsba* felvesszük az *Ország*, a *Lakosság* és a *Terület* mezőt. A következő oszlop *Mező* sorába beírjuk a

```
[Lakosság]*1000000/[Terület]
```

kifejezést. Az oszlop tulajdonságainál beállítjuk a *Rögzített* formátumot 2 tizedessel és a „Népsűrűség” címet.

A feladat megfogalmazásának megfelelően az adattáblát is kiegészíthetjük egy új mezővel (**Európa2** tábla). Először elkészítjük a tábla másolatát (*Fájl/Mentés másként*), mert a továbbiakban az eredeti adattáblát szeretnénk használni. Az új táblát kiegészítjük egy *Szám* (*Egyszeres*) típusú mezővel, amit frissítő lekérdezéssel töltünk fel (*01\_Frissítő*). Az új lekérdezéshez hozzáadjuk az *Európa2* táblát, majd átalakítjuk frissítővé (*Lekérdezés/Frissítő lekérdezés*). A tervezőrácsba felvesszük a *Népsűrűség* mezőt, amelynél a *Módosítás* sorába beírjuk a

```
[Lakosság]*1000000/[Terület]
```

kifejezést. A módosítást a lekérdezés futtatásával végezzük el (*Lekérdezés/Futtatás*). Figyeljünk arra, hogy a frissítő lekérdezés minden egyes megnyitásakor végrehajtjuk a rekordok változtatását!

### 2. feladat

Lekérdezést készítünk (*02\_moviszony*), melyhez hozzáadjuk az *Európa* táblát. A tervezőrácsba felvesszük az *Ország* és a *Terület* mezőt, ez utóbbi megjelenítését kikapcsoljuk. A következő oszlop *Mező* sorába beírjuk a

kifejezést. A *Tulajdonságoknál Rögzített* formátumot választunk 3 tizedessel.

Az előző megoldásban felhasználtuk, hogy Magyarország területe állandó, és ismert érték. Ha egy lekérdezésben változó mennyiségeket tartalmazó kifejezést alkalmazunk, akkor célszerű először egy segédlekérdezéssel (**02\_MagyarországTerülete**) meghatározni a mennyiség aktuális értékét. A segédlekérdezéshez hozzáadjuk az *Európa* táblát. A tervezőrácsba felvesszük az *Ország* és a *Terület* mezőt. Az *Ország* mező feltételéhez beírjuk Magyarországot. A területek arányát meghatározó lekérdezést (**02\_TerületViszony**) az első megoldáshoz hasonlóan hozzuk létre, de a következő kifejezést alkalmazzuk:

```
[Európa]![Terület]/[02_MagyarországTerülete]![Terület]
```

A hivatkozás miatt a lekérdezéshez hozzá kell adnunk a *02\_MagyarországTerülete* táblát is.

### 3. feladat: 03\_Főváros

A lekérdezéshez hozzáadjuk az *Európa* táblát, a *Tervezőrácsba* felvesszük az összes mezőt (\*). Mivel a *Főváros* mezőre rendezést szeretnénk előírni, ezért ezt a mezőt külön is felvesszük, beállítjuk a növekvő rendezést, de kikapcsoljuk a megjelenítését.

**Megjegyzés:** a rendezéshez nem szükséges lekérdezést készíteni. Ha az *Európa* tábla *Adatlap* nézetében rákattintunk egy fővárosra, majd kiválasztjuk a *Rekordok* menü *Rendezés* parancsát, akkor növekvő és csökkenő rendezést is beállíthatunk.

### 4. feladat

Először összesítő lekérdezést készítünk, melyben égtájanként összeadjuk az országok területét (**04\_Terület**). A lekérdezéshez hozzáadjuk az *Európa* táblát, majd a tervezőrácsba felvesszük az *Égtáj* (*Group By*) és a *Terület* (*Sum*) mezőt.

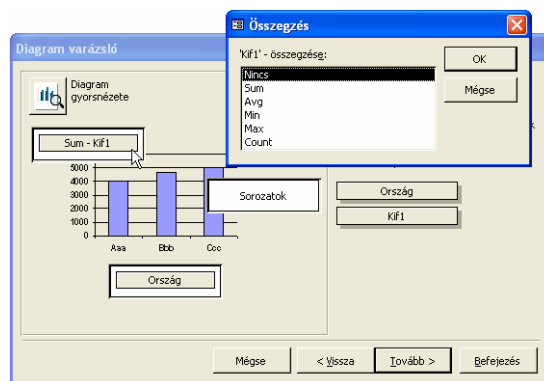
A diagramot jelentésként készítjük el (**04\_TerületDiagram**). A *Beszúrás/Jelentés* menüponttal megjeleníthető *Új jelentés* ablakban kiválasztjuk a *Diagram varázslót* és a *04\_Terület* lekérdezést. A diagram létrehozásának lépései:

- Felvesszük a lekérdezés mindkét mezőjét.
- Kiválasztjuk a kördiagramot.
- A gyorsnézet gombbal ellenőrizzük a diagram megjelenését.
- Címet adunk a diagramnak („Területi eloszlás”).

A diagramot a *jelentés Tervező* nézetében módosíthatjuk. A módosításhoz kattintsunk duplán a diagramra. A dupla kattintásra bejelentkezik az *MS Graph*, melyet a táblázatkezelőknél megszokott módon használhatunk a diagram formázására.

### 5. feladat: 05\_NépsűrűségDiagram

A diagramot az előző feladat megoldásához hasonlóan hozzuk létre a *01\_suru* lekérdezés alapján. A diagram adatait az *Ország* és a *Kif1* mezők tartalmazzák. Típusként oszlopdiagramot választunk. A következő lépésnél duplán kattintunk a függőleges tengely tetején lévő dobozra, majd a megnyíló *Összegezés* ablakban a *Nincs* listaelemet választjuk. Címet adunk a diagramnak, de nem kérünk magyarázatot.



A diagram tervező nézetében fekvő oldalt állítunk be (*Fájl/Oldalbeállítás//Oldal, Fekvő helyzet*). 24 cm-re növeljük a diagram szélességét, és 10 cm-re a magasságát. A vízszintes tengely formázásánál 6 pontos Times New Roman betűtípust választunk *Normál* stílussal. Az *Igazításnál* az elforgatást 90°-ra módosítjuk. A függőleges tengelyen a betűméretet 10 pontosra, a *Skála értéktengelyén* a *Fő léptéket* pedig 200-ra állítjuk.

#### 6. feladat: 06\_Összterület

Összesítő lekérdezést hozunk létre, melyhez hozzáadjuk az *Európa* táblát. A tervezőrácsba felvesszük a *Terület* mezőt (*Sum*). Módosítjuk az oszlop *Címét* („Összterület (km<sup>2</sup>)”) és szélességét.

#### 7. feladat: 07\_Összlakosság

Összesítő lekérdezést hozunk létre, melyhez hozzáadjuk az *Európa* táblát. A tervezőrácsba felvesszük a *Lakosság* mezőt (*Sum*). Módosítjuk az oszlop *Címét* („Összlakosság (millió fő)”) és szélességét.

#### 8. feladat: 08\_ÁtlagosNépsűrűség

A lekérdezéshez hozzáadjuk az *Európa* táblát. A tervezőrács első oszlopának *Mező* sorába beírjuk a  $\text{Sum}([Lakosság]) * 1000000 / \text{Sum}([Terület])$

kifejezést. Módosítjuk az oszlop *Címét* („Átlagos népsűrűség (fő/km<sup>2</sup>)”) és szélességét. Az értéket 2 tizedessel jelenítjük meg.

#### 9. feladat: 09\_NépsűrűségekÁtlaga

Összesítő lekérdezést hozunk létre, melyhez hozzáadjuk a *01\_suru* lekérdezést. A tervezőrácsba felvesszük a *Kifl* mezőt (*Avg*). Módosítjuk az oszlop *Címét* („A népsűrűségek átlaga (fő/km<sup>2</sup>)”) és szélességét. Az értéket 2 tizedessel jelenítjük meg.

#### 10. feladat: 10\_OrszágokSzama

Összesítő lekérdezést hozunk létre, melyhez hozzáadjuk az *Európa* táblát. A tervezőrácsba felvesszük az *Ország* mezőt (*Count*). Módosítjuk az oszlop *Címét* („Az országok száma”) és szélességét.

#### 11. feladat: 11\_LegkevesebbLakos

Lekérdezést hozunk létre, melyhez hozzáadjuk az *Európa* táblát. A tervezőrácsba felvesszük az *Ország* és a *Lakosság* mezőt. Növekvő rendezést állítunk be a *Lakosság* mezőre. A lekérdezés tulajdonságainál a *Csúcsértéket* 1-re állítjuk.

#### 12. feladat: 12\_LegtöbbLakos

Az előző feladathoz hasonlóan oldjuk meg, csak csökkenő rendezést állítunk be.

#### 13. feladat: 13\_LegkisebbNépsűrűség

Lekérdezést hozunk létre, melyhez hozzáadjuk a *01\_suru* lekérdezést. A tervezőrácsba felvesszük az *Ország* és a *Kifl* mezőt. Növekvő rendezést állítunk be a *Kifl* mezőre. A lekérdezés tulajdonságainál a *Csúcsértéket* 1-re állítjuk. Módosítjuk a *Kifl* oszlop címét („Legkisebb népsűrűség (fő/km<sup>2</sup>)”) és szélességét. Az értéket 2 tizedessel jelenítjük meg.

#### 14. feladat: 14\_LegnagyobbNépsűrűség

Az előző feladathoz hasonlóan oldjuk meg, csak csökkenő rendezést állítunk be.

#### 15. feladat: 15\_LegnagyobbOrszágok

Lekérdezést hozunk létre, melyhez hozzáadjuk az *Európa* táblát. A tervezőrácsba felvesszük az *Ország* és a *Terület* mezőt. Csökkenő rendezést állítunk be a *Terület* mezőre. A lekérdezés tulajdonságainál a *Csúcsértékhez* 5-öt írunk.

### 16. feladat: 16\_KisOrszágok

Lekérdezést hozunk létre, melyhez hozzáadjuk az *Európa* táblát. A tervezőrácsba felvesszük az *Ország* és a *Terület* mezőt. Ez utóbbihoz beírjuk a <93036 feltételt.

**Megjegyzés:** a feltételhez célszerűbb beírni a

< [02\_MagyarországTerülete]![Terület]

kifejezést (lásd a 2. feladat megoldását). Ekkor hozzá kell adni a 02\_MagyarországTerülete táblát is a lekérdezéshez.

### 17. feladat: 17\_SokLakos

Lekérdezést hozunk létre, melyhez hozzáadjuk az *Európa* táblát. A tervezőrácsba felvesszük az *Ország* és a *Lakosság* mezőt. Ez utóbbihoz beírjuk a >10 feltételt.

### 18. feladat: 18\_KevésLakos

Az előző feladathoz hasonlóan oldjuk meg, csak a <10 feltételt írjuk be.

### 19. feladat

A feladatot úgy értelmezzük, hogy az egyes égtájakhoz tartozó országok területi megoszlását kérdezi. A megfelelő lekérdezést már a 4. feladat megoldásánál elkészítettük (04\_Terület).

### 20. feladat: 20\_LegnagyobbNyugati

Lekérdezést hozunk létre, melyhez hozzáadjuk az *Európa* táblát. A tervezőrácsba felvesszük az *Ország*, a *Terület* és az *Égtáj* mezőt. Ez utóbbinál megadjuk a „Nyugat” feltételt, és kikapcsoljuk a megjelenítést. A *Terület* mezőnél csökkenő rendezést állítunk be. A lekérdezést tulajdonságainál a *Csúcsérték*hez 1-et írunk.

### 21. feladat: 21\_LegtöbbKeleti

Az előző feladathoz hasonlóan oldjuk meg, csak a *Terület* helyett a *Lakosság* mezőt vesszük fel a tervezőrácsba, és a „Kelet” feltételt írjuk az *Égtáj* mezőhöz.

### 22. feladat: 22\_LegritkábbKözépNyugat

A lekérdezéshez hozzáadjuk az *Európa* táblát és a 01\_suru lekérdezést. Az *Ország* mezőn keresztül összekapcsoljuk a két objektumot (enélkül a lekérdezésben az összes lehetséges rekordpárosítás megjelenne – Descartes-szorzat). A tervezőrácsba felvesszük a lekérdezésből *Ország* és a *Kif1*, illetve az *Európa* táblából az *Égtáj* mezőt. Ez utóbbi megjelenítését kikapcsoljuk, a *Feltétel* sorába pedig a

"Közép" Or "Nyugat"

kifejezést írjuk. A *Kif1* mezőnél növekvő rendezést állítunk be. A lekérdezést tulajdonságainál a *Csúcsérték*et 1-re állítjuk.

**Megjegyzés:** a logikai művelet helyett a Közép és Nyugat feltételeket írhattuk volna a tervezőrács két sorába egymás alá.

### 23. feladat

Az *Európa* tábla tervező nézetében vesszük fel az újabb mezőt. Típusának *Igen/Nem*-et választunk. Az EU-tagokat a tábla *Adatlap* nézetében tudjuk bejelölni.

**Megjegyzés:** a táblából hiányzik Ciprus, amely szintén az Európai Unió tagja.

### 24. feladat

A területeket egy összesítő lekérdezéssel tudjuk meghatározni (**24\_Területek**). A lekérdezéshez hozzáadjuk az *Európa* táblát. A tervezőrácsba felvesszük az *EUtag* (*Group By*) és a *Terület* (*Sum*) mezőt. Ez utóbbi címét módosítjuk („Összterület”).

A kördiagramot a lekérdezésre alapozva a 4. feladathoz hasonlóan hozzuk létre

(**24\_TerületekDiagram**). Címet nem adunk, és nem kérjük a magyarázat megjelenítését sem.

Tervező nézetben két *Felirat* mezőt helyezünk el a diagram mellett, melyek megmagyarázzák a körcikkek jelentését.

**Megjegyzés:** egy logikai (*Igen/Nem*) típusú mező esetén is használhatjuk a *Diagram varázsló* által készített magyarázatot, ha *Kombinált listát* használunk a lekérdezésben a megjelenítéshez.

### 25. feladat: 25\_KevésLakosEUtag

Lekérdezést hozunk létre, melyhez hozzáadjuk az *Európa* táblát. A tervezőrácsba felvesszük az *Ország*, a *Lakosság* és az *EUtag* mezőt. Ez utóbbinál megadjuk az *Igen* feltételt, és kikapcsoljuk a megjelenítést. A *Lakosság* mezőnél növekvő rendezést állítunk be. A lekérdezést tulajdonságainál a *Csúcsérték*hez 3-mat írunk.

### 26. feladat: 26\_KözépEurópaiEUtagok

Lekérdezést hozunk létre, melyhez hozzáadjuk az *Európa* táblát. A tervezőrácsba felvesszük az *Ország*, az *Égtáj* és az *EUtag* mezőt. Ez utóbbiaknál beírjuk a „Közép”, illetve az *Igen* feltételt, és kikapcsoljuk a megjelenítést. Az *Ország* mező címe: „Közép-európai EU-tagok”.

### 27. feladat: 27\_NemEUtagokSzamaÉgtájanként

Összesítő lekérdezést hozunk létre, melyhez hozzáadjuk az *Európa* táblát. A tervezőrácsba felvesszük az *Égtáj* (*Group By*), az *Ország* (*Count*) és az *EUtag* (*Where*) mezőt. Ez utóbbinál beírjuk a *Nem* feltételt. Az *Ország* oszlop címe: „Nem EU-tagok száma”.

### 28. feladat: 28\_EUösszesítések

Összesítő lekérdezést hozunk létre, melyhez hozzáadjuk az *Európa* táblát. A tervezőrácsba felvesszük a *Terület* (*Sum*) és a *Lakosság* (*Sum*) mezőt. A következő oszlopba beírjuk a

`[SumOfLakosság]*1000000/[SumOfTerület]`

kifejezést, az *Összesítés* sorában *Expression*-t választunk. Az értéket 2 tizedessel jelenítjük meg. Az oszlopok címét a tartalmuknak megfelelően módosítjuk.

### 29. feladat

Először az országok listáját készítjük el (**29\_MagyarországhozHasonló**). A lekérdezéshez hozzáadjuk az *Európa* táblát. A tervezőrácsba felvesszük az *Ország* és a *Terület* mezőt. Ez utóbbinál beírjuk a

`Between 50000 And 200000`

feltételt.

Az országok megszámlálásához összesítő lekérdezést készítünk az előző lekérdezés alapján (**29\_HasonlóakSzama**). A tervezőrácsba felvesszük az *Ország* mezőt (*Count*). Az oszlop címe: „A Magyarországhoz hasonló területű országok száma”.

**Megjegyzés:** ha nem a lekérdezés elkészítése a feladat, akkor az országok (rekordok) számát leolvashatjuk az első lekérdezés *Adatlap* nézetében a rekordnavigátor sorában.

### 30. feladat

A lekérdezéshez (**30\_EUadatok**) hozzáadjuk az *Európa* táblát. A tervezőrácsba felvesszük az *Ország*, a *Főváros* és a *Lakosság* mezőt. A következő oszlop *Mező* sorába beírjuk a

`Kerek([Lakosság])`

kifejezést. Az oszlop címe: „Lakosság kerekítve”.

A jelentést (**30\_EUadatok**) a varázslóval készítjük el az előző lekérdezésből kiindulva. A *Kifl* mező alapján végzünk csoportosítást, és beállítjuk az országok neve szerinti növekvő rendezést.

**Megjegyzés:** a *Kerek* (*Round*) függvény második paramétere meghatározza a tizedesjegyek számát (ha hiányzik, akkor egész értékre történik a kerekítés). A függvény a 0,5-et páros egészrész esetén lefelé, páratlan esetén pedig fölfelé kerekíti, például:  $4,5 \approx 4$ ;  $5,5 \approx 6$ . Lásd még a Magatartás, szorgalom 1. feladatának megoldását.

### 31. feladat

Ha a jelentés alapjául szolgáló mindegyik lekérdezés csak egyetlen rekordot (adatot) tartalmaz, akkor készíthetünk egy újabb lekérdezést, amelybe felvesszük ezeket az értékeket, és erre a lekérdezésre alapozzuk a jelentést (lásd a Csoportcím 9. feladatának a megoldását).



Sajnos esetünkben két ország, Málta és Izland rendelkezik a legkevesebb lakossal. Az első megoldásban ezt figyelmen kívül hagyjuk, és csak az egyiket vesszük fel a jelentésbe. Készítsünk lekérdezést, melyhez hozzáadjuk a 6.–14. feladatokban elkészített lekérdezéseket (**31\_EurópaAdatai**). A tervezőrácsba felvesszük az egyes lekérdezésekben szereplő mezőket. Az eredménytábla két sorból fog állni (Málta és Izland miatt), mert az adatbázis-kezelő minden rekordot minden rekorddal összepárosít (Descartes-szorzat). Ezért a lekérdezés *Csúcsérték* tulajdonságához 1-et írunk.

