

Adatbázisok az iskolában 2012

Adatmodellezés

Dr. Balázs Péter

Adatmodell

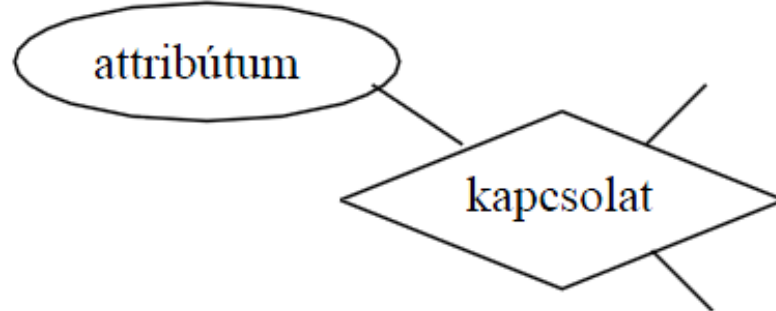
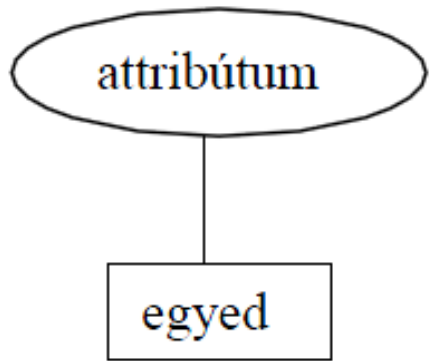
- Nem a konkrét adatokkal, azok előfordulásával, hanem azok típusaival illetve a köztük lévő kapcsolatokkal foglalkozik.
- Egy adatbázis-kezelő rendszer mindig egy adatmodellre épül:
 - Hierarchikus modell
 - Hálós modell
 - Relációs modell
 - Objektorientált modell
 - Objektum-relációs modell

EK-diagramm

- Grafikus leíró eszköz, diagramm segítségével szemléletesen adja meg az adatbázis struktúráját
- Magas szintű, logikai adatmodell, mely egyedtípusokból, köztük lévő kapcsolatokból és az egyes egyedtípusokhoz tartozó attribútumokból épül fel
- „Dia” rajzolóprogram

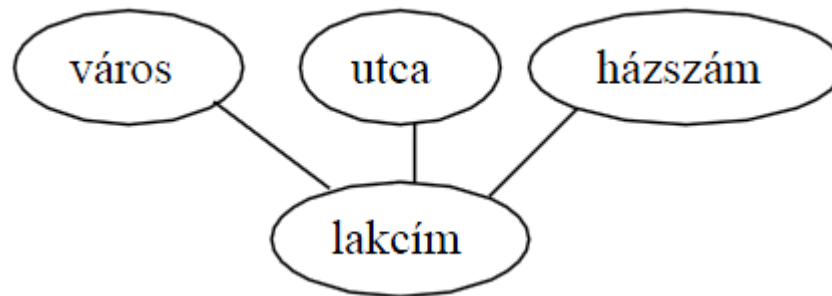
Az EK-diagramm elemei

- a tulajdonságokat (attribútumokat) úgy választjuk meg, hogy azok egyértelműen meghatározzák az egyedeket
- kapcsolatnak is lehetnek attribútumai