A jelentést (**31\_EurópaAdatai**) az így létrehozott lekérdezésre alapozzuk. Hozzáadjuk az összes mezőt, nem végzünk csoportosítást, és függőleges szerkezetet választunk. Tervező nézetben módosítjuk a *Feliratok* és a *Beviteli mezők* szélességét, helyzetét, igazítását. Módosítjuk a *Jelentésfej* feliratát is. Az egységes méretezéshez és elrendezéshez használjuk a *Formátum* menü *Igazítás*, *Méretezés*, *Vízszintes távolság* és *Függőleges távolság* almenüit!



A második megoldást (**31\_EurópaAdatai2**) szintén a *31\_EurópaAdatai* lekérdezés alapján készítjük el, de nem vesszük fel a legkevesebb lakosra vonatkozó két mezőt. Nem adunk hozzá csoportszintet, és függőleges szerkezetet választunk. A *11\_LegkevesebbLakos* lekérdezés beillesztéséhez segédjelentést használunk. (Ennek módját a 33. feladat megoldása ismerteti részletesebben.) A létrehozás során nem csatoljuk a főürlapot a segédürlaphoz (jelentéshez). A beillesztés után elvégezzük a formázás és elrendezés szükséges módosításait.

**Megjegyzés:** egyszerűbb a második megoldás elkészítése, ha létrehozzuk az első megoldás másolatát (*Fájl/Mentés másként*), és abba illesztjük be a segédjelentést.

### 32. feladat

A jelentéseket a varázslóval készítjük el. Kijelöljük a megfelelő lekérdezést, felvesszük a mezőket, nem végzünk csoportosítást és táblázatos elrendezést választunk. (A 19. feladathoz készített jelentésnél a *04\_Terület* lekérdezést használjuk fel.) A tervező nézetben módosítjuk a *Jelentésfej* feliratát.

### 33. feladat

Több egymáshoz nem kapcsolódó lekérdezés alapján segédjelentések segítségével tudunk készíteni összevont jelentést. Először létrehozunk egy üres jelentést a *Tervező* nézetben. Az *Eszközkészletben* bekapcsoljuk a  *Vezérlőelem* varázslót, majd az *Eszközkészletben* kiválasztjuk a  *Segédürlap/segédjelentés* vezérlőelemet, amellyel egy téglalapot rajzolunk a jelentés törzsébe. Ezzel elindítjuk a varázslót, amely végigvezet a folyamaton. A segédjelentés létrehozásának a lépései:

- Meglévő lekérdezést használunk.
- Kiválasztjuk a *25\_KevésLakosEUtag* lekérdezést, és kijelöljük az *Ország* mezőt.
- A varázsló a segédjelentést elmenti az adatbázisba, ezért rákérdez a nevére. Meghagyjuk a felajánlott nevet.
- Elkészül az összevont jelentésbe illesztett segédjelentés, melyet a *Tervező* nézetben méretezhetünk, áthelyezhetünk, formázhatunk. A segédjelentések ugyanolyan szakaszokból állnak, mint bármely más jelentés.

A 26., 27., 28. és 29. feladat lekérdezéséből hasonló módon készítünk az összevont jelentésbe illesztett segédjelentést. A *29\_HasonlóakSzáma* lekérdezést nem használtuk fel. Helyette a *29\_MagyarországhozHasonló* lekérdezésből készült segédjelentés *Jelentésláába* illesztettük be az `=Count([Ország])` kifejezést tartalmazó beviteli mezőt.

## Magyarország

### 1. feladat: Magyarország

Új adatbázist készítünk, melybe importáljuk a *magyaro.txt* fájl tábláját (lásd: Csoportcím, 1. feladat). Az első sor tartalmazza a mezőneveket, nem módosítjuk a felajánlott adattípusokat, és nem adunk elsődleges kulcsot a táblához. A *Tervező* nézetben ékezetesre módosítjuk a mezőneveket.



**Megjegyzés:** a CD-n lévő fájlban hibás a budapesti vállalkozások száma. Ezt átjavítjuk 17,1-ről 171-re. Mivel a táblában Budapest is szerepel a megyék között, a feladatok megoldásában a fővárost szintén megyének tekintjük.

## 2. feladat: 02\_S

Lekérdezést készítünk, melyhez hozzáadjuk a *Magyarország* táblát. A tervezőrácsba felvesszük a *Megye* és a *Népsűrűség* mezőt, ez utóbbi megjelenítését kikapcsoljuk. A *Népsűrűség* mező feltételéhez beírjuk az S betűt. A *Megye* oszlop címe: „Kicsi népsűrűségű megyék”.

## 3. feladat: 03\_kisokt

Lekérdezést készítünk, melyhez hozzáadjuk a *Magyarország* táblát. A tervezőrácsba felvesszük a *Megye* és a *Felsőoktatás* mezőt. A *Felsőoktatás* mező feltételéhez beírjuk a <15 kifejezést. A *Megye* oszlop címe: „A kevés hallgatóval rendelkező megyék”.

## 4. feladat: 04\_SokVállalkozó

Lekérdezést készítünk, melyhez hozzáadjuk a *Magyarország* táblát. A tervezőrácsba felvesszük a *Megye* és a *Vállalkozás* mezőt, ez utóbbi megjelenítését kikapcsoljuk. A *Vállalkozás* mező feltételéhez beírjuk a >100 kifejezést (1000 lakos 10%-a 100 fő). A *Megye* oszlop címe: „Sok vállalkozóval rendelkező megyék”.

## 5. feladat: 05\_SokVállalkozóSM

A lekérdezéshez hozzáadjuk a 04\_SokVállalkozó lekérdezést és a *Magyarország* táblát. A két táblát a *Megye* mezők segítségével összekapcsoljuk. A tervezőrácsba felvesszük a *Magyarország* tábla *Megye* és *Népsűrűség* mezőjét, ez utóbbi megjelenítését kikapcsoljuk. A *Népsűrűség* mező feltételéhez beírjuk az S Or M kifejezést. A *Megye* oszlop címe: „Kicsi vagy közepes népsűrűségű megyék sok vállalkozóval”.

**Megjegyzés:** a logikai művelet helyett két sorba egymás alá írhattuk volna az S és az M betűt a *Népsűrűség* mező feltételénél.

## 6. feladat: 06\_NépsűrűségEloszlás

Összesítő lekérdezést készítünk, melyhez hozzáadjuk a *Magyarország* táblát. A tervezőrácsba felvesszük a *Népsűrűség* (*Group By*) és a *Megye* (*Count*) mezőt. A *Megye* mező címe: „A megyék száma”.

## 7. feladat

A feladatot megoldhatnánk úgy, hogy két lekérdezést készítünk a *Lakás* mező szerinti növekvő, illetve csökkenő rendezéssel, majd a *Csúcsérték* tulajdonságot 1-re állítjuk (lásd például az Európa 11. és 12. feladatának a megoldását). Most azonban olyan megoldást mutatunk be, amelynél egyetlen lekérdezésben jelenítjük meg mindkét szélsőértéket.

Először készítünk egy összesítő lekérdezést, amelyben meghatározzuk a *Lakás* mezőben előforduló legkisebb és legnagyobb értéket (07\_\_LakásMinMax). A lekérdezéshez hozzáadjuk a *Magyarország* táblát. A tervezőrácsba kétszer felvesszük a *Lakás* mezőt. Az egyiknél a *Min*, a másiknál a *Max* függvényt választjuk ki az *Összesítés* sorában. A megfelelő megyéket meghatározó lekérdezéshez (07\_LakásMinMax) hozzáadjuk a *Magyarország* táblát és a 07\_\_LakásMinMax lekérdezést. A tervezőrácsba felvesszük a *Magyarország* tábla *Megye* és *Lakás* mezőjét. A *Lakás* mező feltételeként beírjuk a

```
[07__LakásMinMax]![MinOfLakás] Or [07__LakásMinMax]![MaxOfLakás]
```

kifejezést (a logikai művelet helyett használhattunk volna két egymás alatt lévő cellát is). A *Megye* oszlop címe: „Legkevesebb/legtöbb ember egy lakásra”, a *Lakásé* pedig „Fő/100 lakás”.

## 8. feladat: 08\_LegtöbbVállalkozás

Lekérdezést készítünk, melyhez hozzáadjuk a *Magyarország* táblát. A tervezőrácsba felvesszük a *Megye* és a *Vállalkozás* mezőt. Ez utóbbi szerint csökkenő rendezést írunk elő. A lekérdezés *Csúcsérték* tulajdonságát 1-re állítjuk. A *Megye* mező címe: „A legtöbb vállalkozás”.

### 9. feladat: 09\_LegtöbbbHallgató

Az előző feladathoz hasonlóan oldjuk meg, csak a *Vállalkozás* mező helyett a *Felsőoktatás* mezőt adjuk hozzá a tervezőrácsához.

### 10. feladat: 10\_RitkaKevésVállalkozás

Lekérdezést készítünk, melyhez hozzáadjuk a *Magyarország* táblát. A tervezőrácsba felvesszük a *Megye*, a *Népsűrűség* és a *Vállalkozás* mezőt. Ez utóbbi szerint növekvő rendezést írunk elő. A *Népsűrűség* oszlop *Feltétel* sorába beírjuk az S betűt, és kikapcsoljuk a megjelenítését. A lekérdezés *Csúcsérték* tulajdonságát 1-re állítjuk. A *Megye* mező címe: „Kis népsűrűség – legkevesebb vállalkozás”.

**Megjegyzés:** vedd össze az 5. feladat megoldásával (a kis népsűrűségű megyéket a 02\_S lekérdezés tartalmazza).

### 11. feladat: 11\_KevésLakóLX

Lekérdezést készítünk, melyhez hozzáadjuk a *Magyarország* táblát. A tervezőrácsba felvesszük a *Megye*, a *Népsűrűség* és a *Lakás* mezőt. Ez utóbbira növekvő rendezést írunk elő. A *Népsűrűség* oszlop *Feltétel* sorába beírjuk az "L" or "X" kifejezést, és kikapcsoljuk a megjelenítését. A lekérdezés *Csúcsérték* tulajdonságát 1-re állítjuk. A *Megye* mező címe: „Legkevesebb lakó – L vagy X népsűrűség”.

### 12. feladat

A feladatot úgy értelmezzük, hogy a kis népsűrűségű megyék adatait kell megjeleníteni. Mivel a 02\_S lekérdezésben a *Magyarország* tábla több mezője sem szerepel, ezért először lekérdezést készítünk, amely tartalmazza a szükséges adatokat (**12\_\_SMegyék**). A lekérdezéshez hozzáadjuk a *Magyarország* táblát. A tervezőrácsba felvesszük az összes mezőt. A *Népsűrűség* mező *Feltétel* sorába beírjuk az S betűt, és kikapcsoljuk a megjelenítését.

A jelentést a 12\_\_SMegyék lekérdezés alapján készítjük el (**12\_\_SMegyék**). Felvesszük az összes mezőt, nem adunk hozzá csoportszintet, előírjuk a *Megye* mező szerinti növekvő rendezést. *Táblázatos* megjelenítést választunk, tetszőleges stílussal. A *Tervező* nézetben módosítjuk a *Jelentésfej* feliratát, az oszlopcímeket pedig kiegészítjük a mértékegységgel. Egy felirat mezőn belül a *Shift+Enter* segítségével hozhatunk létre sortörést.

### 13. feladat: 13\_03\_kisokt

A jelentést varázslóval készítjük el a 03\_kisokt lekérdezés alapján, de csak a *Megye* mezőt vesszük fel. Nem adunk hozzá csoportszintet, és nem írunk elő rendezést a varázslóban. *Táblázatos* elrendezést választunk tetszőleges stílussal. A rendezést a *Tervező* nézetben hozzuk létre a *Nézet* menü *Rendezés és csoportosítás* parancsával. A mezőlistában kiválasztjuk a *Felsőoktatás* mezőt, növekvő sorrenddel. A *Jelentésfej* és a beviteli mezők feliratát az adatoknak megfelelően módosítjuk.

### 14. feladat

A jelentéshez először egy újabb lekérdezést készítünk azokból a lekérdezésekből, melyek eredménytáblája csak egyetlen sort tartalmaz (**14\_\_Összevont**). A lekérdezéshez a 8.–11. feladatokban elkészített lekérdezéseket adjuk hozzá a bennük szereplő mezőkkel.

A varázslóval készítünk egy új jelentést (**14\_\_Összevont**), melybe felvesszük a 14\_\_Összevont lekérdezés mezőit. Nem adunk hozzá csoportszintet, és nem írunk elő rendezést. Függőleges szerkezetet választunk tetszőleges stílussal. *Tervező* nézetben módosítjuk a *Jelentéstörzs* feliratait és a mezők szélességét. A 6. és 7. feladat lekérdezéseit segédjelentésként illesztjük az összevont jelentésbe. (A segédjelentés készítésének és beillesztésének a módját lásd az Európa 33. feladatnál.) Nem választunk a csatoláshoz mezőket.

## Kémiai elemek

### 1. feladat

Először importáljuk a *kemelem.txt* táblát (**Elem**) egy új adatbázisba. Az első sor tartalmazza a mezőneveket, a hibaüzenetet vegyük tudomásul az OK gombra való kattintással. Az *Atomsugar*, *Elektronegativitas* és *Gyakorisag* mezők típusát *Egyszeresre* módosítjuk. A szövegfájl sorainak a végén elhelyezett fölösleges tabulátorok miatt keletkező utolsó mezőt (*Mező9*) nem importáljuk (a fölösleges tabulátorok következtében kaptuk a hibaüzenetet is). A táblához nem adunk meg elsődleges kulcsot. *Tervező* nézetben módosítjuk a mezők elnevezését, *Adatlap* nézetben pedig a szélességét.

A táblához *Tervező* nézetben hozzáadunk egy *Szám (Dupla)* típusú új mezőt (*RendszámPerMóltömeg*), melyet frissítő lekérdezéssel (**01\_RendszámPerMóltömeg**) töltünk fel adatokkal. A frissítő lekérdezéshez hozzáadjuk az *Elemek* táblát. A tervezőrácsba felvesszük a *RendszámPerMóltömeg* mezőt, melynek *Módosítás* sorába beírjuk a

```
[Rendszám]/[Móltömeg]
```

kifejezést. Az új mezőt a lekérdezés futtatása tölti fel adatokkal. Az *Elem* tábla *Tervező* nézetében módosítjuk a megjelenített tizedeshelyek számát (*Formátum: Rögzített, Tizedeshelyek: 3*).

**Megjegyzés:** adattáblában nem célszerű a mezőkből számítható értékeket tárolni. Ehhez inkább lekérdezést használjunk, melynek egy oszlopába beírjuk a megfelelő kifejezést.

### 2. feladat: 02\_Sűrűség

A megoldást az előző feladathoz hasonlóan készítjük el. A frissítő lekérdezés *Módosítás* sorába a

```
1,667E9*[Móltömeg]/[Atomsugár]^3
```

kifejezést írjuk. A szorzótényező a mértékegységek átváltása miatt szükséges. A sűrűséget  $\text{kg/m}^3$ -ben kapjuk meg. Ezt jelezzük az új mező címében. (A valódi sűrűség meghatározásához az értékeket a gömb térfogatának megfelelően még  $4\pi/3$ -mal kellene elosztani.)

A MS Access eleve nem számítja ki a kifejezés értékét ott, ahol valamelyik hivatkozott mező (*Móltömeg* vagy *Atomsugár*) üres. Ha mégis szeretnénk nyomatékosítani a feltételt, akkor írjuk be a *Feltétel* sorába a

```
[Atomsugár] Is Not Null
```

kifejezést.

### 3. feladat: 03\_Elemek

A jelentést a varázslóval készítjük el. Felvesszük az *Elem* tábla összes mezőjét, nem írunk elő csoportosítást, kijelöljük a rendszám szerinti növekvő rendezést. *Táblázatos* szerkezetet választunk fekvő laptájolással. Az oszlopszélességeket a *Tervező* nézetben úgy módosítjuk, hogy a feliratok is kiférjenek a lapra.

**Megjegyzés:** Törtszámokat tartalmazó mezők esetén célszerű előírni a tizedesjegyek számát (*Rögzített* formátummal). Így kapunk helyiérték szerint egymás alá rendezett számokat.

### 4. feladat: 04\_Hányados

A diagramon a rendszám és a móltömeg hányadosát ábrázoljuk a vegyjelek függvényében (lásd az Európa példa 4. feladatát). A *Beszúrás/Jelentés* menüpontban kiválasztjuk a *Diagram* varázslót és az *Elem* táblát. Felvesszük a *Vegyjel*, majd a *RendszámPerMóltömeg* mezőket. A kapcsolatot oszlopdiagramon ábrázoljuk. A *Sum - RendszámPerMóltömeg* mezőre történő dupla kattintással letiltjuk az összegezést. Nem igénylünk magyarázatot a diagramon. A *Fájl* menü *Oldalbeállítás* parancsával elfektetjük a lapot, majd a *Tervező* nézetben megnöveljük a diagram méretét.

**Megjegyzés:** A vízszintes tengelyen célszerű lenne a vegyjelek helyett a rendszámot feltüntetni (növekvő sorrendben).

### 5. feladat

A **05\_Sűrűségek** lekérdezésbe felvesszük az *Elem* tábla *Vegyjel* és *Sűrűség* mezőjét. A *Sűrűség* mezőre előírjuk az *Is Not Null* feltételt.

A diagramot az előző feladathoz hasonlóan a lekérdezés alapján készítjük el (**05\_SűrűségDiagram**).

#### 6. feladat: 06\_Elektronegativitások

A feladatot a 4. feladathoz hasonlóan oldjuk meg.

#### 7. feladat: 07\_HuszedikSzázad

Az *Elem* tábla alapján lekérdezést készítünk, melybe felvesszük a *Név* és *Felfedezés* mezőket. A *Felfedezés* mező feltétele:

```
>"1900" And <"a"
```

A második összehasonlításra az „ókor” és a „középkor” szavak miatt volt szükség.

**Megjegyzés:** a 20. század – mint tudjuk, – 1901-ben kezdődött!

#### 8. feladat

A **08\_GyakoriElemek** lekérdezésbe felvesszük a *Vegyjel* és *Gyakoriság* mezőket. Ez utóbbira előírjuk a

```
> 0,0003
```

feltételt (a gyakoriságok nem százalékos értékekben szerepelnek a táblában). A tervezőrácsban előírjuk a gyakoriságok szerinti csökkenő rendezést.

A diagramot az előző feladatoknak megfelelően készítjük el (**08\_GyakoriElemek** jelentés).

**Megjegyzés:** a kis gyakoriságú elemek oszlopainak megjelenítéséhez a függőleges tengelyen válasszunk logaritmikus skálát. A tengely formázásánál a kategóriatengely metszéspontjára adjunk meg 0,0001-et (**08\_GyakoriElemekLog**).

#### 9. feladat

A legkisebb sűrűségű elemet meghatározó lekérdezésbe (**09\_MinSűrűség**) vegyük fel a *Név* és a *Sűrűség* mezőt. A *Sűrűség* mezőnél írjunk elő növekvő rendezést. Mivel a mező üres cellákat is tartalmaz, a *Feltételnél* adjuk meg az *Is Not Null* kifejezést. A lekérdezés *Csúcsérték* tulajdonságát állítsuk 1-re.

A legnagyobb sűrűségű elemet hasonló módon határozzuk meg, de csökkenő rendezéssel (**09\_MaxSűrűség**). A legegyszerűbben úgy járhatunk el, ha az előző lekérdezésről készítünk egy másolatot (*Fájl/Mentés másként*), majd módosítjuk a tervezőrácsot. Az *Is Not Null* feltételre nincsen szükség.

**Megjegyzés:** ha a legkisebb és legnagyobb értéket egy összesítő lekérdezésben meghatározzuk, akkor a hozzájuk tartozó elemek nevét egyetlen lekérdezésben is megjeleníthetjük (**09\_MaxMinSűrűség, 09\_MaxMinSűrűség**). A módszert a Magyarország adatbázis 7. feladatában mutattuk be.

#### 10. feladat: 10\_Ókor, 10\_Középkor, 10\_18szd, 10\_19szd, 10\_20szd

Az egyes lekérdezésekbe vegyük fel a *Név* és a *Felfedezés* mezőket. A feltételek:

	Mező:	Felfedezés
Ókor:	Feltétel:	"ókor"
Középkor:		"középkor"
18. század:		>"1700" And <"1801"
19. század:		>"1800" And <"1901"
20. század:		>"1900" And <"2001"

A feladat szövegével ellentétben a táblában egyetlen elemnél sem szerepel középkori évszám. Ha a szemléltetés kedvéért a teljes 17. századot a középkorhoz soroljuk, akkor a megfelelő feltétel:

```
"középkor" Or < "1701"
```

**Megjegyzés:** A megoldásnál figyelembe vettük, hogy például a 18. század 1701-től 1800-ig tartott!

## Magatartás, szorgalom

A feladatok megoldásához importáljuk egy új adatbázisba a *magszorg.txt* adatait. A varázsló első ablakában a *Speciális* gombra kattintva állítsuk be a *Közép-európai (Windows)* kódlapot. A szövegfájlban az *ertek* mező után fölösleges tabulátorok szerepelnek (*Mező5 – Mező10*), ezeket ne importáljuk. Az első sor tartalmazza a mezőneveket. Ne adjunk elsődleges kulcsot a táblához. A feladatok megoldása miatt javítsuk át a *Ki* mezőben *Húr Kata* nevét *Húr Katalinra*. Ehhez felhasználhatjuk a *Szerkesztés* menü *Csere* parancsát.

A 7. feladat miatt írjuk át Bán Tamás minden szorgalomjegyét 5-ösre!

### 1. feladat: 01\_Jegyek

Készítsünk csoportosító lekérdezést, amelybe felvesszük a *Kirol (Group By)*, *mi (Group By)* és *ertek (Avg)* mezőket. Az *ertek* mezőnél rögzített formátumot írjunk elő, 0 tizedessel!

A megoldáshoz elegendő volt a megjelenítésnél kerekíteni az átlagokat. A következő feladathoz azonban szükségünk lesz a kerekített értékekre, ezért ezeket is előállítjuk. Vegyünk fel egy újabb mezőt, melybe írjuk be a

```
Jegy: Int(Avg([ertek])+0,5))
```

kifejezést. Az *Összesítés* sorában válasszuk az *Expression*-t. A mező megjelenítését nem tilthatjuk meg, ezért a szélességet állítsuk 0-ra. Ekkor a további lekérdezésekben is 0 lesz a mezőszélesség, ezért szükség esetén majd növeljük meg.

A kerekítéshez az egészrész (*Int*) függvényt használtuk, mert a *Kerek* függvény a szokásostól eltérően az öt tizedet páros számoknál lefelé kerekíti (például  $2,5 \approx 2$ , de  $3,5 \approx 3$ )! A *Kerek* függvény alkalmazásához adjunk az átlaghoz egy nagyon kicsi számot, például:

```
Kerek(Avg([ertek])+0,0001; 0)
```

A függvény második paramétere a tizedesjegyek számát határozza meg.

**Megjegyzés:** ha a magatartás és a szorgalom jegyeket két külön lekérdezésben határoztuk volna meg, akkor a további feladatok megoldása némileg egyszerűbb lehetne.

### 2. feladat: 02\_MagStatisztika, 02\_SzorgStatisztika

A magatartásjegyek statisztikáját előállító összesítő lekérdezésbe vegyük fel a *01\_Jegyek* lekérdezést, kétszer a *Jegy* mezőt (*Group By*, illetve *Count*) és a *mi* mezőt (*Where*). Ez utóbbira feltételként írjuk elő a "magatartás" szót.

A szorgalomjegyek statisztikáját hasonlóan készíthetjük el a "szorgalom" feltétel alkalmazásával.

### 3. feladat: 03\_MagDiagram, 03\_SzorgDiagram

A diagramokat az előző lekérdezések alapján a diagramvarázslóval készíthetjük el. Szükség esetén módosítsuk a vízszintes, illetve a függőleges tengelyen megjelenő mezőket. A mezőgombokat az egerrel húzhatjuk rá a tengelyeknél lévő dobozokra. A jobb áttekinthetőség kedvéért célszerű megváltoztatni a függőleges tengely beosztását.

### 4. feladat: 04\_MagatartásEltérés, 04\_SzorgalomEltérés

A magatartás értékelések eltérésének számításánál adjuk hozzá a lekérdezéshez a *MagatartásSzorgalom* táblát, továbbá a *01\_Jegyek* lekérdezést. Kapcsoljuk őket össze a *Kirol* és a *mi* mezők segítségével (különben a lekérdezésben a két tábla Descartes-szorzata fog megjelenni)! Vegyük fel a tervezőrácsba a tábla *Ki*, *ertek* és *mi*, illetve a lekérdezés *AvgOfertek* mezőjét. Készítsünk egy számított mezőt az

```
Eltérés: [ertek]-[AvgOfertek]
```

kifejezés beírásával. A *Ki* mező feltétele: `[01_Jegyek]![Kirol]`, a *mi* mező feltétele pedig "magatartás" legyen! A *Ki* mező feltétele biztosítja, hogy egy-egy diák esetén csak a saját értékelésével kerül összehasonlításra a jegyek átlaga.

A *mi* mezőt nem szükséges megjeleníteni a lekérdezésben.

A szorgalom jegyekre vonatkozó lekérdezést hasonlóan készíthetjük el.

## 5. feladat: 05\_MagAlábecslés, 05\_MagFölébecslés, 05\_SzorgAlábecslés, 05\_SzorgFölébecslés

A 4. feladat lekérdezéseiben szereplő eltérések minimumát, illetve maximumát kell meghatározni a diákok nevével együtt. Adjuk hozzá a lekérdezésekhez a *04\_MagatartásEltérés*, illetve a *04\_SzorgalomEltérés Ki és Eltérés* mezőjét. Írjunk be a lekérdezések *Csúcsérték* tulajdonságához 1-et. Állítsunk be az alábecslésnél növekvő, a fölébecslésnél pedig csökkenő rendezést.

## 6. feladat: 06\_MagJóBecsítés, 06\_SzorgJóBecsítés

Az előző lekérdezésekhez hasonlóan oldjuk meg, de az *Eltérés* mező helyett az

```
Eltér: Abs([Eltérés])
```

kifejezés értékeit rendezzük, a csúcsérték megjelenítésével.

## 7. feladat: 07\_Példás

A lekérdezéshez a 4. feladat lekérdezéseit adjuk hozzá, majd összekapcsoljuk őket a *Ki* mező segítségével. A tervezőrácsba felvesszük a *Ki* mezőt és mindkét lekérdezésből az *ertek* mezőt. Mindkét *ertek* mező feltételéhez beírunk 5-öt.

## 8. feladat

Először segédlekérdezést készítünk (**08\_\_LegjobbJegy**), melyben meghatározzuk a legjobb magatartásjegyet (ez nem biztos, hogy 5-ös). Az összesítő lekérdezésbe felvesszük a *04\_MagatartásEltérés* lekérdezés *ertek* mezőjét (*Max*).

A legkevésbé szorgalmas diákokat listázó lekérdezéshez (**08\_JóRossz**) hozzáadjuk a segédlekérdezést és a *04\_SzorgalomEltérés*, illetve *04\_MagatartásEltérés* táblát. Ez utóbbi két táblát összekapcsoljuk a *Ki* mező segítségével. Összekapcsoljuk a *04\_Magatartáseltérés* lekérdezése *ertek* mezőjét és a *08\_LegjobbJegy* lekérdezés *MaxOfertek* mezőjét is.

A tervezőrácshoz hozzáadjuk a *04\_SzorgalomEltérés* tábla *Ki* és *ertek* mezőjét. Ez utóbbira növekvő rendezést írunk elő, a megjelenítését ki is kapcsolhatjuk. A lekérdezés *Csúcsérték* tulajdonságát 1-re állítjuk.

## 9. feladat

Először egy segédlekérdezésben (**09\_\_VáltozóSzorgalom**) kiválogatjuk a változó szorgalmú diákokat.

A feladat megoldását előállító összesítő lekérdezéshez (**09\_MagatartásÁtlag**) hozzáadjuk a segédlekérdezést és a *01\_Jegyek* lekérdezést. A két lekérdezést összekapcsoljuk a *Kirol* mező segítségével. A tervezőrácsba felvesszük a *01\_Jegyek* lekérdezés *AvgOfertek* (*Avg*) és *mi* (*Where*) mezőjét. Ez utóbbira előírjuk a "magatartás" feltételt.

## 10. feladat: 10\_Magatartás

A jelentést a szokásos módon hozzuk létre a *04\_MagatartásEltérés* lekérdezés *Ki* és *ertek* mezőinek felvételével. *Tervező* nézetben helyezzünk el a jelentéstörzsbe egy újabb beviteli mezőt, amelybe a következő kifejezést írjuk:

```
=IIf([ertek]=5; "példás"; IIf([ertek]=4; "jó"; IIf([ertek]=3; "változó";  
IIf([ertek]=2; "rossz"; "hiba"))))
```

Az *IIf* függvény az Excel *Ha* függvényéhez hasonló szerepet játszik.

Általánosabb megoldáshoz jutunk, ha először készítünk egy új táblát, melyben felsoroljuk az egyes jegyek szóbeli minősítését (**10\_\_Minősítés**). Készítsünk egy új lekérdezést is, amelyhez hozzáadjuk a *04\_MagatartásEltérés* lekérdezést és a *10\_Minősítés* táblát (**10\_\_Értékelés**). Kapcsoljuk őket össze az *ertek* mező segítségével. A tervezőrácshoz adjuk hozzá a *Ki*, az *ertek* és a *Minősítés* mezőket. A lekérdezés futtatásakor megkapjuk a jegyek szöveges értékelését. A jelentés elkészítésénél a megfelelő mezők felvételén kívül nincsen más tennivalónk.

## Lemezbolt

Bár az MS Access közvetlenül is meg tud nyitni dBase fájlokat, a feladat szövegének megfelelően importáljuk a *Lemezbolt.dbf* rekordjait egy új adatbázisba (*Beszűrés/Tábla/Tábla importálása*).

### 1. feladat: 01\_A

Lekérdezést készítünk, melyhez hozzáadjuk a *Lemezbolt* tábla *SZERZO*, *CIM* és *TIPUS* mezőjét. A *SZERZO* mezőre előírjuk a "Republic" feltételt. Szükség esetén letilthatjuk a mező megjelenítését.

**Megjegyzés:** ha csak az egymástól különböző címeket akarjuk megjeleníteni (a típus nélkül), akkor a lekérdezés *Egyedi értékek* tulajdonságát állítsuk *Igen*-re (**01\_Címek**).

### 2. feladat: 02\_B

Összesítő lekérdezést készítünk a *CIM* (*Count*) és a *SZERZO* (*Where*) mező felvételével. Ez utóbbira előírjuk az "LGT" feltételt.

**Megjegyzés:** a számláláshoz (*Count*) általában célszerű kulcsmezőt felhasználni. Mindenképpen olyan mezőt válasszunk, amely egyik rekordnál sem üres!

### 3. feladat: 03\_C

Összesítő lekérdezést készítünk a *CIM* (*Count*) és a *TIPUS* (*Where*) mező felvételével. Ez utóbbira előírjuk a "CD" feltételt.

### 4. feladat: 04\_D

Összesítő lekérdezést készítünk a *SZERZO* (*Group By*) és a *CIM* (*Count*) mező felvételével. A *SZERZO* mezőre növekvő rendezést írunk elő.

### 5. feladat: 05\_E

A lekérdezésbe felvesszük a *SZERZO*, *CIM*, *AR* és *TIPUS* mezőket. A feltételek:

Mező:	<i>SZERZO</i>	<i>AR</i>	<i>TIPUS</i>
Feltétel:	"Demjén Ferenc"	>3000	"CD"

A *TIPUS* mező megjelenítését le is tilthatjuk.

### 6. feladat: 06\_F

A lekérdezésbe felvesszük a *SZERZO* és az *AR* mezőket. Ez utóbbira csökkenő rendezést írunk elő. A lekérdezés *Csúcsérték* tulajdonságát 1-re állítjuk. Szükség esetén letiltjuk az *AR* mező megjelenítését.

### 7. feladat

A megoldásban kihasználjuk, hogy a CD-n és kazettán is megjelent kiadványok címének első 12 karaktere megegyezik. (A táblát célszerű lett volna úgy feltölteni adatokkal, hogy ezek címek teljesen megegyezzenek. A CD vagy kazetta darabszámát inkább külön mezőben tároljuk!)

Először segédlekérdezést készítünk (**07\_\_Címek**), amely tartalmazza a *SZERZO* és a *TIPUS* mezőket, illetve a

```
Kif1: Left([CIM]; 12)
```

kifejezés alkalmazásával a *CIM* mező első 12 karakterét.

A mindkét adathordozón megjelent kiadványokat összesítő lekérdezéssel listázzuk ki

(**07\_CDésKazetta**), melybe felvesszük a segédlekérdezés *SZERZO* (*Group By*), *Kif1* (*Group By*) és *TIPUS* (*Count*) mezőjét. Ez utóbbira előírjuk az =2 feltételt. A mező megjelenítését le is tilthatjuk.

**Megjegyzés:** megoldásunk szépséghibája, hogy a lekérdezésben a címnek csak az első 12 karaktere jelenik meg. Ezt elkerülhetjük, ha a *Lemezbolt* tábla *CIM* mezőjéből töröljük a „(2 CD)”-hez, „(2 MC)”-hez hasonló utalásokat. Ezek egyébként is ellentmondanak annak az elvnek, hogy a mezők minél egyszerűbb adatot tartsanak. A darabszám megjelölése nem tartozik a címhez!

## Kard

Készítsünk adatbázist, melynek egy új táblájába importáljuk a *Kard.xls* adatait. Az első sor tartalmazza az oszlopfejléceket. Elsődleges kulcsként válasszuk ki az *Azonosító* mezőt. Ha pontosan követni akarjuk a feladat szövegét, akkor nyissuk meg a táblát *Tervező* nézetben, és töröljük az *Azonosító* mezőt (vagy ne is importáljuk). Vegyük fel újra, de *Számláló* típusal, majd jelöljük ki elsődleges kulcsként. (A típus módosítását csak így tudjuk elvégezni.)

### 1. feladat: 01\_Aranyérmesek

A lekérdezésbe vegyük fel a *Név*, *Év*, *Helyszín* és *Helyezés* mezőt. Ez utóbbira adjuk meg az =1 feltételt, majd tiltsuk le a megjelenítését. Az *Év* mezőre írjunk elő növekvő rendezést.

### 2. feladat: 02: München

A lekérdezésbe vegyük fel a *Név*, *Helyezés* és *Helyszín* mezőt. Ez utóbbira adjuk meg a "München" feltételt, majd tiltsuk le a megjelenítését.

### 3. feladat: 03\_KárpátiRudolf

A lekérdezésbe vegyük fel a *Helyszín*, *Év*, *Helyezés* és *Név* mezőt. Ez utóbbira adjuk meg a "Kárpáti Rudolf" feltételt, majd tiltsuk le a megjelenítését.

### 4. feladat: 04\_MagyarÉrmekSzáma

Az összesítő lekérdezésbe vegyük fel a *Helyezés* (*Count*) és az *Ország* (*Where*) mezőt. Ez utóbbira adjuk meg a "MA" feltételt.

### 5. feladat: 05\_MagyarÉrmesek

Az összesítő lekérdezésbe felvesszük a *Név* (*Group By*), a *Helyezés* (*Count*) és az *Ország* (*Where*) mezőt. Ez utóbbira előírjuk a "MA" feltételt.

### 6. feladat: 06\_SzázadEleje

A lekérdezésbe felvesszük a *Név*, *Ország*, *Év* és *Helyezés* mezőt. A feltételek:

Mező:	Ország	Év
Feltétel:	<>"MA"	Between 1900 And 1950

### 7. feladat: 07\_TöbbszörösBajnok

Az összesítő lekérdezésbe felvesszük a *Név* (*Group By*), az *Év* (*Count*) és a *Helyezés* (*Where*) mezőt. A feltételek:

Mező:	Év	Helyezés
Feltétel:	>1	1

**Megjegyzés:** vegyük észre, hogy a *Helyezés* mezőnél a feltétel a rekordokra, az *Év* mezőnél pedig a megjelenítésre vonatkozik.

### 8. feladat: 08\_AranyérmekOrszágoként

Az összesítő lekérdezésbe felvesszük az *Ország* (*Group By*), az *Év* (*Count*) és a *Helyezés* (*Where*) mezőt. Ez utóbbinál előírjuk az =1 feltételt.

### 9. feladat

Először egy segédlekérdezést készítünk (09\_HelyszínÉv), amely tartalmazza a helyszíneket és az éveket. Az összesítő lekérdezésben mindkét mezőnél a *Group By* parancsot választjuk ki.

A megoldást szolgáltató összesítő lekérdezéshez (09\_Többször) hozzáadjuk a segédlekérdezés *Helyszín* (*Group By*) és *Év* (*Count*) mezőjét. Ez utóbbira előírjuk a >1 feltételt.

### 10. feladat: 10\_Olaszok

A lekérdezésbe felvesszük az *Egyéni* tábla *Év*, *Helyszín* és *Ország* mezőjét. Ez utóbbira előírjuk az "OL" feltételt, majd kikapcsoljuk a megjelenítését. A rekordokat az *Év* mező szerint növekvő sorba rendezzük. A lekérdezés *Csúcsérték* tulajdonságát 1-re, *Egyedi értékek* tulajdonságát pedig *Igen*-re állítjuk.