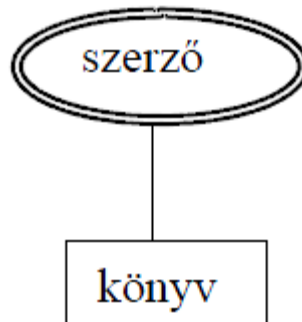


Összetett és többértékű attribútumok

- Összetett attribútum: több elemi attribútumból áll



- Többértékű attribútum: halmaz vagy lista az értéke

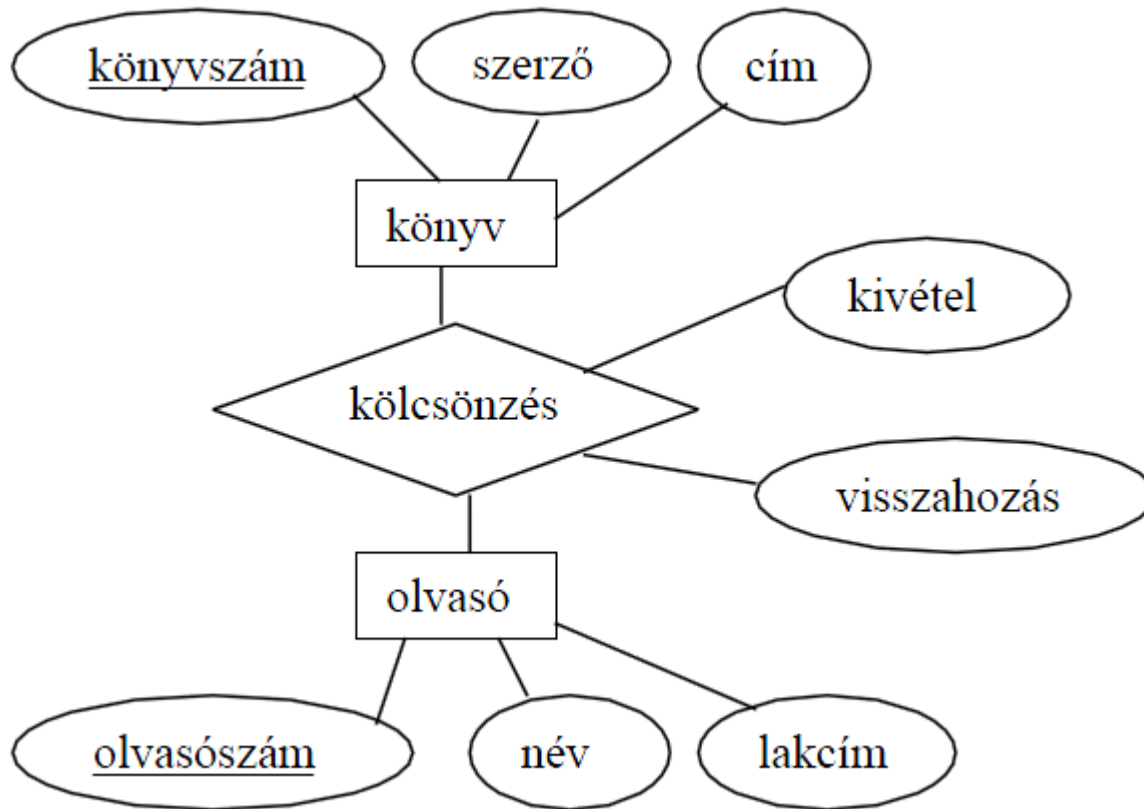


Kulcs

- Egy vagy több attribútum együtt, amely egyértelműen meghatározza az egyedet
- Halmaztartalmazásra minimális, azaz elhagyva valamit is belőle, a megmaradt attribútumok már nem azonosítanak egyértelműen
- A diagramon a kulcsot a megfelelő attribútumok aláhúzásával jelöljük

Példa

- Egy könyvtár könyveit és olvasóit modellezzük, nyilvántartást vezetünk a kölcsönzési és a visszahozási időpontokról is (az olvasókat és a könyveket egy általunk bevezetett könyvszám illetve olvasószám attribútummal azonosítjuk egyértelműen, azaz ezek lesznek a kulcsok)

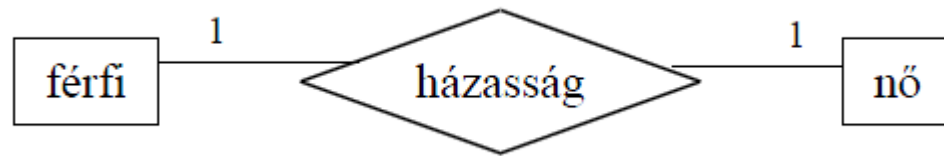


Kapcsolatok típusai

- Kettőnél több egyed közötti (nem jelöljük)
- Két egyed közötti
 - **1:1** kapcsolat

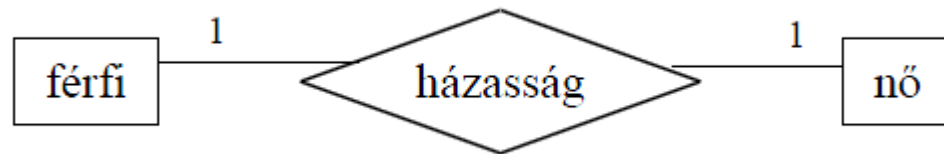
Kapcsolatok típusai

- Kettőnél több egyed közötti (nem jelöljük)
- Két egyed közötti
 - **1:1** kapcsolat (például férj-feleség kapcsolat Európában)