### 11. feladat

Először segédlekérdezést készítünk, amely meghatározza Kovács Pál 1950 előtti helyezését (11\_KovácsPál). Felvesszük az *Egyéni* tábla *Név*, *Év* és *Helyezés* mezőjét a következő feltételekkel:

Mező:	Név	Év
Feltétel:	"Kovács Pál"	<1950



Itt kihasználtuk, hogy Kovács Pál 1950 előtt csak egy érmet nyert.

A **11\_AzonosÉrem** lekérdezéshez hozzáadjuk az *Egyéni* táblát és a segédlekérdezést. A két táblát a *Helyezés* mező segítségével összekapcsoljuk. A tervezőrácsba felvesszük az *Egyéni* tábla *Név* és *Év* mezőjét a következő feltételekkel:

Mező:	Név	Év
Feltétel:	<>"Kovács Pál"	<1950

## 12. feladat: 12\_OrszágokEredmények

A jelentést a varázsló segítségével hozzuk létre az *Egyéni* tábla alapján. Felvesszük az *Ország*, a *Helyezés*, illetve a *Név* mezőt. Csoportosítunk az *Ország* és a *Helyezés* szerint. Szükség esetén a *Tervező* nézetben módosítjuk a jelentés formátumát.

## 13. feladat: 13\_Éremtábla (lekérdezés és jelentés)

Az áttekinthetőség kedvéért kereszt táblás lekérdezést készítünk. Felvesszük az *Egyéni* táblát, majd átváltunk *Kereszt táblás* típusra. A tervezőrácsba hozzáadjuk az *Ország* (*Group By*, *Sorfejléc*), a *Helyezés* (*Group By*, *Oszlopfejléc*) és a *Név* (*Count*, *Érték*) mezőt.

A jelentést varázslóval készítjük el. Felvesszük a *13\_Éremtábla* lekérdezés összes mezőjét. Nem adunk hozzá csoportszinteket. Az országok kiemelését (például félkövér betűk) és a mezők szélességének módosítását a *Tervező* nézetben végezzük el.

## 14. feladat: 14\_Öt néltöbb Érem (lekérdezés és jelentés)

Az összesítő lekérdezésbe felvesszük az *Egyéni* tábla *Ország* (*Group By*) és *Név* (*Count*) mezőjét. Ez utóbbira előírjuk a *>5* feltételt.

A jelentést a lekérdezés alapján készítjük el.

## Uralkodók

Elvégezzük az *uralkodo.dbf* fájl importálását egy új adatbázisba. Töröljük az *AZ* mezőt, majd újra létrehozuk *Számláló* típusúval. Kijelöljük elsődleges kulcsnak.

A további bonyodalmak elkerülése végett szűrjük be a hiányzó szóközt a 12-es azonosítójú II. Bélánál, illetve a 33-as azonosítójú I. Ulászlónál a sorszám és a keresztnév közé.

## 1. feladat: 01\_Névsor

A feladatot egyszerűen megoldhatjuk úgy, hogy a táblára (vagy egy lekérdezésre) előírunk *Név* szerinti rendezést. Ebben az esetben azonban a nevek előtt lévő római számokat (mint latin betűket) is figyelembe veszi a program a rendezéskor.

Elegánsabb megoldást kapunk, ha szétválasztjuk a sorszámot (vagy más előtagot) és a keresztnévet. Erre a névben szereplő pont (.) karaktert használhatjuk. Készítsünk egy lekérdezést, amelybe felvesszük a *NÉV*, a *KEZDŐ* és a *VÉGSŐ* mezőt. Egészítsük ki két számított mezővel, melyek a következő kifejezéseket tartalmazzák:

```
Kif1: Left([NÉV];InStr(1;[NÉV];"."))
```

```
Kif2: LTrim(Right([NÉV];Len([NÉV])-Len([Kif1])))
```

Az első kifejezés kiemeli a névből az első pontig a karaktereket. A második kifejezés a pont után következő részt határozza meg, de elhagyja a baloldaltól az esetleges szóközt (*LTrim* függvény). A lekérdezés rendezését a *Kif2* mező szerint az *Adatlap* nézetben írjuk elő, különben a program paraméterként rákérdez a *Kif1* értékére. Ha zavar a megjelenítésük, akkor csökkentsük nullára a *Kif1* és *Kif2* mezők szélességét.

**Megjegyzés:** a rendezés során „(Kis) Károly” a lista elejére kerül. Ezt kiküszöbölhetnénk egy bonyolultabb kifejezés alkalmazásával. Helyette inkább kerüljük el a zárójel (és más speciális karakterek) alkalmazását az olyan mezőben, amely nevet tartalmaz. A feladat nem írta elő az azonos keresztnévek esetén a (római) sorszám szerinti rendezést, így ezt nem is végeztük el. (Lásd az Árpád-ház 1. feladatát.)

## 2. feladat: 02\_Lászlók

Vegyük fel a lekérdezésbe a *NÉV* mezőt a

Like "\*lászló\*" And Not Like "\*ulászló\*"

feltétellel. Az uralkodás hosszát a

Kif1: [VÉGSŐ]-[KEZDŐ]

kifejezéssel számítjuk ki.

### 3. feladat: 03\_HosszSor

Az előző feladathoz hasonlóan oldjuk meg, de nem írunk elő feltételt a *NÉV* mezőre. A lekérdezést a *Kif1* mező szerint rendezzük.

### 4. feladat: 04\_IstvánokSzama

Az összesítő lekérdezésbe felvesszük az *AZ (Count)* és a *Név (Where)* mezőt. Ez utóbbinál megadjuk a Like "\*istván\*" feltételt.

### 5. feladat: 05\_BudaiUdvar

A lekérdezésbe felvesszük a *NÉV*, a *KEZDŐ* és a *VÉGSŐ* mezőket a következő feltételekkel:

Mező:	KEZDŐ	VÉGSŐ
Feltétel:	<= 1347	>= 1347

**Megjegyzés:** ha az évszám helyett mindkét feltételbe beírjuk az [Év?] kifejezést, akkor olyan paraméteres lekérdezést kapunk, amely a megadott évszámhoz megkeresi az akkor uralkodó királyt (05\_MikorKi). Próbáljuk ki a lekérdezést az 1163-as évszám beírásával.

### 6. feladat: 06\_TízÉv

A lekérdezést a 3. feladatra alapozva készítjük el. A *Kif1* mezőnél előírjuk a >10 feltételt.

### 7. feladat: 07\_KirályokSzama

Az összesítő lekérdezésbe felvesszük az *Uralkodó* tábla *NÉV (Count)*, *KEZDŐ (Where)* és *VÉGSŐ (Where)* mezőjét. A két évszámot tartalmazó mezőre vonatkozó feltétellel logikai „vagy” műveletet végzünk:

Mező:	KEZDŐ	VÉGSŐ
Feltétel:	Between 1300 And 1399	
vagy:		Between 1300 And 1399

**Megjegyzés:** a lekérdezést a feladat szövegének megfelelően készítettük el, bár a XIV. század 1301-től 1400-ig tartott! Lásd még az Árpád-ház 14. feladatának megoldását!

### 8. feladat

Az al-lekérdezésben meghatározzuk Mátyás uralkodásának kezdő évét (08\_\_MátyásKezd):

Mező:	NÉV	KEZDŐ
Feltétel:	"Mátyás"	

A királyokat összesítő lekérdezésbe (08\_MátyásElőtt) felvesszük az al-lekérdezést és az *Uralkodó* táblát. A tervezőrácshoz hozzáadjuk az *Uralkodó* tábla *NÉV (Count)* és *VÉGSŐ (Where)* mezőjét. Ez utóbbira előírjuk a

<=[08\_\_MátyásKezd]![KEZDŐ]

feltételt.

### 9. feladat: 09\_MaxKirály

A 3. feladat megoldását úgy módosítjuk, hogy a lekérdezés *Csúcsérték* tulajdonságát 1-re állítjuk.

### 10. feladat: 10\_TízÉv

A jelentést a 6. feladat lekérdezése alapján készítjük el.

### 11. feladat: 11\_Lászlók

A jelentést a 2. feladat lekérdezése alapján készítjük el.

## Sydney

Importáljuk a táblát egy új adatbázisba. Töröljük az *AZ* mezőt, majd újra felvesszük *Számláló* típussal. Kijelöljük elsődleges kulcsnak. Az *EGYÉNI* mező megjelenítési tulajdonságánál *Jelölőnégyzet* típust választunk. A táblát *Pontszerzők* néven mentjük.

Megjegyezzük, hogy nem célszerű egy adatbázisban többértékű, illetve összetett mezőket kialakítani! A feladatok megoldásánál ezért nem vesszük figyelembe, hogy a *NÉV* mező egy rekordnál valójában hány nevet tartalmaz.

### 1. feladat: 01\_HelyezésiSorrend

A lekérdezésbe felvesszük a *SPORTÁG*, *HELYEZÉS* és *NÉV* mezőket (ebben a sorrendben). Az utóbbi két mezőre növekvő rendezést írunk elő.

### 2. feladat: 02\_TóthKrisztina

A lekérdezésbe felvesszük a *NÉV*, *SPORTÁG*, *VERSENYSZÁM* és *HELYEZÉS* mezőt. A *NÉV* mezőre előírjuk a

```
Like "*Tóth Krisztina*"
```

feltételt.

### 3. feladat: 03\_Kammerer

Az összesítő lekérdezésbe felvesszük az *AZ (Count)* és a *NÉV (Where)* mezőt. Ez utóbbira előírjuk a

```
Like "*Kammerer Zoltán*"
```

feltételt.

### 4. feladat: 04\_SportágHelyezésSzám

Az összesítő lekérdezésbe felvesszük a *SPORTÁG (Group By)* és az *AZ (Count)* mezőket.

### 5. feladat: 05\_Hely

Összesítő lekérdezéssel megszámloljuk a *Pontszerzők* tábla rekordjainak a számát.

### 6. feladat: 06\_EzüstökSzáma

Az összesítő lekérdezésbe felvesszük az *AZ (Count)* és a *HELYEZÉS (Where)* mezőt. Ez utóbbira előírjuk az *=2* feltételt.

### 7. feladat: 07\_Súlyemelők

A lekérdezésbe felvesszük a *NÉV* és *SPORTÁG* mezőt. Ez utóbbira előírjuk a "súlyemelés" feltételt.

### 8. feladat: 08\_Vívás

A lekérdezésbe felvesszük a *VERSENYSZÁM* és *SPORTÁG* mezőt. Ez utóbbira előírjuk a "vívás" feltételt, majd letiltjuk a megjelenítését. A lekérdezés Egyedi értékek tulajdonságát *Igen*-re állítjuk. (Ez itt nem feltétlenül szükséges, mert minden versenyszám csak egyszer fordul elő.)

### 9. feladat: 09\_DobogósKajakKenu

A lekérdezésbe felvesszük a *NÉV*, *SPORTÁG* és *HELYEZÉS* mezőt. A feltételek:

Mező:	<i>SPORTÁG</i>	<i>HELYEZÉS</i>
Feltétel:	"kajak-kenu"	<4

### 10. feladat

Egy segédlekérdezéssel meghatározzuk Igaly Diána helyezését (**10\_IgalyHelyezés**). A versenyzők felsorolásához készítünk egy új lekérdezést (**10\_IgalyvalAzonos**), amelyhez hozzáadjuk a segédlekérdezést és a *Pontszerzők* táblát. A tervezőrácsba felvesszük a *Pontszerzők* tábla *NÉV*, illetve *HELYEZÉS* mezőjét. A feltételek:

Mező:	<i>NÉV</i>	<i>HELYEZÉS</i>
Feltétel:	<>"Igaly Diána"	[10_IgalyHelyezés]![HELYEZÉS]

**Megjegyzés:** a helyezésre vonatkozó feltétel helyett a mező segítségével összekapcsolhattuk volna a *Pontszerzők* táblát és a segédlekérdezést.

### 11. feladat: 11\_Csapat (lekérdezés és jelentés)

A lekérdezésbe felvesszük a *HELYEZÉS*, *NÉV*, *SPORTÁG* és *EGYÉNI* mezőt. Ez utóbbira előírjuk a Nem feltételt. A *HELYEZÉS*, illetve *NÉV* mezőkre növekvő rendezést állítunk be.

A jelentést a lekérdezés alapján készítjük el. A helyezés kiemelését csoportosítással végezzük.

### 12. feladat: 12\_Egyéni (lekérdezés és jelentés)

A lekérdezésbe felvesszük a *SPORTÁG*, *NÉV*, *HELYEZÉS* és *EGYÉNI* mezőt. A feltételek:

Mező:            *HELYEZÉS*            *EGYÉNI*

Feltétel:        <4                            Igen

A *SPORTÁG*, illetve *NÉV* mezőkre növekvő rendezést állítunk be.

A jelentést a lekérdezés alapján készítjük el. A sportág kiemelését csoportosítással végezzük.

## Járvány

### 1. feladat: Orvos tábla

A táblát *Tervező* nézetben hozzuk létre a feladatban szereplő mezőkkel. Megjegyezzük, hogy a feladat szerinti tábla nem felel meg a harmadik normálformának, mert tranzitív függéseket tartalmaz. A körzet például függ a kórháztól, azon keresztül pedig a megyétől is. Célszerű lenne ezeket a mezőket külön táblákban tárolni.

### 2. feladat: Jelentés tábla

A táblát *Tervező* nézetben hozzuk létre a feladatban szereplő mezőkkel. A betegségeket számokkal kódoltuk. Ezeket egy további tábla rendelné hozzá a megnevezésekhez.

### 3. feladat

A feladatot ilyen módon nem lehet megoldani. Ha két rekordnál az elsődleges kulcs különbözik (ezt biztosítja a *Számláló* típusú *Azonosító* mező), akkor a többi mező értéke már megegyezhet. Célszerű lenne a *Dátum*, *Orvos* és *Betegség* mezőket kijelölni elsődleges kulcsként. Így már védekezhetnénk az ismételt lejelentés ellen. A *Jelentés* táblába egyébként is fölösleges egy *Számláló* típusú mezőt felvenni.

### 4. feladat

Az összekapcsolást a *Kapcsolatok* ablakban végezzük el. Vegyük fel mindkét táblát, majd húzzuk rá az *Orvos* tábla *Kód* mezőjét a *Jelentés* tábla *Orvos* mezőjére. Jelöljük be a *Hivatkozási integritás megőrzését*.

A kapcsolat kialakítása után az *Orvos* táblában a Segéd tábla mezőre kattintva megnyithatjuk az orvos által leadott jelentések listáját.

### 5. feladat

Az *Orvos* táblába három megye 2-4 kórházának körzetében vettünk fel néhány települést. A rendelők címét és a betegek számát nem töltöttük ki, mert nem játszanak szerepet a további feladatok megoldásában.

A *Jelentés* tábla a 2005. február 1. és 2005. február 20. között leadott jelentéseket tartalmazza (hiányosan).

### 6. feladat: 06\_Influenza

Az influenza kódja legyen: 4. Az összesítő lekérdezésbe vegyük fel a *Jelentés* tábla *Dátum* (*Group By*), *Napi* (*Sum*) és *Betegség* (*Where*) mezőjét. Ez utóbbira írjuk elő az =4 feltételt (influenza).

**Megjegyzés:** ha a feltétel helyére a [Betegségkód?] kifejezést írjuk, akkor olyan paraméteres lekérdezést kapunk, amely az összegezés előtt rákérdez a betegség kódjára.

## 7. feladat: 07\_Tegnap

A kereszttáblás lekérdezésbe vegyük fel mindkét táblát. A tervezőrácshoz adjuk hozzá az *Orvos* tábla *Körzet* mezőjét (*Group By, Sorfejléc*), továbbá a *Jelentés* tábla *Betegség* (*Group By, Oszlopfejléc*), *Napi* (*Sum, Érték*) és *Dátum* (*Where*) mezőjét. Ez utóbbira adjuk meg a `Date()-1` feltételt.

**Megjegyzés:** Adatlap nézetre váltva valószínűleg üres eredménytábla jelenik meg, hiszen az adatbázis-kezelő mindig az aktuális dátumot veszi figyelembe. A lekérdezés kipróbálásához írjuk át a számítógép rendszerdátumát egy 2005. február 2. és 21. közötti értékre.

## 8. feladat: 08\_MúltHét

A kereszttáblás lekérdezésbe vegyük fel mindkét táblát. A tervezőrácshoz adjuk hozzá az *Orvos* tábla *Megye* mezőjét (*Group By, Sorfejléc*), továbbá a *Jelentés* tábla *Betegség* (*Group By, Oszlopfejléc*), *Napi* (*Sum, Érték*) és *Dátum* (*Where*) mezőjét. Ez utóbbira adjuk meg a

`Between Date()-1 And Date()-8`

feltételt.

**Megjegyzés:** mivel a 9. és 10. feladat szövege alapján nem érkezik be minden nap minden orvostól jelentés, ezért a tábla nem tartalmaz „nincs” nevű betegséget. A megbetegedés hiányára egyébként egyértelműen utal a *Napi* mező 0 értéke. A kipróbálásra vonatkozóan lásd az előző feladat megjegyzését.

## 9. feladat

Először egy összesítő segédlekérdezésben (**09\_Orvosok**) összegyűjtjük az előző napon jelentést küldő orvosokat. Felvesszük a *Jelentés* tábla *Orvos* (*Group By*) és *Dátum* (*Where*) mezőjét. Ez utóbbira megadjuk a `Date()-1` feltételt. Egy újabb összesítő lekérdezésben (**09\_OrvosokSzama**) pedig összeszámoljuk a segédjelentés rekordjait.

A megjelenítéssel kapcsolatban lásd a 7. feladat megjegyzését.

## 10. feladat: 10\_NemKüldött

Felhasználjuk a 9. feladat segédlekérdezését, amelyben kiválasztottuk azokat az orvosokat, akik küldtek be jelentést az előző napon. Most azon orvosok listájára van szükség, akik nem szerepelnek ebben a segédlekérdezésben. Adjuk hozzá az új lekérdezéshez az *Orvos* táblát, és vegyük fel a tervezőrácsba a *Kód*, illetve a *Név* mezőt. A *Kód* mező feltétele:

`Not In (Select Orvos From 09_Orvosok)`

A lekérdezést a *Név* mező szerint rendezhetjük. Mivel vannak azonos nevű orvosok, nem célszerű letiltani a *Kód* mező megjelenítését.

Figyelem! A *09\_Orvosok* lekérdezést nem szabad hozzáadni az új lekérdezéshez!

**Megjegyzés:** a tervezőráccsal nem hozhatunk létre úgynevezett SQL-specifikus lekérdezéseket (ezekről részletesebben lásd az MS Access Súgóját). Ide tartoznak a segédlekérdezések is, főleg az EXISTS, ANY, IN és ALL operátorok vagy a segédlekérdezésbe ágyazott újabb segédlekérdezések alkalmazása esetén. A beágyazott SELECT utasítást azonban beírhatjuk a tervezőrács *Mező* vagy *Feltétel* sorába.

## Kosárlabda

Importáljuk a megadott fájlokat egy új adatbázisba. Módosítsuk a *Szám* típusú mezők méretét *Hosszú egésze*re, és adjuk meg a kulcsokat. A *Statisztika* táblánál a *Sor* és a *Mez* mezők kijelölése után a *Szerkesztés* menüből tudjuk kiválasztani az *Elsődleges kulcs* parancsot. A *Mérkőzés* tábla *Hazai* mezőjének megjelenítését váltsuk át *Jelölőnégyzetre*.

Az adatbázis *Kapcsolatok* ablakában végezzük el a táblák összekapcsolását a hivatkozási integritás megőrzésével.

A *Mérkőzés* tábla 1. sorszámú rekordjánál javítsuk ki az ellenfél nevében a gépelési hibát (két szóköz van a Budai és a XI. között).

### 1. feladat: 01\_Névsor

A lekérdezésbe vegyük fel a *Játékos* tábla *Név*, *Magasság* és *Mez* mezőjét. A *Név* mezőnél írjunk elő rendezést.

### 2. feladat: 02\_SorMagasság

A lekérdezésbe vegyük fel a *Játékos* tábla *Név* és *Magasság* mezőjét. Ez utóbbinál írjunk elő csökkenő rendezést.

### 3. feladat: 03\_RavaszPont

Az összesítő lekérdezésbe vegyük fel a *Statisztika* tábla *Pont* mezőjét (*Sum*) és a *Játékos* tábla *Név* mezőjét (*Where*). Ez utóbbira írjuk elő a "Ravasz Zoltán" feltételt.

### 4. feladat: 04\_BiztosHármas

Az összesítő lekérdezéshez adjuk hozzá mindhárom táblát. A tervezőrácsba vegyük fel az alábbi mezőket és feltételeket:

Mező:	HÁRMAS	NÉV	ELLENFÉL
Tábla:	Statisztika	Játékos	Mérkőzés
Összesítés:	Sum	Where	Where
Feltétel:		"Biztos Géza"	"Vörös Rókák"

**Megjegyzés:** Biztos Gézának egyetlen hárompontos dobása sincs, a *Mérkőzés* táblában pedig nem is szerepel Vörös Rókák nevű csapat. Listázzuk ki inkább Vastag Ábel hárompontos dobásainak számát a Polimer Buksi csapata ellen.

### 5. feladat: 05\_Stat

A lekérdezésbe vegyük fel mindhárom táblát, majd a feladat szövegében említett mezőket. Állítsunk be növekvő rendezést az *ELLENFÉL* és a *HAZAI* mezőre.

### 6. feladat: 06\_FonóStat

Módosítsuk az előző feladat lekérdezését úgy, hogy a *NÉV* mezőhöz beírjuk a "Fonó Péter" feltételt.

### 7. feladat: 07\_VastagStat

Alakítsuk át az 5. feladat megoldását összesítő lekérdezéssé. Töröljük az *ELLENFÉL*, illetve a *HAZAI* mezőt. A *NÉV* mezőnél válasszuk a *Where* záradékot, a többinél pedig a *Sum* függvényt. A *NÉV* mezőre írjuk elő a "Vastag Ábel" feltételt.

A bedobott büntetők százalékanak számításához hozzunk létre egy új mezőt (*Expression*) a következő kifejezéssel:

```
Kif1: [SumOfBJó]/[SumOfBKis]
```

A mező formátumánál állítsunk be százalékos megjelenítést.

### 8. feladat: 08\_HáromLista

Az összesítő lekérdezésbe vegyük fel a *Játékos* tábla *NÉV* (*Group By*) és a *Statisztika* tábla *HÁRMAS* (*Sum*) mezőjét. A *NÉV* mezőre írunk elő rendezést.

### 9. feladat: 09\_PontÁtlag

Az összesítő lekérdezésbe vegyük fel a *Játékos* tábla *NÉV* (*Group By*) és a *Statisztika* tábla *PONT* (*Avg*) mezőjét. A *PONT* mezőre írunk elő csökkenő rendezést. Az átlag megjelenítését rögzített formátumban, két tizedessel végezzük.

### 10. feladat: 10\_Játszott

Az összesítő lekérdezésbe vegyük fel a *Játékos* tábla *NÉV* (*Group By*) és a *Statisztika* tábla *SOR* (*Count*) mezőjét.

### 11. feladat

A feladatot úgy értelmezzük, hogy minden egyes mérkőzés után meg kell határoznunk az egyes játékosok által összesen szerzett pontot, és az addigi mérkőzésekre vonatkozó átlagot. A kumulált értékek miatt jelentés készítünk.

Először létrehozunk egy lekérdezést (**11\_\_PontStat**), amely tartalmazza a *Játékos* tábla *NÉV*, a *Mérkőzés* tábla *ELLENFÉL* és *DÁTUM*, továbbá a *Statisztika* tábla *PONT* mezőjét.

Az új jelentést (**11\_\_PontStat**) a következő lépések végrehajtásával készítjük el:

1. A *Beszúrás/Jelentés* menüponttal megnyitható *Új jelentés* ablakban kiválasztjuk a *Tervező* nézetet és a *11\_\_PontStat* lekérdezést.
2. A *Nézet/Rendezés és csoportosítás* ablakban előírjuk a *NÉV*, majd a *DÁTUM* szerinti csoportosítást. Ehhez állítsuk a *NÉV*-nél a *Csoportfej*, a *DÁTUM*-nál pedig *Csoportláb* tulajdonságot *Igen*-re.
3. Helyezzünk egy beviteli mezőt a *NÉV* fejléc szakaszba. Tulajdonságlapjának *Adatpaneljén* kössük hozzá a *NÉV* mezőhöz. A hozzá tartozó feliratmezőbe írjuk be: „Név:”.
4. A *Törzs* szakaszban helyezzünk el három beviteli mezőt. Kössük őket hozzá az *ELLENFÉL*, a *DÁTUM*, illetve a *PONT* mezőhöz. Lássuk el a feliratmezőket a megfelelő utalással.
5. A *DÁTUM* lábléc szakaszban helyezzünk el három beviteli mezőt a következő feliratokkal és kifejezésekkel:

<i>Felirat:</i>	<i>Kifejezés:</i>	<i>Név:</i>
Mérkőzésszám:	=Count ( [ PONT ] )	Számlál
Összpont:	=Sum ( [ PONT ] )	Összpont
Átlag:	= [ Összpont ] / [ Számlál ]	

A *Tulajdonságok Egyéb* paneljén adjuk a mezőknek a fenti táblázatban látható nevet. A *Számlál* és *Összpont* mezők *Futó összeg* tulajdonságát állítsuk *Csoportonkénti összegezésre*. Az átlagot tartalmazó mezőt *Rögzített* formátumban, 2 tizedessel jelenítsük meg.

6. Vonalazással, a betűk formázásával alakítsunk ki áttekinthető szerkezetet.
7. Az oldalfej szélességét csökkentjük 0-ra. Az oldallábra helyezzünk el oldalszámot (*Beszúrás/Oldalszámok*).
8. Jelenítsük meg a *Jelentésfejet* (*Nézet/Jelentésfej*), és lássuk el címmel a jelentést.

## 12. feladat

Először egy-egy összesítő segédlekérdezésbe kigyűjtjük a győztes, illetve vesztes mérkőzések számát (**12\_\_Győztes**, **12\_\_Vesztes**). Mindkét lekérdezéshez hozzáadjuk a *Mérkőzés* tábla *SOR* (*Count*) és *SAJÁTPONT* (*Where*) mezőjét a *>[ELLENPONT]*, illetve a *<[ELLENPONT]* feltétellel. A feladat megoldását szolgáltató összesítő lekérdezésbe (**12\_\_CsapatPont**) felvesszük a *Mérkőzés* tábla *SAJÁTPONT* (*Avg*) és *ELLENPONT* (*Avg*) mezőjét, továbbá a két segédlekérdezés *CountOfSOR* (*First*) mezőjét. Az átlagokra írjunk elő rögzített formátumot 2 tizedessel.

## 13. feladat: 13\_Táblázat

A lekérdezésben megjelenítjük a *Mérkőzés* tábla *DÁTUM*, *ELLENFÉL*, *SAJÁTPONT* és *ELLENPONT* mezőjét.

## 14. feladat: 14\_Győzelem

A lekérdezésben megjelenítjük a *Mérkőzés* tábla *ELLENFÉL*, *DÁTUM*, *SAJÁTPONT* és *ELLENPONT* mezőjét. Ez utóbbira előírjuk a *<[SAJÁTPONT]* feltételt.

## 15. feladat: 15\_Vereség

A lekérdezésben megjelenítjük a *Mérkőzés* tábla *ELLENFÉL*, *DÁTUM*, *SAJÁTPONT* és *ELLENPONT* mezőjét. Ez utóbbira előírjuk a *>[SAJÁTPONT]* feltételt. A *DÁTUM* mezőnél beállítjuk a növekvő sorrendben történő rendezést.

## 16. feladat: 16\_BüntetőStat

Az összesítő lekérdezéshez hozzáadjuk a *Játékos* tábla *NÉV* (*Group By*), illetve a *Statisztika* tábla *BKIS* (*Sum*) és *BJÓ* (*Sum*) mezőjét. Egy újabb mezőbe (*Expression*) beírjuk a

```
Kif1: IIf(Sum([BKIS])<>0; Sum([BJÓ])/Sum([BKIS]); 0)
```

kifejezést. A mező formátumát *Százalékra* állítjuk, és előírjuk a csökkenő rendezést.

## 17. feladat: 17\_FaultStat

Az összesítő lekérdezésbe felvesszük a *Játékos* tábla *NÉV* (*Group By*), illetve kétszer a *Statisztika* tábla *FAULT* mezőjét (*Sum*, *Avg*). Az átlagokat tartalmazó mezőre előírjuk a növekvő rendezést, és rögzített formátumot állítunk be 2 tizedessel.

## 18. feladat: 18\_SzerzettPont

Először oldjuk meg a 19. feladatot, majd a segédlekérdezésére alapozva készítsük el a jelentést. Indítsuk el a *Jelentés varázslót*. Válasszuk ki a *19\_Egyéni* lekérdezés mezőit. Ne végezzünk csoportosítást. Írjunk elő név szerinti rendezést. Tervező nézetben növeljük meg a jelentésláb méretét, és helyezzünk el egy beviteli mezőt, melybe a


=Sum ( [ SumOfPONT ] )


kifejezést írjuk.

## 19. feladat

Először összesítő segédlekérdezést készítünk (**19\_Egyéni**), melybe felvesszük a *Játékos* tábla *NÉV* (*Group By*), illetve kétszer a *Statisztika* tábla *PONT* mezőjét (*Sum*, *Avg*). Az átlagot megjelenítő mezőnél rögzített formátumot állítunk be 2 tizedessel.

A feladat megoldásához oszlopos elrendezésű autoűrlapot készítünk a segédlekérdezés alapján

(**19\_Egyéni**). A lista létrehozásához kapcsoljuk be az *Eszközkészletben* a  *Vezérlőelem*

*varázslót*, majd helyezzünk el egy  kombipanelt az űrlapfejben. A varázslónál jelöljük be, hogy a program keresse meg a kiválasztott névnek megfelelő rekordot. A következő lépésben vegyük fel a *NÉV* mezőt. A továbbiakban módosíthatjuk az oszlopszélességet, majd megadhatjuk a vezérlőelem címkéjét („Keresett név:”). Az elemek átrendezésével, formázásával áttekinthetőbbé tehetjük az űrlap megjelenését.

## 20. feladat: 20\_Mérkőzés

Autoűrlapot készítünk oszlopos elrendezéssel a *Mérkőzés* tábla alapján.

## Vizsga

### 1. feladat

A feladat szövege alapján a következő táblákat készítjük el:

*Dolgozó* tábla (a dolgozók adatait tárolja)

<i>DolgozóKód</i>	a dolgozó kódja	<i>Számláló</i> , kulcs
<i>Név</i>	a dolgozó neve	<i>Szöveg</i>
<i>SzülHely</i>	a dolgozó születési helye	<i>Szöveg</i>
<i>Dátum</i>	a dolgozó születési dátuma	<i>Dátum</i>
<i>AnyjaNeve</i>	a dolgozó anyjának leánykori neve	<i>Szöveg</i>
<i>Beosztás</i>	a dolgozó munkahelyi beosztása	<i>Szöveg</i>
<i>Részleg</i>	a dolgozó munkahelyi részlege	<i>Szöveg</i>

*Vizsgafajta* tábla (a részvizsgák adatait tárolja)

<i>FajtaKód</i>	a részvizsga kódja	<i>Számláló</i> , kulcs
<i>Képesítés</i>	a képesítés megnevezése	<i>Szöveg</i>
<i>Vizsga</i>	a részvizsga megnevezése	<i>Szöveg</i>
<i>Költség</i>	a részvizsga ára	<i>Szám</i>

*Vizsga* tábla (a vizsgák adatait tárolja)

<i>VizsgaKód</i>	a vizsga kódja	<i>Számláló</i> , kulcs
<i>DolgozóKód</i>	a vizsgázó dolgozó kódja	<i>Szám</i>
<i>Dátum</i>	a vizsga dátuma	<i>Dátum</i>
<i>FajtaKód</i>	a letett részvizsga kódja	<i>Szám</i>
<i>Sikeres</i>	sikeres volt-e a vizsga	logikai ( <i>Igen/Nem</i> )
<i>Eredmény</i>	a vizsga százalékos eredménye	<i>Szám</i>




A táblák között a következő kapcsolatokat alakítjuk ki a hivatkozási integritás megőrzésével:  
*Dolgozó.DolgozóKód* → *Vizsga.DolgozóKód*, *Vizsgafajta.FajtaKód* → *Vizsga.FajtaKód*

## 2. feladat: 02\_DolgozóŰrlap, 02\_VizsgafajtaŰrlap

Oszlopos autoűrlapokat hozunk létre a megfelelő tábla alapján.

## 3. feladat: 03\_VizsgaŰrlap

Először oszlopos autoűrlapot készítünk a *Vizsga* tábla alapján. Utána váltsunk át *Tervező* nézetbe. Mivel a *VizsgaKód* mezőt nem kell (nem is lehet) kitölteni, ezért törölhetjük az űrlapból. Változtassuk kombipanelre a *DolgozóKód* mező típusát (*Formátum/Típus megváltoztatása*). Nyissuk meg a mező tulajdonságlapját. Az *Adatpanelen* kattintsunk a *Sorforrás* beviteli mezőjére, majd a megjelenő  *Lekérdezés-szerkesztő* ikonra. Hozzunk létre lekérdezést a *Dolgozó* tábla alapján. A tervezőrácsba vegyük fel a *DolgozóKód* és *Név* mezőt (ebben a sorrendben). A szerkesztőablak zárásakor mentjük a módosítást és frissítsük a tulajdonságot (*Igen* válasz a kérdésre).

Az *Adatpanelen* végezzük el a következő módosításokat:

Kötött oszlop:	1
Csak listaelem:	Igen
Automatikus kiterjesztés:	Igen

A *Formátum-panelen* végrehajtandó beállítások:

Oszlopszám:	2
Oszlopszélességek:	0 cm; 3 cm

A *FajtaKód* beviteli mező módosítását hasonló módon hajtjuk végre. (A lekérdezést a *Vizsgafajta* tábla alapján hozzuk létre.) Az űrlapon megjelenítettük a *Képesítés* és *Vizsga* mezőt is.

**Megjegyzés:** a kombipaneleket a *Vezérlőelem* varázslóval is elkészíthetjük. Lásd például a Kosárlabda 19. feladatát.

## 4. feladat: 04\_Vizsgaűrlap2

A 3. feladatban elkészített űrlap *Dátum* mezőjének tulajdonságlapján írjuk be az *Alapértelmezett értékhez* (*Adatpanel*) a

```
DLast("Dátum"; "Vizsga")
```

kifejezést, a vizsga típusát jelölő mezőnél pedig a

```
DLast("FajtaKód"; "Vizsga")
```

kifejezést.


## 5. feladat

Lásd a táblákban szereplő rekordokat. Az adatok beírásához az elkészített űrlapokat használtuk.

## 6. feladat: 06\_SikeresVizsgák

A lekérdezésbe felvesszük mindhárom táblát a bennük szereplő mezőkkel (kivéve a kódokat tartalmazó mezőket). A *Sikeres* mezőre előírjuk az *Igen* feltételt.

## 7. feladat: 07\_SikeresJelentés

A jelentést a varázslóval készítjük el. Felvesszük az előző feladat lekérdezésének *Részleg*, *Név*, *Képesítés* és *Vizsga* mezőjét. Előírjuk a részlegenkénti és név szerinti csoportosítást. A jelentés *Tervező* nézetében megjelenítjük a *Részleg* szakasz láblécét (*Nézet/Rendezés és csoportosítás*), és elhelyezünk benne egy  oldaltörés vezérlőelemet.

## 8. feladat: 08\_Részvizsga

A lekérdezéshez adjuk hozzá a *06\_SikeresVizsgák* lekérdezés *Név*, *Részleg* és *Vizsga* mezőjét. Ez utóbbihoz írjuk be a

```
[Melyik részvizsga?]
```

feltételt. A lekérdezés futtatásakor az adatbázis-kezelő rákérdez a részvizsga nevére (paraméteres lekérdezés).

### 9. feladat: 08\_RészvizsgaJelentés

Táblázatos autojelentést készítünk az előző feladat lekérdezése alapján. A jelentés *Tervező* nézetében előírjuk a *Részleg* szerinti csoportosítást, és a nevek alapján történő rendezést (*Nézet/Rendezés és csoportosítás*).

Helyezzünk el a jelentésfejben egy beviteli mezőt, melybe írjuk be a

=*[Melyik részvizsga?]*

kifejezést.

Az oldallábban lévő *=Now()* kifejezést cseréljük le a

*=Date()* & " Készítette: X. Y."

kifejezésre.

### 10. feladat

Először összesítő segédlekérdezést készítünk (**10\_\_UtolsóNap**), melyben meghatározzuk a legutolsó vizsganap dátumát. A tervezőrácshoz hozzáadjuk a *Vizsga* tábla *Dátum (Max)* mezőjét. A feladat megoldását szolgáltató lekérdezéshez (**10\_\_UtolsóNapEredmény**) hozzáadjuk a *Dolgozó*, a *Vizsga* és a *VizsgaFajta* táblát, illetve a segédlekérdezést. A tervezőrácsba felvesszük a *Dolgozó* tábla *Név* mezőjét, a *VizsgaFajta* tábla *Képesítés* és *Vizsga* mezőjét, továbbá a *Vizsga* tábla *Eredmény* és *Dátum* mezőjét. Ez utóbbit összekapcsoljuk a *10\_\_UtolsóNap* tábla *MaxOfDátum* mezőjével.