Kapcsolatok típusai

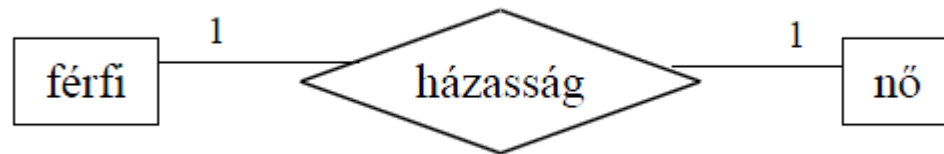
- Kettőnél több egyed közötti (nem jelöljük)
- Két egyed közötti
 - **1:1** kapcsolat (például férj-feleség kapcsolat Európában)



- **1:N** kapcsolat

Kapcsolatok típusai

- Kettőnél több egyed közötti (nem jelöljük)
- Két egyed közötti
 - **1:1** kapcsolat (például férj-feleség kapcsolat Európában)

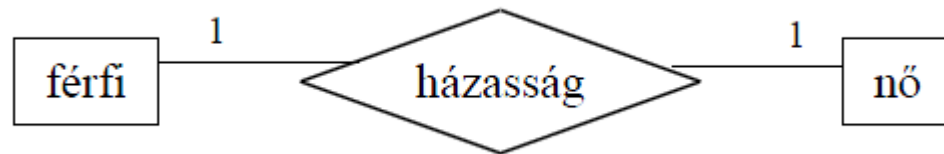


- **1:N** kapcsolat (például anya-gyerek kapcsolat)



Kapcsolatok típusai

- Kettőnél több egyed közötti (nem jelöljük)
- Két egyed közötti
 - **1:1** kapcsolat (például férj-feleség kapcsolat Európában)



- **1:N** kapcsolat (például anya-gyerek kapcsolat)



- **N:M** kapcsolat (hasonlóan jelölendő)

Relációs adatmodell

- Az EK-diagram konkrét adatmodelltől független modellezést tesz lehetővé
- Az általunk használt konkrét adatmodell a **relációs adatmodell**
- Elemei a **táblák**, melyek **sorokból (rekordokból)** épülnek fel.
- Kulcs: egy vagy több attribútum, mely a sort egyértelműen azonosítja és minimális.
- A kulcsban lévő attribútumokat **elsődleges attribútumoknak** nevezzük. Az összes többi attribútum **másodlagos**.

Relációs adatmodell

DOLGOZÓK

d. kód	d. neve	d. címe	fizetés
001	Kovács Pál	Szeged	25.000
002	Nagy Anett	Szeged	30.000
003	Kovács Pál	Pécs	28.000
004	Kis Géza	Pécs	25.000

- Amennyiben csak a táblázat nevét, első sorát (az attribútumok neveit) valamint a kulcsokat adjuk meg, úgy **relációs adatbázissémáról** beszélünk.

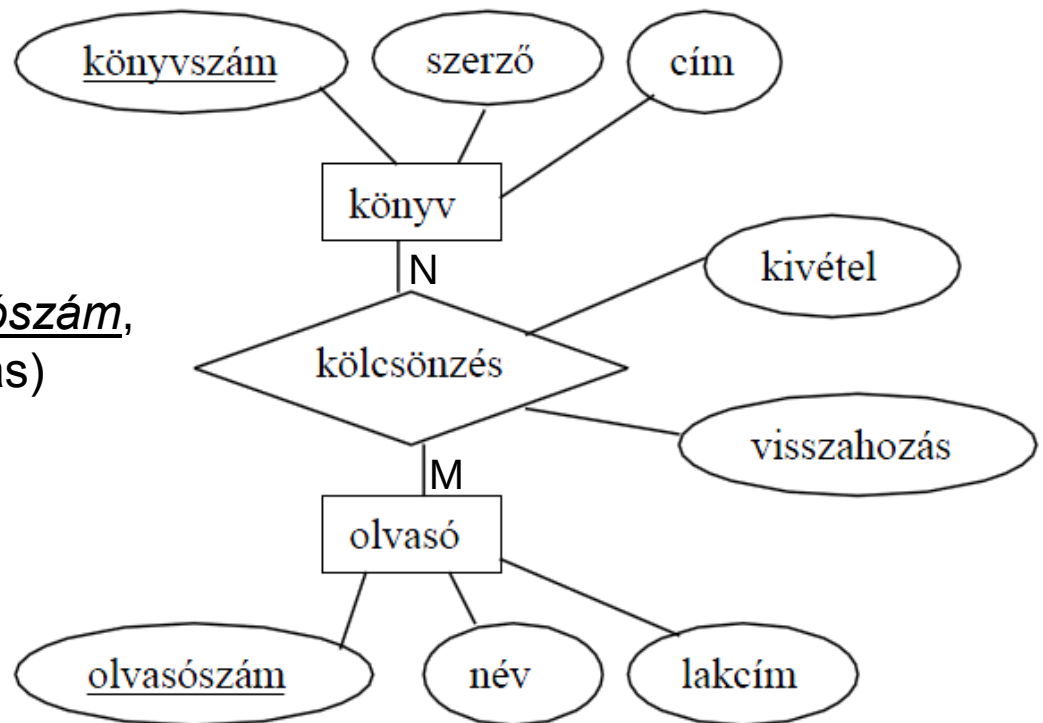
Relációs adatbázisséma készítése EK-modellből