### 11. feladat: 11\_UtolsóNapEredmény

A *Jelentés varázslóval* készítjük el. Felvesszük az előző feladat lekérdezésének *Képesítés*, *Vizsga*, *Név* és *Eredmény* mezőjét. Előírjuk a *Képesítés* majd a *Vizsga* szerinti csoportosítást. Beállítjuk először az *Eredmény* szerinti csökkenő, másodszor a *Név* szerinti növekvő rendezést. A jelentésfejben elhelyezünk egy beviteli mezőt a következő kifejezéssel:

=*"A "* & *[Dátum]* & *"-i vizsga eredményei"*

### 12. feladat: 12\_Vizsganap

A lekérdezéshez hozzáadjuk a *Dolgozó*, *Vizsga* és *Vizsgafajta* táblát. A tervezőrácsba felvesszük a *Vizsgafajta* tábla *Képesítés* és *Vizsga* mezőjét, a *Dolgozó* tábla *Név* mezőjét (ilyen sorrendben), illetve a *Vizsga* tábla *Dátum* és *Sikeres* mezőjét. Ez utóbbira megadjuk az *Igen* feltételt, a *Dátum* mezőre pedig az

*[Írja be a dátumot]*

feltételt. Az első három mezőre előírjuk a növekvő rendezést. Végezetül letiltjuk a *Dátum* és a *Sikeres* mező megjelenítését.

### 13. feladat: 13\_VizsgaÖsszesítés

Az összesítő lekérdezéshez hozzáadjuk a *Dolgozó*, *Vizsga* és *Vizsgafajta* táblát. A tervezőrácsba felvesszük a *Vizsgafajta* tábla *Képesítés (Group By)* és *Vizsga (Group By)* mezőjét, a *Dolgozó* tábla *Név (Count)* mezőjét, illetve a *Vizsga* tábla *Sikeres (Where)* mezőjét. Ez utóbbira előírjuk az *Igen* feltételt.

### 14. feladat: 14\_VizsgaÖsszesítésJelentés

Táblázatos autojelentést készítünk az előző feladat lekérdezése alapján. Megnöveljük a *Jelentésláb* méretét, majd egy beviteli mezőbe beírjuk a következő kifejezést:

=*Sum(CountOfNév)*

### 15. feladat: 15\_Vizsganapok

Az összesítő lekérdezéshez hozzáadjuk a *Vizsga* és *Vizsgafajta* táblát. A tervezőrácsba felvesszük *Vizsga* tábla *Dátum* (*Group By*) mezőjét, illetve a *Vizsgafajta* tábla *Képesítés* (*Group By*) és *Vizsga* (*Group By*) mezőjét.

**Megjegyzés:** a feladatot összesítés nélkül is megoldhatjuk, ha a lekérdezés *Egyedi értékek* tulajdonságát *Igen*-re állítjuk.

### 16. feladat: 16\_MelyikMikor

A jelentést varázslóval készítjük el. Felvesszük az előző feladat lekérdezésének mezőit. Előírjuk a *Képesítés*, azon belül pedig a *Vizsga* mező szerinti csoportosítást. Megjelenítjük a *Vizsga* szakasz láblécét. Elhelyezünk benne egy beviteli mezőt a következő kifejezéssel:

```
= "Összesen: " & Count([Dátum]) & " vizsganap"
```

### 17. feladat: 17\_RészvizsgákSzáma

Az összesítő lekérdezéshez hozzáadjuk a *Vizsga* és *Vizsgafajta* táblát. A tervezőrácsba felvesszük a *Vizsgafajta* tábla *Képesítés* (*Group By*), *Vizsga* (*Group By*) és *Fajtakód* (*Group By*) mezőjét (ebben a sorrendben), illetve a *Vizsga* tábla *Dátum* (*Count*) mezőjét. A *Képesítés*, továbbá a *Fajtakód* mezőre rendezést írunk elő.

### 18. feladat: 18\_SikertelenVizsgák

A lekérdezéshez hozzáadjuk a *Dolgozó*, *Vizsga* és *Vizsgafajta* táblát. A tervezőrácsba felvesszük a *Vizsgafajta* tábla *Képesítés* és *Vizsga* mezőjét, a *Dolgozó* tábla *Név* mezőjét (ilyen sorrendben), illetve a *Vizsga* tábla *Eredmény* és *Sikeres* mezőjét. Ez utóbbira megadjuk a *Nem* feltételt. A *Képesítés* és a *Vizsga* mezőre növekvő rendezést írunk elő.

### 19. feladat: 19\_SikertelenVizsgákJelentés

Először elkészítjük az előző feladat lekérdezésének másolatát, melyhez hozzáadjuk a *Dolgozó* tábla *Részleg* és *Beosztás* mezőjét is (**19\_SikertelenSegédlekérdezés**). A jelentést varázslóval hozzuk létre. Felvesszük a segédlekérdezés *Részleg*, *Beosztás*, *Név*, illetve *Eredmény* mezőjét. Előírjuk a *Részleg* és a *Beosztás* szerinti csoportosítást, majd az *Eredmény* szerinti csökkenő rendezést. A feliratok módosítását *Tervező* nézetben végezhetjük el.

### 20. feladat

A lekérdezéshez a 13. feladat megoldását fogjuk felhasználni. Készítsük el a lekérdezés másolatát is, csak az *Igen* feltételt írjuk át *Nem*-re (**20\_SikertelenÖsszes**). A feladat megoldását szolgáló lekérdezésbe (**20\_SikeresÉsSikertelen**) vegyük fel mindkét lekérdezést és a *Vizsgafajta* táblát. A *Vizsgafajta* táblát kapcsoljuk hozzá mindkét lekérdezéshez a *Képesítés*, illetve a *Vizsga* mező segítségével. A kapcsolatok illesztési tulajdonságát állítsuk be úgy, hogy a *Vizsgafajta* tábla minden rekordja szerepeljen az eredménytáblában (2. eset, laza illesztés). A tervezőrácsba adjuk hozzá a *Vizsgafajta* tábla *Képesítés* és *Vizsga* mezőjét, illetve mindkét lekérdezésből a *CountOfNév* mezőt.

### 21. feladat: 21\_SikeresÉsSikertelenDiagram

A feladatot a *Diagram* varázslóval oldjuk meg. Felvesszük a *Vizsga* és a két *CountOfNév* mezőt. Oszlopdiagramot választunk. A vízszintes tengelyen lévő dobozban elhelyezzük a *Vizsgafajta.Vizsga*, a függőleges tengelyhez pedig a két *CountOfNév* mezőt. A diagram elrendezését, feliratait a *Tervező* nézetben módosíthatjuk.

### 22. feladat: 22\_RészvizsgákSzáma

Az összesítő lekérdezéshez hozzáadjuk mindhárom táblát. A tervezőrácsba felvesszük a *Dolgozó* tábla *Név* (*Group By*) mezőjét, illetve a *Vizsgafajta* tábla *Képesítés* (*Group By*) és *FajtaKód* (*Count*) mezőjét. A *Név* mezőre növekvő rendezést írunk elő.

**Megjegyzés:** ebben a feladatban az összes vizsgát figyelembe vettük, függetlenül attól, hogy sikeres volt-e. A sikeres vizsgák számlálását a 27. feladat megoldása végzi el.

### 23. feladat: 23\_DolgozóVizsgái

A lekérdezéshez hozzáadjuk a *Dolgozó* tábla *Név* mezőjét, a *Vizsgafajta* tábla *Képesítés* és *Vizsga* mezőjét, továbbá a *Vizsga* tábla *Sikeres*, illetve *Dátum* mezőjét. Ez utóbbira előírjuk a növekvő rendezést. A *Név* mezőhöz beírjuk a [Név?], a *Sikeres* mezőhöz pedig az Igen feltételt.

### 24. feladat: 24\_Statisztika

Az összesítő lekérdezéshez hozzáadjuk a *Dolgozó* tábla *Név* (*Group By*) mezőjét, a *Vizsgafajta* tábla *Képesítés* (*Group By*) mezőjét, továbbá háromszor a *Vizsga* tábla *Eredmény* (*Avg*, *Min*, *Max*) mezőjét. A *Név* mezőhöz beírjuk a [Név?] feltételt.

### 25. feladat: 25\_VizsgaStatisztika

A jelentést varázslóval készítjük el. Felvesszük a *Vizsgafajta* tábla *Képesítés* és *Vizsga* mezőjét, illetve a *Vizsga* tábla *Eredmény* mezőjét. Előírjuk a *Képesítés* és *Vizsga* mezők szerinti csoportosítást. Az *Összesítési beállításoknál* bejelöljük az átlagot (*Átl*), a minimumot (*Min*) és a maximumot (*Max*). A *Megjelenítés* során *Csak összesítést* kérünk. A mezők feliratát a *Tervező* nézetben módosítjuk. Az összegzésre vonatkozó mezők helyett készítünk egy új beviteli mezőt a `=Count([Eredmény])`

kifejezéssel.

### 26. feladat: 26\_RészvizsgákSzáma

Az összesítő lekérdezésbe felvesszük a *Vizsgafajta* tábla *Képesítés* (*Group By*) és *Vizsga* (*Count*) mezőjét.

### 27. feladat: 27\_SikeresRészvizsgák

A 22. feladat megoldását alakítjuk át. Felvesszük a tervezőrácsba a *Vizsga* tábla *Sikeres* (*Where*) mezőjét, és előírjuk rá az Igen feltételt.

**Megjegyzés:** a megoldás során feltételeztük, hogy sikeres vizsgát senki nem ismételt meg. Így egy részvizsgából egy dolgozónak legfeljebb egy sikeres vizsgája lehet.

### 28. feladat: 28\_MindenVizsgátLetett

A lekérdezéshez hozzáadjuk a 26. és 27. feladatok lekérdezéseit. A két táblát a *Képesítés* mező segítségével összekapcsoljuk. Összekapcsoljuk a *CountOfFajtakód* és a *CountOfVizsga* mezőt is. A tervezőrácsba felvesszük a 27. feladat *Név*, *Képesítés*, illetve *CountOfFajtakód* mezőjét.

### 29. feladat: 29\_Oklevél

Hozzunk létre egy új jelentést *Tervező* nézetben (*Beszúrás/Jelentés*). Az *Új jelentés* ablakban jelöljük ki az előző feladat lekérdezését. A jelentésablak mellett megjelenő mezőlistából helyezzük át a *Név* és a *Képesítés* mezőt a jelentéstörzsbe (ezt beviteli mezők elhelyezésével és kötésével is megtehetjük). Lássuk el a jelentéstörzset feliratokkal. A jelentéstörzs aljára tegyünk az *Eszközkészletből* oldaltörést. Az aktuális dátum és képek beillesztésével is díszíthetjük az oklevelet.

### 30. feladat

Először segédlekérdezésekben megszámoljuk a sikertelen, illetve az ismételt vizsgákat. A

**30\_SikertelenSzámlál** összesítő lekérdezésbe vegyük fel a *Dolgozó* tábla *Név* (*Group By*), a *Vizsgafajta* tábla *Képesítés* (*Group By*) és *Vizsga* (*Group By*), illetve a *Vizsga* tábla *Vizsgakód* (*Count*) és *Sikeres* (*Where*) mezőjét. Ez utóbbira írjuk elő a Nem feltételt.

A **30\_TöbbszöriSzámlál** összesítő lekérdezésbe vegyük fel a *Dolgozó* tábla *Név* (*Group By*), a *Vizsgafajta* tábla *Képesítés* (*Group By*) és *Vizsga* (*Group By*), illetve a *Vizsga* tábla *Vizsgakód* (*Count*) mezőjét. Ez utóbbira írjuk elő a >1 feltételt.

A feladat megoldását szolgáltató lekérdezéshez (**30\_TöbbszöriPróbálkozások**) adjuk hozzá a két segédlekérdezést. Kapcsoljuk őket össze a *Név*, *Képesítés* és *Vizsga* mezők segítségével. A tervezőrácsba vegyük fel ezt a három mezőt, és mindkét táblából a *CountOfVizsgakód* mezőt.

### 31. feladat

Először segédlekérdezést készítünk (**31\_MindigSikertelen**). Létrehozzuk az előző feladat megoldásának másolatát. Összekapcsoljuk a *CountOfVizsgakód* mezőket is.

A másolatból töröljük a *30\_\_SikertelenSzámlál.CountOfVizsgakód* mezőt. A jelentést ezen segédlekérdezés alapján állítjuk elő (**31\_\_MindigSikertelenJelentés**).

A többszöri sikeres részvizsgák listázásához segédlekérdezést készítünk (**31\_\_TöbbszöriSikeres**). Elkészítjük a 27. feladat lekérdezésének másolatát, majd kiegészítjük a *Vizsgafajta* tábla *Vizsga (Group By)* mezőjével. A *FajtaKód* mezőre előírjuk a >1 feltételt. A jelentést ezen lekérdezés alapján hozzuk létre (**31\_\_TöbbszöriSikeresJelentés**).

**Megjegyzés:** a példa-adatbázisban nem szerepel ez utóbbi lekérdezés feltételének megfelelő rekord.

## Mozi

Importáljuk egy új adatbázisba a táblákat. Ellenőrizzük, és szükség esetén módosítsuk a mezők típusát. A *Szám* típusú mezők méretét *Hosszú egésze*re változtassuk. Jelöljük ki az elsődleges kulcsokat. Hozzuk létre a feladatgyűjteményben szereplő kapcsolatokat. A kapcsolatoknál jelöljük be a hivatkozási integritás megőrzését.

### 1. feladat: 01\_MoziLista

Vegyük fel a lekérdezésbe a *Mozi* tábla *NÉV* mezőjét, és írjunk elő rendezést.

### 2. feladat: 02\_MagyarFilm

Vegyük fel a lekérdezésbe a *Film* tábla *CÍM* és *SZÁRMAZÁS* mezőjét. Ez utóbbira írjuk elő a

Like `"*magyar"`

feltételt.

### 3. feladat: 03\_MagyarMozi

Vegyük fel a lekérdezésbe a *Mozi*, *Vetíti* és *Film* táblákat. Adjuk hozzá a tervezőracshoz a *Mozi.NÉV*, illetve a *Film.SZÁRMAZÁS* mezőt. Ez utóbbira írjuk elő a

Like `"*magyar"`

feltételt, és tiltsuk le a megjelenítését. A lekérdezés *Egyedi értékek* tulajdonságát állítsuk *Igen*-re.

**Megjegyzés:** a *Mozi* és a *Film* táblák közti kapcsolatot biztosító *Vetíti* táblát akkor is hozzá kell adnunk a lekérdezéshez, ha egyetlen rekordját sem használjuk fel!

### 4. feladat: 04\_MoziMűsor

Vegyük fel a lekérdezésbe a *Film*, *Vetíti* és *Mozi* táblákat. Adjuk hozzá a tervezőracshoz a *Film.Cím*, illetve a *Mozi.NÉV* mezőt. A *CÍM* mezőre írunk elő rendezést.

### 5. feladat: 05\_AmerikaiDB

Vegyük fel az összesítő lekérdezésbe a *Film* tábla *FKÓD (Count)* és *SZÁRMAZÁS (Where)* mezőjét. Ez utóbbira írjuk be a

Like `"*amerikai"`

feltételt.

### 6. feladat: 06\_FilmIdőrend

A lekérdezésbe vegyük fel a *Film* tábla *CÍM* és *GYÁRTÁSI\_ÉV* mezőjét. Ez utóbbira írunk elő növekvő rendezést.

### 7. feladat: 07\_MozikSzáma

A feladatgyűjtemény szövegéből kimaradt egy pont a 13 után. Helyesen: „A 13. kerületben hány Art mozi van?”

Az összesítő lekérdezésbe vegyük fel a *Mozi* tábla *MKÓD (Count)*, *KERÜLET (Where)* és *TÍPUS (Where)* mezőjét. A *KERÜLET*-re írjuk elő az =13, a *TÍPUSRA* pedig az "art" feltételt. Próbáljuk ki a lekérdezést a 7. kerülettel is.

### 8. feladat: 08\_Táncos

A lekérdezésbe vegyük fel a *Film* tábla *CÍM* és *HOSSZ* mezőjét. A *CÍM* mezőre írjuk elő a "Táncos a sötétben" feltételt.

### 9. feladat: 09\_Szereplők

Vegyük fel a lekérdezésbe a *Film*, a *Szerepel* és az *Ember* táblát. Töröljük a kapcsolatot a *Film* és az *Ember* tábla között. Adjuk hozzá a tervezőrácsához a *Film.CÍM*, illetve az *Ember.NÉV* mezőt. A *CÍM* mezőre írjuk elő a "Moszkva tér" feltételt. A mező megjelenítését ki is kapcsolhatjuk.

### 10. feladat: 10\_MagyarSzínész

Vegyük fel a lekérdezésbe a *Film*, a *Szerepel* és az *Ember* táblát. Töröljük a kapcsolatot a *Film* és az *Ember* tábla között. Adjuk hozzá a tervezőrácsához az *Ember.NÉV*, illetve a *Film.SZÁRMAZÁS* mezőt. A *SZÁRMAZÁS* mezőre írjuk elő a

Like "\*magyar\*"

feltételt. A mező megjelenítését ki is kapcsolhatjuk. A lekérdezés *Egyedi értékek* tulajdonságát állítsuk *Igen*-re.

### 11. feladat: 11\_NRendező

Vegyük fel a lekérdezésbe az *Ember* és a *Név* táblát. Adjuk hozzá a tervezőrácsához az *Ember.NÉV*, illetve a *Film.CÍM* mezőt. Ez utóbbira írjuk elő a "Nexxt" feltételt. A mező megjelenítését ki is kapcsolhatjuk.

### 12. feladat: 12\_MitNézzünk

Adjuk hozzá a lekérdezéshez a *Film*, a *Vetíti* és a *Mozi* táblát. Vegyük fel a tervezőrácsba a *Film.CÍM*, *Film.MŰFAJ*, illetve a *Mozi.TÍPUS* mezőt. Írjuk elő a *MŰFAJ* mezőnél a "vígjáték", a *TÍPUS* mezőnél pedig a "<>"multi" feltételt. E két mező megjelenítését le is tilthatjuk. A lekérdezés *Egyedi értékek* tulajdonságát állítsuk *Igen*-re.

### 13. feladat: 13\_FilmekSzáma

Az összesítő lekérdezéshez adjuk hozzá a *Mozi* és a *Vetíti* táblát. A tervezőrácsba vegyük fel a *Mozi.NÉV* (*Group By*), illetve a *Vetíti.FKÓD* (*Count*) mezőt.