- **Egyedek átírása:** az EK-modell minden egyedéhez felírunk egy relációs adatbázissémát, melynek attribútumai az egyed attribútumai, kulcsa az egyed kulcsattribútumaiból áll
- **Kapcsolatok átírása:**
 - 1:1 kapcsolat esetén kiválasztjuk a kapcsolatban résztvevő egyedek egyikét (bármelyiket) és annak relációs sémájába felvesszük új attribútumként a másik egyed kulcsattribútumait, valamint a kapcsolat attribútumait.
 - 1:N kapcsolat esetén az N oldali egyed relációs sémáját bővítjük úgy, mint 1:1 kapcsolat esetén
 - N:M kapcsolat vagy kettőnél több egyed közötti kapcsolat esetén új relációs sémát veszünk fel, melynek attribútumai a kapcsolódó egyedek kulcsattribútumai valamint a kapcsolat saját attribútumai

Példa

- a kölcsönzés N:M típusú kapcsolat
- ha valamely attribútum egy másik séma elsődleges kulcsára hivatkozik, akkor azt **külső kulcsnak** nevezzük, és dőlt írással jelöljük

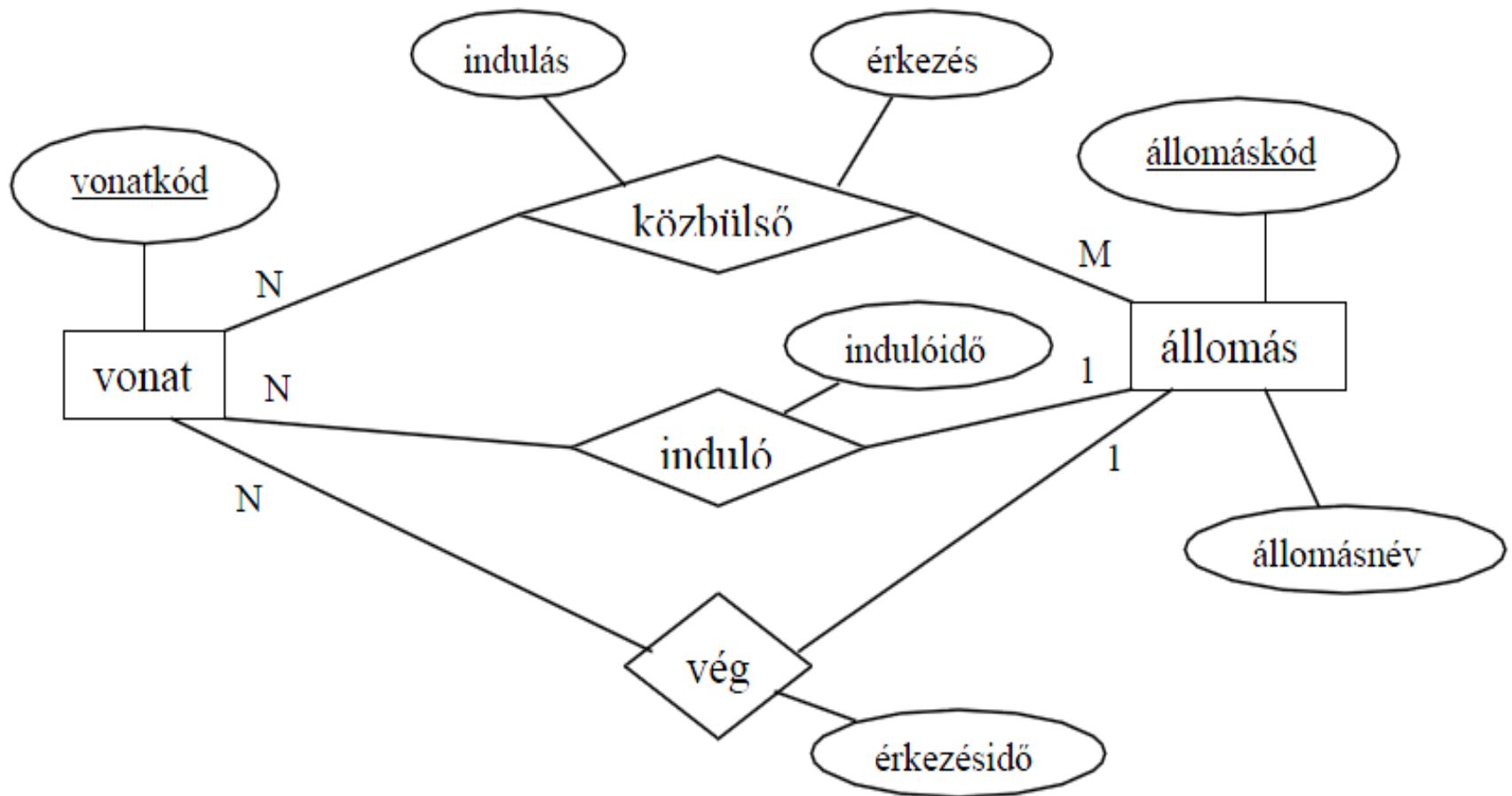
KÖNYV(könyvszám, szerző, cím)
OLVASÓ(olvasószám, név, lakcím)
KÖLCSÖNZÉS(könyvszám, olvasószám,
kivétel, visszahozás)



Feladat

- Vasúti járatok induló-, közbűlső- és végállomásait akarjuk nyilvántartani, beleértve az állomásokról való indulás és érkezés idejét is. A járatot egy járatszám azonosítja, az állomásokról az állomás nevét és az egyértelműen azonosító állomáskódját tartjuk nyilván. Írjuk fel az EK-diagrammot, majd a relációs adatbázissémákat.

Az EK-diagramm



A relációs adatbázisséma

- Az egyedekre és az N:M kapcsolatokra vonatkozó átírási szabályok alkalmazása után:
 - **vonat** (vonatkód)
 - **állomás** (állomáskód, állomásnév)
 - **közbülsőállomás** (állomáskód, vonatkód, indulás, érkezés)

A relációs adatbázisséma

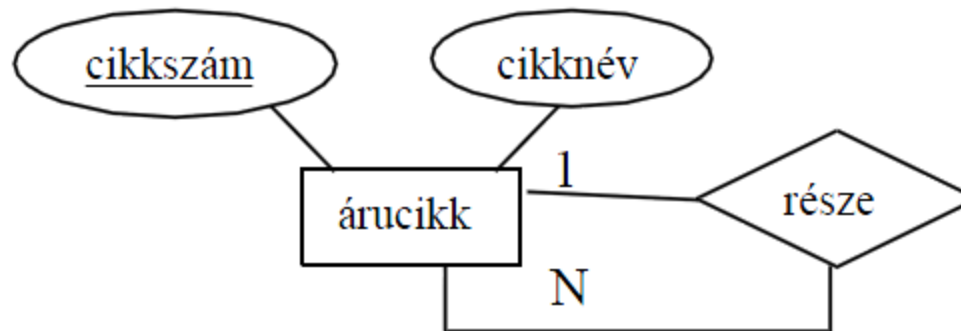
- Az egyedekre és az N:M kapcsolatokra vonatkozó átírási szabályok alkalmazása után:
 - **vonat** (vonatkód)
 - **állomás** (állomáskód, állomásnév)
 - **közbülsőállomás** (állomáskód, vonatkód, indulás, érkezés)
- Az indulóállomásra vonatkozó 1:N kapcsolat esetén a **vonat** sémát bővítenünk kell:
 - **vonat** (vonatkód, *állomáskód*, indulóidő)