### 14. feladat: 14\_NemAmerikai

Először elkészítjük azoknak a moziknak a listáját, melyek játszanak amerikai filmet. Adjuk hozzá a **14\_\_Amerikai** segédlekérdezéshez a *Mozi*, *Vetíti* és *Film* táblát. Vegyük fel a tervezőrácsba a *Mozi.Név* és *Film.SZÁRMAZÁS* mezőt. Ez utóbbira írjuk elő a Like "\*amerikai" feltételt, és kapcsoljuk ki a megjelenítését. Állítsuk a lekérdezés *Egyedi értékek* tulajdonságát *Igen*-re. Mentsük a lekérdezést.

A **14\_NemAmerikai** lekérdezéshez adjuk hozzá a *Mozi* táblát, a tervezőrácsba vegyük fel a *NÉV* mezőt. A *Feltétel* sorába írjuk be a következő SQL-parancsot:

```
Not In (select * from 14__Amerikai)
```

SQL parancs alkalmazása helyett felhasználhatjuk az Access nem egyezőket kereső varázslóját is. A lekérdezés-objektumok Új menüjéből válasszuk ki a varázslót (*Nem egyezőket kereső lekérdezés varázsló*). Az első ablakban jelöljük ki a *Mozi* táblát, a következő ablakban pedig a **14\_\_Amerikai** segédlekérdezést. Továbbhaladva kattintsunk a <=> gombra (az Access felismerte a mindkét táblában szereplő *Név* mezők azonosságát), majd válasszuk ki a *NÉV* mező megjelenítését. A varázsló befejezésével elkészül az új lekérdezés (**14\_NemAmerikai2**).

**Megjegyzés:** Vizsgáljuk meg ez utóbbi lekérdezés tervező-nézetét. Bemutatja, hogyan hozhattuk volna létre a lekérdezést a varázsló nélkül. A lekérdezéshez hozzá kell adni a *Mozi* táblát és a **14\_\_Amerikai** lekérdezést. A két táblát a *NÉV* mező segítségével össze kapcsoljuk, majd a kapcsolat illesztési tulajdonságainál bejelöljük a bal oldali szoros illesztést (2. pont). Végül a tervezőtáblában a segédlekérdezés *NÉV* mezőjére előírjuk az *Is Null* feltételt.

### 15. feladat: 15\_KétMozi

Az összesítő lekérdezésbe felvesszük a *Film* tábla *CÍM* (*Group By*) és a *Vetíti* tábla *MKÓD* (*Count*) mezőjét. Ez utóbbira előírjuk a >2 feltételt.

### 16. feladat: 16\_VetítésekSzáma

Az előző feladat megoldásából töröljük az *MKÓD* mezőre vonatkozó feltételt.

### 17. feladat: 17\_MűfajDB

Az összesítő lekérdezésbe felvesszük a *Film* tábla *MŰFAJ* (*Group By*) és *FKÓD* (*Count*) mezőjét.

### 18. feladat: 18\_FilmdrámaDB

Az összesítő lekérdezésbe felvesszük a *Film* tábla *SZÁRMAZÁS* (*Group By*) és *MŰFAJ* (*Where*) mezőjét. Ez utóbbira előírjuk a "filmdráma" feltételt.

### 19. feladat: 19\_Leghosszabb

A lekérdezésbe felvesszük a *Film* tábla *CÍM* és *HOSSZ* mezőjét. Ez utóbbira csökkenő rendezést írunk elő. A lekérdezés *Csúcsérték* tulajdonságát 1-re állítjuk.

### 20. feladat: 20\_TöbbRendező

Az összesítő lekérdezéshez hozzáadjuk az *Ember* és a *Film* táblát. A tervezőrácsba felvesszük az *Ember* tábla *NÉV* (*Group By*), illetve a *Film* tábla *FKÓD* (*Count*) mezőjét. Ez utóbbira előírjuk a >1 feltételt.

### 21. feladat: 21\_MaxSzínész

Az összesítő lekérdezésbe felvesszük az *Ember*, a *Szerepel*, továbbá a *Film* táblát. Töröljük az *Ember* és a *Film* tábla közötti kapcsolatot. A tervezőrácsba hozzáadjuk az *Ember* tábla *NÉV* (*Group By*), illetve a *Film* tábla *FKÓD* (*Count*) mezőjét. Ez utóbbira csökkenő rendezést írunk elő. A lekérdezés *Csúcsérték* tulajdonságát 1-re állítjuk.

### 22. feladat: 22\_SzínészRendező

A lekérdezésbe felvesszük a *Szerepel*, *Ember* és *Film* táblát. A tervezőrácsba hozzáadjuk a *Film* tábla *CÍM* mezőjét.

### 23. feladat: 23\_Legrégebbi

A lekérdezésbe felvesszük a *Film* tábla *CÍM* és *GYÁRTÁSI\_ÉV* mezőjét. Ez utóbbira növekvő rendezést írunk elő. A lekérdezés *Csúcsérték* tulajdonságát 1-re állítjuk.

### 24. feladat: 24\_Film

Oszlopos autoűrlapot készítünk a *Film* tábla alapján.

**Megjegyzés:** az adatbevitelt segíti, ha az *FKÓD* mező *Számláló* típusú. Célszerű az űrlapon a műfaj és rendező mezőt kombipanellel megjeleníteni (lásd a Kosárlabda 19. feladatának megoldását).

### 25. feladat: 25\_FilmIdőrend

A 6. feladat lekérdezése alapján oszlopos autojelentést hozunk létre, melyet utólag formázhatunk.

### 26. feladat: 26\_Moziműsor

A *04\_Moziműsor* lekérdezés alapján oszlopos autojelentést hozunk létre, melyet utólag formázhatunk.

### 27. feladat: 27\_Moszkvatér

A 9. feladat lekérdezése alapján oszlopos autojelentést hozunk létre, melyet utólag formázhatunk. A rendezést a *Nézet* menü *Rendezés és csoportosítás* parancsával állítjuk be.

## Adatok a Földről – 1.

### 1. feladat: Föld tábla

Importáljuk egy új adatbázisba a *cia\_adat.txt* fájl megadott mezőit (felhasználhatjuk a feladatgyűjtemény CD-jén lévő *mdb* fájlt is). Az első sor tartalmazza a mezőneveket. Kulcsnak jelöljük ki az *Azonosító* mezőt. A női és férfi népességre vonatkozó mezők méretét változtassuk *Hosszú egészre*. Az adatbázis neve: **Föld1**.

### 2. feladat: 02\_Lakosság

A tábla *Tervező* nézetében felvesszünk egy új, *Szám* (*Hosszú egész*) típusú mezőt (*Lakosság*). Egy új lekérdezéshez hozzáadjuk a tábla *Lakosság* mezőjét, majd átalakítjuk *Frissítő* lekérdezéssé (*Lekérdezés* menü).

A *Módosítás* sorba beírjuk a

[0-14\_no] + [15-64\_no] + [65-->\_no] + [0-14\_fffi] + [15-64\_fffi] + [65-->\_fffi]

kifejezést. A bonyolult mezőnevek miatt célszerű a *Kifejezésszerkesztőt* használni. A módosítást a lekérdezés futtatásával hajtjuk végre.

**Figyelem!** A rekordok módosítására (az összlakosság kiszámítására) a lekérdezés minden egyes megnyitásakor sor kerül!

**Megjegyzés:** a törzsadatokat tartalmazó táblában nem célszerű a többi mezőből kiszámítható értékeket tárolni.

### 3. feladat: 03\_Népsűrűség

A feladatot az előző feladat mintájára oldjuk meg. A *Népsűrűség* mező mérete: *Dupla*. A *Módosítás* sor kifejezése:

[Lakosság]/[fold]

### 4. feladat: 04\_NagyOrszágok

A lekérdezésbe vegyük fel az *ország* és *osszter* mezőt. Ez utóbbira írjunk elő csökkenő rendezést. A lekérdezés *Csúcsérték* tulajdonságát állítsuk 25-re.

### 5. feladat: 05\_KisOrszágok

Az előző lekérdezést módosítjuk úgy, hogy növekvő rendezést írunk elő, 40-es csúcsértékkel.

### 6. feladat: 06\_NagyNépsűrűség

A lekérdezésbe vegyük fel az *ország*, a *fovaros* és a *Népsűrűség* mezőt. Ez utóbbira írunk elő csökkenő rendezést. A lekérdezés *Csúcsérték* tulajdonságát állítsuk 10-re.

### 7. feladat: 07\_Kontinensek

Az összesítő lekérdezésbe felvesszük a *foldresz* (*Group By*), *fold* (*Sum*), *Lakosság* (*Sum*) és *ország* (*Count*) mezőt.

### 8. feladat: 08\_Országok

Az összesítő lekérdezésbe felvesszük az *osszter* (*Sum*) és *viz* (*Sum*) mezőt. Egy-egy újabb mezőbe (*Expression*) beírjuk a

Kif1: [SumOfosszter]/510072000

illetve a

Kif2: [SumOfviz]/361132000

kifejezést. E két mezőre *Százalék* formátumot állítunk be.

### 9. feladat: 09\_Eurázsia

A lekérdezésbe felvesszük a *foldresz*, *ország*, *fovaros* és *osszter* mezőt. Ez utóbbira csökkenő rendezést állítunk be. A *foldresz* mezőre beírjuk a

"Europe" Or "Asia"

feltételt. Szükség esetén letiltjuk a *foldresz* és az *osszter* mező megjelenítését.

**Megjegyzés:** a két feltételt a „vagy” művelet alkalmazása helyett egymás alá is írhatjuk a Tervezőrácsban.

### 10. feladat: 10\_EurázsiaÖsszesen

Az összesítő lekérdezésbe felvesszük a *fold* (*Sum*), *Lakosság* (*Sum*) és *foldresz* (*Where*) mezőt. Ez utóbbira előírjuk a

"Europe" Or "Asia"

feltételt.



### 11. feladat: 11\_MagyarországhozHasonló

A lekérdezésbe felvesszük az *ország* és *osszter* mezőt. Készítünk egy új mezőt a következő kifejezéssel:

```
Kif1: Abs([osszter] - 93030)
```

A mezőre növekvő rendezést írunk elő. A lekérdezés *Csúcsérték* tulajdonságát 21-re állítjuk.

### 12. feladat: 12\_Tízszernagyobb

Az összesítő lekérdezésbe felvesszük a *foldresz* (*Group By*), *ország* (*Count*) és *osszter* (*Where*) mezőt. Ez utóbbira előírjuk a  $>930300$  feltételt. (Lásd még a következő feladat megoldását.)

### 13. feladat

A feladat megoldásánál Magyarország területét is az adattáblából olvassuk ki. Ehhez készítünk egy segédlekérdezést (**13\_Magyarország**), amelybe felvesszük az *ország* és *osszter* mezőt. Az *ország* mezőre előírjuk a "Hungary" feltételt.

A feladat megoldását jelentő összesítő lekérdezéshez (**13\_Tízszernagyobb**) hozzáadjuk a *Föld* táblát és a segédlekérdezést. A tervezőrácsba felvesszük a *Föld* tábla *foldresz* (*Group By*), *ország* (*Count*) és *osszter* (*Where*) mezőit. Ez utóbbira előírjuk a

```
<0,1*[13_Magyarország]![osszter]
```

feltételt.

### 14. feladat: 14\_NincsTerület

A lekérdezésbe felvesszük az *ország*, *fold*, illetve *viz* mezőt. A *fold* és *viz* mezőre a tervezőrács két külön sorában („vagy” művelet!) előírjuk a

```
0 Or Is Null
```

feltételt.

### 15. feladat: 15\_KözelTízmillió

A lekérdezésbe felvesszük az *ország* és *Lakosság* mezőt. Ez utóbbira előírjuk az *Is Not Null* feltételt. Készítünk egy új mezőt a következő kifejezéssel:

```
Kif1: Abs([Lakosság] - 10000000)
```

A mezőre növekvő rendezést írunk elő. A lekérdezés *Csúcsérték* tulajdonságát 10-re állítjuk.

### 16. feladat

A feladat megoldásánál Magyarország népsűrűségét is az adattáblából olvassuk ki. Ehhez készítünk egy segédlekérdezést (**16\_MagyarNépsűrűség**), amelybe felvesszük az *ország* és *Népsűrűség* mezőt. Az *ország* mezőre előírjuk a "Hungary" feltételt.

A feladat megoldását szolgáltató lekérdezéshez (**16\_SűrűségKözelMagyar**) hozzáadjuk a segédlekérdezést és *Föld* táblát. A tervezőrácsba felvesszük a *Föld* tábla *ország* és *Népsűrűség* mezőjét. Készítünk egy új mezőt a következő kifejezéssel:

```
Kif1: Abs([Föld]![Népsűrűség] - [16_MagyarNépsűrűség]![Népsűrűség])/[16_MagyarNépsűrűség]![Népsűrűség]
```

A mezőre előírjuk a  $<0,1$  feltételt.

Az eltérés mezőre százalékos, a *Népsűrűség* mezőre pedig 2 tizedesre történő megjelenítést állítottunk be.

### 17. feladat: 17\_KözelMagyarÖsszesítés

Elkészítjük az előző lekérdezés másolatát. Töröljük a *Népsűrűség*, majd felvesszük a *foldresz* mezőt. A lekérdezést átalakítjuk összesítő lekérdezéssé. Az *Összesítés* sorában a következő záradékokat választjuk:

Mező:	foldresz	ország	Kif1
Összesítés:	Group By	Count	Where

### 18. feladat: 18\_FérfiakNők

A lekérdezésbe felvesszük a *Föld* tábla *ország* mezőjét. Készítünk még két számított mezőt az alábbi kifejezések beírásával:

Kif1: [0-14\_fffi] + [15-64\_fffi] + [65-->\_fffi]

Kif2: [0-14\_no] + [15-64\_no] + [65-->\_no]

### 19. feladat: 19\_FérfiNőArány

Az előző lekérdezést kiegészítjük a következő két számított mezővel:

Kif3: [Kif1]/[Lakosság]

Kif4: [Kif2]/[Lakosság]

A mezőket 3 tizedessel jelenítjük meg.

### 20. feladat: 20\_TöbbFérfi

A lekérdezéshez hozzáadjuk a 18. feladat lekérdezését és a *Föld* táblát. A két táblát az *ország* mező segítségével összekapcsoljuk. A tervezőrácsba felvesszük az *ország* és *fovaros* mezőt. Ez utóbbihoz beírjuk a [Kif1]>[Kif2] feltételt.

**Megjegyzés:** a lekérdezés bezárása és ismételt megnyitása után figyeljük meg, hogyan módosította az Access a tervezőrácsot.

### 21. feladat: 21\_KontinensNőFérfi

Az összesítő lekérdezésbe felvesszük a *Föld* tábla *foldresz* (*Group By*), illetve a női és férfi korosztályok lélekszámát tartalmazó mezőt. Ez utóbbiakra kiválasztjuk a *Sum* függvényt.

### 22. feladat: 22\_KontinensÖsszes

A lekérdezéshez hozzáadjuk a 21. feladat lekérdezését. A tervezőrácsba felvesszük a *foldresz* mezőt. Készítünk még három számított mezőt a következő kifejezésekkel:

Kif1: [SumOf0-14\_no]+[SumOf0-14\_fffi]

Kif2: [SumOf15-64\_no]+[SumOf15-64\_fffi]

Kif3: [SumOf65-->\_no]+[SumOf65-->\_fffi]

A kifejezések összeállításához használjuk a *Kifejezésszerkesztőt*.

### 23. feladat: 23\_FiatalOrszágok

Mivel a korosztályokra vonatkozó százalékos adatok csak 2 tizedesre kerekített értékeket tartalmaznak, a lélekszám alapján újra kiszámítjuk az értéküket. A lekérdezésbe felvesszük a *Föld* tábla *ország* mezőjét. Készítünk egy számított mezőt a következő kifejezéssel:

Kif1: ([0-14\_no]+[0-14\_fffi])/[Lakosság]

A számított mezőre csökkenő rendezést írunk elő. A lekérdezés *Csúcsérték* tulajdonságát 10-re állítjuk.

### 24. feladat: 24\_ÖregOrszágok

Az előző feladat lekérdezését átalakítjuk úgy, hogy a legidősebb korosztályok népességének összegét határozza meg.

### 25. feladat: 25\_RosszArány

A lekérdezésbe felvesszük a *Föld* tábla *ország* mezőjét. Készítünk egy számított mezőt a következő kifejezéssel:

Kif1: [0-14\_%]+[15-64\_%]+[65-->\_%]

A számított mezőre előírjuk a <>1 feltételt.

**Megjegyzés:** bár Tajvannál a százalékos értékek összege 1, mégis szerepel a lekérdezés eredménytáblájában. A bináris számrendszer használata miatt erre az ellentmondásra mindig kell számítanunk, ha lebegőpontos értékeket (törtszámokat) hasonlítunk össze. Ezért az egyenlő (=),

nem egyenlő (<>) feltételek helyett adjunk meg olyan intervallumot, amely már biztosan jelzi a két mennyiség viszonyát. Módosítsuk például a számított mező feltételét a következőképpen:

```
Not Between 0,999 And 1,001
```

Így már nem találunk hibát az eredménytáblában.

## 26. feladat: 26\_HatBetűs

A lekérdezésbe vegyük fel a *Föld* tábla *ország* mezőjét. A mezőre írjuk elő a

```
Len([ország])=6
```

feltételt.

## 27. feladat: 27\_LeghosszabbNév

A lekérdezésbe vegyük fel a *Föld* tábla *ország* mezőjét. Készítsünk egy új, számított mezőt a `Len([ország])` kifejezéssel. Írjunk elő a mezőre csökkenő rendezést. A lekérdezés *Csúcsérték* tulajdonságát állítsuk 1-re.

**Megjegyzés:** mivel az *ország* mező az adatbázisok készítésére vonatkozó elvekkel ellentétben nem csak az *országnevet* tartalmazza, így a lekérdezéssel a leghosszabb karaktersorozatot határoztuk meg.

## 28. feladat: 28\_HétBetűsÁzsiai

A lekérdezésbe vegyük fel a *Föld* tábla *ország*, *fovaros* és *foldresz* mezőjét. A *fovaros* mezőre írjuk elő a

```
Len([fovaros])=7
```

feltételt, a *foldresz* mezőre pedig az "Asia" feltételt.

## 29. feladat: 29\_ÖtBetűsAfrikai

A lekérdezésbe vegyük fel a *Föld* tábla *ország* és *foldresz* mezőjét. Az *ország* mezőre írjuk elő a

```
Len([ország])=5
```

feltételt, a *foldresz* mezőre pedig az "Africa" feltételt.