A relációs adatbázisséma

- Az egyedekre és az N:M kapcsolatokra vonatkozó átírási szabályok alkalmazása után:
 - **vonat** (vonatkód)
 - **állomás** (állomáskód, állomásnév)
 - **közbülsőállomás** (állomáskód, vonatkód, indulás, érkezés)
- Az indulóállomásra vonatkozó 1:N kapcsolat esetén a **vonat** sémát bővítenünk kell:
 - **vonat** (vonatkód, állomáskód)
- A **vonat** sémát a végállomásra vonatkozó 1:N kapcsolat miatt tovább bővítjük (az állomáskód mező kétszer szerepel a sémában, ezért meg kell különböztetnünk őket névlegesen)
 - **vonat** (vonatkód, indulóállomáskód, indulóidő, végállomáskód, érkezésiidő)

Rekurzív kapcsolat

- egy egyed ugyanabban a kapcsolatban többször vesz részt
- **Példa:** egy számítástechnikai kereskedés árul számítógép-konfigurációkat, de árul külön alkatrészeket is, melyek azonban konfigurációknak is lehetnek elemei)



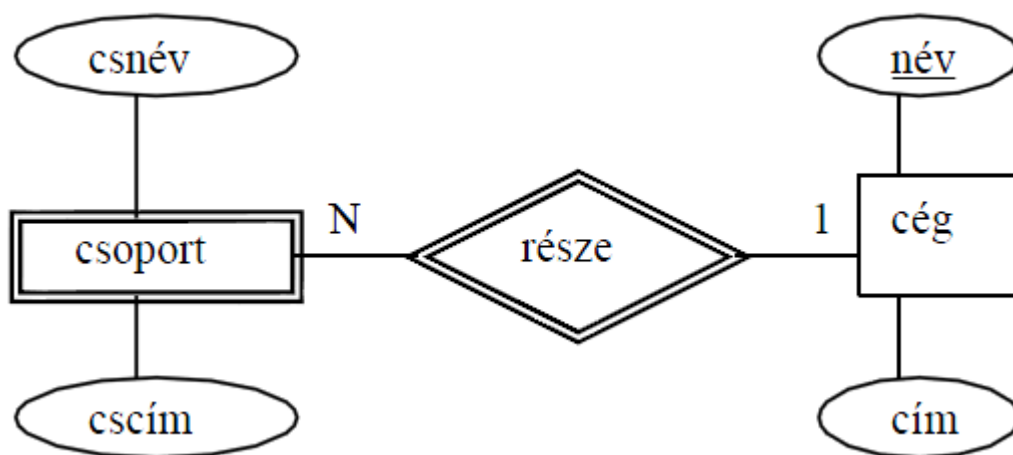
- **árucikk** (cikkszám, cikknév)
- Az 1:N kapcsolat átírása után (megkülönböztetve a két *cikkszám* mezőt):
- **árucikk** (cikkszám, cikknév, *minekrésze_cikkszám*)

Gyenge entitás

- attribútumai nem határozzák meg egyértelműen csak a kapcsolatai révén lesz egyértelműen meghatározott (jele: kettős téglalap)
- az ilyen egyedet meghatározó kapcsolat neve **meghatározó kapcsolat** (jele: kettős rombusz)
- a meghatározó kapcsolat mindig 1:N típusú
- gyenge entitások leképezése: a gyenge entitás relációssémáját bővíteni kell a meghatározó kapcsolat(ok)ban szereplő egyed(ek) kulcsával

Példa gyenge entitásra

- A csoportnév nem kulcs (sok cégnél lehet ugyanolyan nevű csoport), de ha a kapcsolaton belül a céget is be vesszük az azonosításba, akkor egyértelmű lesz, hogy melyik csoportról beszélünk.



CSOPORT(csnév, cscím, név)
CÉG(név, cím)

Összetett és többértékű attribútumok leképezése

- Összetett attribútumok: a sémát úgy vesszük fel, hogy abban csak elemi attribútumok legyenek
- Többértékű attribútumok:
 - Új tábla felvétele. A KÖNYV(könyvszám, szerző, cím) sémát helyettesítjük a KÖNYV(könyvszám, cím) és SZERZŐ(könyvszám, szerző) sémákkal.
 - Ha a sorrend is fontos, akkor a SZERZŐ táblát egy sorszám mezővel kell bővíteni: SZERZŐ(könyvszám, sorszám, szerző).

Normalizálás

- Cél: redundancia csökkentése → adatbiztonság növelése
- **1. normálforma:** a cellákban csak egyszerű adatok vannak (nincsenek értékthalmazok, érték n-esek)
- **2. normálforma:** minden másodlagos attribútum teljesen függ a kulcstól, azaz a kulcsból bármely attribútumot elhagyva már nem áll fenn függőség
- **3. normálforma:** nincs kulcstól való tranzitív függés
- **Tranzitív függés:** elsődleges attribútum → másodlagos attribútum → másodlagos attribútum

Példa

- **számla** (számlaszám, dátum, vevőkód, vevőnév, vevőcím, sorok)
- a *sorok* mező a számlán szereplő sorokat jelzi
- egy sor felépítése a következő: (árukód, árunév, egységár, mennyiség)
- egy sort az árukód egyértelműen azonosít, azaz ugyanazon a számlán nem szerepelhet két ugyanolyan árukódot tartalmazó sor

1. normálforma

- A *sorok* cella nem egyszerű, hiszen egy sor 4 bejegyzést tartalmaz.
- **megoldás:**
 - válasszuk le a nem egyszerű cellát az eredeti sémából
 - vegyünk fel egy új sémát, melynek mezői a leválasztott cella mezői valamint az eredeti séma kulcsa ('új séma kulcsa' = 'eredeti séma kulcsa' + 'leválasztott cella kulcsa')

számla (számlaszám, dátum, vevőkód, vevőnév, vevőcím, sorok)
egy sor felépítése: (árukód, árunév, egységár, mennyiség)

számla (számlaszám, dátum, vevőkód, vevőnév, vevőcím)
sorok (számlaszám, árukód, árunév, egységár, mennyiség)

2. normálforma

számla (számlaszám, dátum, vevőkód, vevőnév, vevőcím)

sorok (számlaszám, árukód, árunév, egységár, mennyiség)

- A **számla** sémában a kulcs egyelemű, így a kulcsból bármely attribútumot elhagyva már nem áll fenn függőség (hiszen az üres halmaz nem határozhat meg semmit), tehát a **számla** séma már 2 NF-ben van.
- A **sorok** séma nincs 2NF-ben, mert az *árukód* egyedül is meghatározza, az *árunév* mezőt (az *egységár* mezőt nem, mert az függ attól, mikor adták ki a számlát, tehát a *számlaszám* mezőtől is).
- **megoldás:**
 - válasszuk le azokat a cellákat az eredeti sémából, melyeket a kulcs része egyértelműen meghatároz (jelen esetben az *árunév* mezőt)
 - vegyünk fel egy új sémát, melynek mezői a leválasztott mezők és az őket meghatározó kulcsrészlet (jelen esetben az *árukód* mező) ('új séma kulcsa' = 'leválasztott mezőket meghatározó kulcsrészlet')

számla (számlaszám, dátum, vevőkód, vevőnév, vevőcím)

sorok (számlaszám, árukód, egységár, mennyiség)

áruk (árukód, árunév)

3. normálforma

számla (számlaszám, dátum, vevőkód, vevőnév, vevőcím)

sorok (számlaszám, árukód, egységár, mennyiség)

áruk (árukód, árunév)

- A **sorok** illetve az **áruk** sémában nincs tranzitív (többlépéses) függés, így azok már 3NF-ben vannak.
- A **számla** sémában a következő tranzitív függés van: a *számlaszám* meghatározza a *vevőkód* mezőt, az pedig a *vevőnév* illetve a *vevőcím* mezőket: *számlaszám* → *vevőkód* → *vevőnév*, *vevőcím*
- **megoldás:**
 - az eredeti sémában csak a középső (csatoló) mezőt hagyjuk meg (jelenleg: *vevőkód*)
 - vegyünk fel egy új sémát, melynek kulcsa a csatolómező lesz, mezői pedig a tranzitív függés második felét alkotó mezők

számla (számlaszám, dátum, *vevőkód*)

vevők (vevőkód, vevőnév, vevőcím)

sorok (számlaszám, árukód, egységár, mennyiség)