## 30. feladat: 30\_GBetűsNégyBetűs

Az előző lekérdezés *ország* mezőjére vonatkozó feltételét módosítsuk a következőképpen:

```
Len([ország])=4 And Like "*g*"
```

## 31. feladat: 31\_TBetűsFőváros

A lekérdezésbe vegyük fel a *Föld* tábla *ország* és *fovaros* mezőjét. Ez utóbbira írjuk elő a "t\*" feltételt.

## Adatok a Földről – 2.

### 1. feladat: Föld tábla

Importáljuk egy új adatbázisba a *cia\_adat.txt* fájl megadott mezőit (felhasználhatjuk a feladatgyűjtemény CD-jén lévő *mdb* fájlt is). Az első sor tartalmazza a mezőneveket. Kulcsnak jelöljük ki az *Azonosító* mezőt. Figyeljünk a mezők típusára. Az éveknél *Egész*, a telefonoknál *Hosszú egész*, a többi *Szám* típusú mezőnél pedig *Dupla* mezőméretet válasszunk. Az adatbázis neve: **Föld2**.

### 2. feladat: 02\_Országok

Oszlopos autojelentést hozunk létre a *Föld* tábla alapján. *Tervező* nézetben *Oldaltörés* vezérlőelemet illesztünk be a jelentéstörzs aljára.

### 3. feladat: 03\_FissAdatok

A lekérdezéshez hozzáadjuk az *ország*, *termev* és *energev* mezőt. A két utóbbira előírjuk a  $\geq 2000$  feltételt.

#### 4. feladat: 04\_Bizonytalan

Felhasználjuk, hogy ha egy országról nincs adat, akkor azt sem tudjuk, melyik évből származnak az értékek. Elegendő tehát csak az évekre vonatkozó mezők tartalmát megvizsgálni. A lekérdezéshez hozzáadjuk az *ország*, *termev* és *energev* mezőt. E két utóbbira külön sorban („vagy” kapcsolat!) írjuk elő az `IS NULL` feltételt.

#### 5. feladat: 05\_SokSzolgáltatás

A lekérdezéshez hozzáadjuk az *ország* és *szolgaltat* mezőt. Ez utóbbira előírjuk a `>0,5` feltételt.

#### 6. feladat: 06\_SokSzolgáltatás

Először segédlekérdezést hozunk létre (**06\_SokSzolgáltatás**). Elkészítjük a *05\_SokSzolgáltatás* lekérdezés másolatát, és felvesszük a *foldresz* mezőt is. A jelentéshez varázslót használunk. Előírjuk a földrészenkénti csoportosítást és az országok szerinti rendezést.

#### 7. feladat: 07\_SokSzolgáltatásÖsszes

Az összesítő lekérdezésbe felvesszük a *foldresz* (*Group By*), az *ország* (*Count*) és a *szolgaltat* (*Where*) mezőt. Ez utóbbira előírjuk a `>0,5` feltételt.

**Megjegyzés:** bár a feladat szövege többes számot használ („fenti feltételeknek”), csak az 5. feladat feltételét vettük figyelembe. A 3. és 4. feladat feltételének együttes alkalmazása ugyanis nem egyértelmű.

#### 8. feladat: 08\_Mezőgazdasági

A lekérdezéshez hozzáadjuk az *ország* és *mezogazd* mezőt. Ez utóbbira előírjuk a `>0,9` feltételt. (Ilyen ország nincs az adattáblában.)

#### 9. feladat: 09\_Átlagos

Az összesítő lekérdezésbe felvesszük a *foldresz* (*Group By*), *ipar* (*Avg*), *mezogazd* (*Avg*) és *szolgaltat* (*Avg*) mezőt. Az átlagokat jelző mezőkre százalékos formátumot írunk elő.

#### 10. feladat: 10\_CsakFosszilis

A lekérdezésbe felvesszük az *ország* és *fosszilis* mezőt. Ez utóbbira előírjuk az `=1` feltételt.

#### 11. feladat: 11\_Fosszilis

Mivel az előző feladatban csak a 100%-osan fosszilis energiát termelő országok szerepeltek, így a feladat szövegében olvasható sorrendnek nincs értelme. Ezért a megoldásban kontinensenként csoportosítva minden országot kilistázunk, a fosszilis energiatermelés szerinti növekvő sorrendben. A jelentést a *Föld* tábla alapján a varázslóval készítjük el. Felvesszük a *foldresz*, *ország* és *fosszilis* mezőt. Előírjuk a földrészenkénti csoportosítást, illetve a *fosszilis* mező szerinti rendezést. A *fosszilis* mezőre kérjük az átlag, a minimum és a maximum megjelenítését is.

#### 12. feladat: 12\_CsakFosszilisÖsszes

Az összesítő lekérdezésbe felvesszük a *foldresz* (*Group By*), *ország* (*Count*) és *fosszilis* (*Where*) mezőt. Ez utóbbira előírjuk az `=1` feltételt.

#### 13. feladat: 13\_FosszilisIpari

Az összesítő lekérdezésbe felvesszük a *foldresz* (*Group By*), *ipar* (*Avg*) és *fosszilis* (*Where*) mezőt. Ez utóbbira előírjuk az `=1` feltételt. Az ipar mezőre százalékos formátumot írunk elő.

#### 14. feladat: 14\_VanNukleáris

A lekérdezésbe felvesszük az *ország* és *nuklearis* mezőt. Ez utóbbira előírjuk a `>0` feltételt és a csökkenő rendezést.

#### 15. feladat: 15\_VanEgyébEnergia

A lekérdezésbe felvesszük a *Föld* tábla *ország* mezőjét. Készítünk két számított mezőt a következő kifejezésekkel:

Kif1: [fosszilis] + [vizi] + [nuklearis]

Kif2: 1 - [Kif1]

A *Kif1* mezőre előírjuk a <1 feltételt. A *Kif2* mezőt 4 tizedessel jelenítjük meg.

## 16. feladat

Először elkészítjük a 15. feladat lekérdezésének másolatát (**16\_VanEgyébKontinensenként**), melybe felvesszük a *foldresz* mezőt is. A jelentést ennek a lekérdezésnek az alapján, varázslóval hozzuk létre (**16\_VanEgyébKontinensenként**). Előírjuk a földrészenkénti csoportosítást és a *Kif2* mező szerinti csökkenő rendezést.

## Adatok a Földről – 3.

### 1. feladat: Föld tábla

Importáljuk egy új adatbázisba a *cia\_adat.txt* fájl megadott mezőit (felhasználhatjuk a feladatgyűjtemény CD-jén lévő *mdb* fájlt is). Az első sor tartalmazza a mezőneveket. A zászlók hosszú leírása miatt a *zaszlo* mező *Feljegyzés* típusú legyen. Nem jelölünk ki elsődleges kulcsot. Az adatbázis neve: **Föld3**.

### 2. feladat: 02\_Red

Sajnos a „red” karaktersorozat sok, a leírásban szereplő egyéb angol szónak lehet része (például centered, colored, bordered stb.). Ezért először célszerű a *Szerkesztés/Keresés* paranccsal rákeresni a red-re (a mező bármely részében), és tájékozódni, hogy milyen szöveggörnyezetben fordulhat elő. A következő eseteket találtuk (□ jelöli a szóközt):

a leírás elején:	red□
a leíráson belül:	□red□
a leírás végén:	□red
felsorolásban:	□red,
kezdő felsorolásban:	red,
mellékmondat végén:	□red;
összetett kifejezés előtagjaként:	□red-

A lekérdezésbe tehát felvesszük az *ország* és a *zaszlo* mezőt, majd ez utóbbi feltételeként a tervezőrácsban egymás alatt megadjuk a következő kifejezéseket (ügyeljünk a szóközre!):  
"red \*", "\* red \*", "\* red", "\* red,\*", "red,\*", "\* red;\*", "\* red-"

**Megjegyzés:** némileg egyszerűsíthetjük az esetek feltárását, ha olyan lekérdezést készítünk, amely megkeresi a leírásban a „red” karaktersorozatot, majd kiírja a balra és jobbra lévő karakterekkel együtt (**02\_Keres**). Sajnos így csak az első előfordulást találjuk meg a zászló leírásában, így kimarad például a „red-” karaktersorozat (Óceánia).

### 3. feladat: 03\_HáromCsík

A lekérdezésbe felvesszük az *ország* és *zaszlo* mezőt. Ez utóbbira előírjuk a

Like `"*three equal horizontal bands"`

feltételt.

### 4. feladat: 04\_PirosFehérZöld

Egy segédlekérdezéssel (lásd a 2. feladat megjegyzését) meggyőződhetünk arról, hogy a „white” és „green” karaktersorozatok csak önálló szóként szerepelnek a zászlók leírásában. Így a 2. feladat lekérdezését kell tovább szűkíteni a

Like `"*white"` And Like `"*green"`

feltétellel.

## Árpád-ház

Importáljuk a táblát egy új adatbázisba. A *Szám* típusú mezők méretét változtassuk *Egészre*, a *Király* mező megjelenítését pedig *Jelölőnégyzetre*. Jelöljük ki a tábla elsődleges kulcsát.

### 1. feladat: 01\_Névsor

A feladat megoldását átgondolva látható, hogy a tábla nem felel meg az adatbázis mezőivel kapcsolatos elvárásnak, mert összetett tulajdonságot is tartalmaz (*Gyermek* mező: a király sorszáma; jelzője, például Vak; neve; a neve másik formában, például Endre – András). Ilyen módon teljesen értelmetlen névsort kialakítani. (Milyen sorrendben álljon Szt. István, Vak Béla és I. László?) Ezért módosítsuk a rekordokat a zárójelben lévő karaktersorozatok törlésével. Szükség esetén ezeket a jelzőket egy újabb mezőben tárolhatjuk.

A lekérdezésbe vegyük fel a *Gyermek*, *Király*, *Mettől* és *Meddig* mezőt. A *Király* mezőre adjuk meg az *Igen* feltételt. A névsor elkészítéséhez válasszuk szét a sorszámot és a keresztnévet. Ezt a következő három kifejezéssel tehetjük meg:

```
Kif1: InStr(1;[Gyermek];" ")
Kif2: Mid([Gyermek];[Kif1]+1)
Kif3: IIf([Kif1]<>0;Left([Gyermek];[Kif1]-1);" ")
```

A lekérdezés *Adatlap* nézetében jelöljük ki a *Kif2* és *Kif3* mezőt, és állítsunk be növekvő rendezést. A kifejezéseket tartalmazó mezők szélességét csökkentjük nullára.

**Megjegyzés:** a névsor kialakítása sokkal egyszerűbb lenne, ha nem használnánk az adatbázisban összetett tulajdonságot, azaz a királyok sorszámát és keresztnévét külön mezőben tárolnánk!

### 2. feladat: 02\_Endre

A lekérdezéshez adjuk hozzá a *Gyermek*, *Mettől* és *Meddig* mezőt. A *Gyermek* mezőre írjuk elő a *Like "Endre"*

feltételt. Készítsünk egy számított mezőt a

```
Kif1: [Meddig] - [Mettől]
```

kifejezés beírásával.

### 3. feladat: 03\_Gyerekek

Az összesítő lekérdezéshez adjuk hozzá a *Szülő* (*Group By*) és *Gyermek* (*Count*) mezőt. Árpád itt nem szereplő apjának kiszűréséhez a *Szülő* mezőhöz írjuk be az *Is Not Null* feltételt.

### 4. feladat: 04\_KirályGyerekek

Az előző feladat lekérdezéséhez vegyük hozzá a *Király* (*Where*) mezőt is. Írjuk be az *Igen* feltételt.

### 5. feladat

Először készítünk egy segédlekérdezést (**05\_Királyok**), amely kilistázza a királyokat a szülőkkel együtt. (Felhasználhatnánk az 1. feladat megoldását is, de bővíteni kéne a *Szülő* mezővel. Ráadásul az adatbázis-kezelő minden egyes futtatásnál fölöslegesen kiszámítaná a bonyolult kifejezés értékét.) A lekérdezésbe felvesszük a *Család* tábla *Szülő*, *Gyermek* és *Király* mezőjét. Ez utóbbira előírjuk az *Igen* feltételt.

A következő segédlekérdezéssel (**05\_KirálynakKirály**) előállítjuk azoknak a királyoknak a listáját, akiknek a szülője is uralkodó volt. Adjuk hozzá a lekérdezéshez kétszer (!) a *05\_Királyok* lekérdezést. A táblákat kössük össze a *Szülő* és *Gyermek* mezők közötti kapcsolat kialakításával. A tervezőrácsba vegyük fel a *Szülő*, illetve a *Gyermek* mezőt.

A feladat megoldását szolgáltató összesítő lekérdezésbe (**05\_KirálynakÖsszesen**) vegyük fel a *05\_KirálynakKirály* lekérdezés *Szülő* (*Group By*) és *Gyermek* (*Count*) mezőjét.

**Megjegyzés:** a király utóddal rendelkező uralkodók listáját előállíthatjuk az *In* operátorban alkalmazott *SELECT* utasítással is (**05\_KirálynakKirály2**). Ekkor elegendő felvenni a *Család*

táblát a *Szülő*, *Gyermek* és *Király* mezőkkel. Ez utóbbira írjuk elő az *Igen* feltételt, a *Szülő* mezőre pedig a

```
In (SELECT Gyermek FROM Család WHERE Király = YES)
```

feltételt. Az összesítő lekérdezést a fentihez hasonló módon hozzuk létre.

#### 6. feladat: 06\_HosszSorrrend

A lekérdezésbe vegyük fel a *Család* táblát. A tervezőrácsához adjuk hozzá a *Gyermek* és a *Király* mezőt. Ez utóbbira írjuk elő az *Igen* feltételt. Készítsünk egy számított mezőt a következő kifejezéssel:

```
Kif1: [Meddig] - [Mettől]
```

A lekérdezést rendezzük a számított mező szerinti csökkenő sorrendben.

#### 7. feladat: 07\_Géza

Az összesítő lekérdezésbe vegyük fel a *Család* tábla *Király* (*Where*), és kétszer a *Gyermek* (*Where*, illetve *Count*) mezőjét. A feltételek:

Mező:	Király	Gyermek
Feltétel:	Igen	Like "*Géza*"

#### 8. feladat: 08\_Vége

Az összesítő lekérdezésbe vegyük fel a *Család* tábla *Meddig* (*Max*) és *Király* (*Where*) mezőjét. Ez utóbbira írjuk elő az *Igen* feltételt.

#### 9. feladat: 09\_BélaApja

A lekérdezésbe vegyük fel a *Család* tábla *Szülő*, illetve *Gyermek* mezőjét. Ez utóbbira írjuk elő a "IV. Béla" feltételt és tiltsuk le a megjelenítését.

#### 10. feladat

Az előző feladat megoldásához hasonlóan egy segédlekérdezéssel (**10\_IstvánApja**) meghatározzuk IV. István apját. A feladat megoldását szolgáltató lekérdezéshez (**10\_IstvánNagyapja**) hozzáadjuk a segédlekérdezést és a *Család* táblát. A *Család* tábla *Gyermek* mezőjét hozzákapsoljuk a segédlekérdezés *Szülő* mezőjéhez. A tervezőrácsba felvesszük a *Család* tábla *Szülő* mezőjét.

#### 11. feladat

Az előző feladatok megoldásához hasonlóan egy segédlekérdezéssel (**11\_LászlóApja**) meghatározzuk II. László apját. A feladat megoldását szolgáltató lekérdezéshez (**11\_LászlóTestvérei**) hozzáadjuk a segédlekérdezést és a *Család* táblát. A két táblát a *Szülő* mező segítségével összekapcsoljuk. A lekérdezésbe felvesszük a *Család* tábla *Gyermek* mezőjét. Előírjuk rá a <>"II. László" feltételt.

#### 12. feladat

Az előző feladatok megoldásához hasonlóan egy segédlekérdezéssel (**12\_VakBélaApja**) meghatározzuk II. László apját, majd nagyapját (**12\_VakBélaNagyapja**). A feladat megoldását szolgáltató lekérdezéshez (**12\_Nagybácsi**) hozzáadjuk a *12\_VakBélaApja*, illetve a *12\_VakBélaNagyapja* lekérdezést és a *Család* táblát. E két utóbbi táblát a *Szülő* mező segítségével összekapcsoljuk. A lekérdezésbe felvesszük a *Család* tábla *Gyermek* mezőjét. Előírjuk rá a

```
<>[12_VakBélaApja]![Szülő]
```

feltételt.

#### 13. feladat: 13\_ÖtÉv

A lekérdezésbe felvesszük a *Család* tábla *Gyermek* mezőjét. Készítünk egy számított mezőt a

```
Kif1: [Meddig] - [Mettől]
```

kifejezés alkalmazásával. A mezőre előírjuk a  $<5$  feltételt. A rendezést az 1. feladatnak megfelelő módon végezzük.

#### 14. feladat: 14\_KirályokSzama

Az összesítő lekérdezéshez hozzáadjuk a *Család* tábla *Gyermek* (Count), *Király* (Where), *Mettől* (Where) és *Meddig* (Where) mezőjét az alábbi feltételekkel:

Mező:	Király	Mettől	Meddig
Feltétel:	Igen	$\leq 1099$	$\geq 1000$

**Megjegyzés:** a lekérdezést a feladat szövegének megfelelően készítettük el, bár a XI. század 1001-től 1100-ig tartott. Lásd még az Uralkodók 7. feladatának megoldását!

#### 15. feladat

Az al-lekérdezésben (**15\_GézaMeddig**) meghatározzuk, hogy meddig uralkodott II. Géza. Felvesszük a *Gyermek* és *Meddig* mezőt. A *Gyermek* mezőre előírjuk a "II. Géza" feltételt. A feladat megoldását szolgáltató összesítő lekérdezéshez (**15\_GézaUtániak**) hozzáadjuk a *Család* táblát és a segédlekérdezést. Töröljük a köztük lévő kapcsolatot. A tervezőrácsba felvesszük a *Család* tábla *Gyermek* (Count), illetve *Mettől* (Where) mezőjét. Ez utóbbira előírjuk a

```
>=[15_GézaMeddig]![Meddig]
```

feltételt.

#### 16. feladat: 16\_MaxKirály

A 6. feladat megoldásának *Csúcsérték* tulajdonságát 1-re állítjuk.

#### 17. feladat: 17\_Adatfelvitel

Oszlopos autoűrlapot készítünk a *Család* tábla alapján.

#### 18. feladat: 18\_ÖtÉvJelentés

Táblázatos autojelentést készítünk a 13. feladat lekérdezése alapján. A *Tervező* nézetben töröljük a kifejezéseket tartalmazó mezőket, és módosítjuk a *Gyermek* mező feliratát.

#### 19. feladat: 19\_HosszSorrendJelentés

Táblázatos autojelentést készítünk a 6. feladat lekérdezése alapján. A *Tervező* nézetben töröljük a *Király* mezőt, és módosítjuk a *Gyermek* mező feliratát.

#### 20. feladat: 20\_GyerekekJelentés

Táblázatos autojelentést készítünk a 3. feladat lekérdezése alapján.