áruk (árukód, árunév)

Példa

- Normalizáljuk az alábbi DVDkölcsönző adatbázissémát
- **DVD** (DVD száma, film címe, film száma, rendelés száma, rendelés dátuma, sorok)
- egy DVD-n csak egy film van, de egy film több példányban is meglehet
- a rendelés dátuma azt jelzi, hogy a kölcsönző mikor vásárolta a filmet
- a *sorok* mező azt tartalmazza, hogy mikor ki kölcsönözte ki a DVD-t
- egy sor felépítése a következő: (kölcsönzés száma, tag sorszáma, tag neve, kölcsönzés dátuma, visszahozás dátuma)
- egy sort a kölcsönzés száma egyértelműen azonosít, a kölcsönzés száma úgy jön létre, hogy a tag sorszámahoz “hozzáragasztják” a kölcsönzés dátumát

1. normálforma

DVD (DVD száma, film címe, film száma, rendelés száma, rendelés dátuma, sorok)
Egy sor felépítése: (kölcsönzés száma, tag sorszáma, tag neve, kölcsönzés dátuma, visszahozás dátuma)

1. normálforma

DVD (DVD száma, film címe, film száma, rendelés száma, rendelés dátuma, sorok)
Egy sor felépítése: (kölcsönzés száma, tag sorszáma, tag neve, kölcsönzés dátuma, visszahozás dátuma)

felbontjuk a *sorok* cellát, mert az nem egyszerű:

DVD (DVD száma, film címe, film száma, rendelés száma, rendelés dátuma)

sorok (DVD száma, kölcsönzés száma, tag sorszáma, tag neve, kölcsönzés dátuma, visszahozás dátuma)

2. normálforma

DVD (DVD száma, film címe, film száma, rendelés száma, rendelés dátuma)

sorok (DVD száma, kölcsönzés száma, tag sorszáma, tag neve, kölcsönzés dátuma, visszahozás dátuma)

2. normálforma

DVD (DVD száma, film címe, film száma, rendelés száma, rendelés dátuma)

sorok (DVD száma, kölcsönzés száma, tag sorszáma, tag neve, kölcsönzés dátuma, visszahozás dátuma)

A **sorok** sémát felbontjuk, mert ott a *kölcsönzés száma* önmagában is egyértelműen azonosítja a *tag sorszáma*, *tag neve*, *kölcsönzés dátuma* illetve *visszahozás dátuma* mezőket:

DVD (DVD száma, film címe, film száma, rendelés száma, rendelés dátuma)

sorok (DVD száma, kölcsönzés száma)

kölcsönzések (kölcsönzés száma, tag sorszáma, tag neve, kölcsönzés dátuma, visszahozás dátuma)

3. normálforma

DVD (DVD száma, film címe, film száma, rendelés száma, rendelés dátuma)

sorok (DVD száma, kölcsönzés száma)

kölcsönzések (kölcsönzés száma, tag sorszáma, tag neve, kölcsönzés dátuma, visszahozás dátuma)

3. normálforma

DVD (DVD száma, film címe, film száma, rendelés száma, rendelés dátuma)

sorok (DVD száma, kölcsönzés száma)

kölcsönzések (kölcsönzés száma, tag sorszáma, tag neve, kölcsönzés dátuma, visszahozás dátuma)

A **DVD** séma tranzitív függései:

DVD száma → film száma → film címe

DVD száma → rendelés száma → rendelés dátuma

A **kölcsönzések** séma tranzitív függése:

kölcsönzés száma → tag sorszáma → tag neve

3. normálforma

DVD (DVD száma, film címe, film száma, rendelés száma, rendelés dátuma)

sorok (DVD száma, kölcsönzés száma)

kölcsönzések (kölcsönzés száma, tag sorszáma, tag neve, kölcsönzés dátuma, visszahozás dátuma)

A **DVD** séma tranzitív függései:

DVD száma → film száma → film címe

DVD száma → rendelés száma → rendelés dátuma

A **kölcsönzések** séma tranzitív függése:

kölcsönzés száma → tag sorszáma → tag neve

sorok (DVD száma, kölcsönzés száma)

kölcsönzések (kölcsönzés száma, tag sorszáma, kölcsönzés dátuma, visszahozás dátuma)

tagok (tag sorszáma, tag neve)

DVD (DVD száma, film száma, rendelés száma)

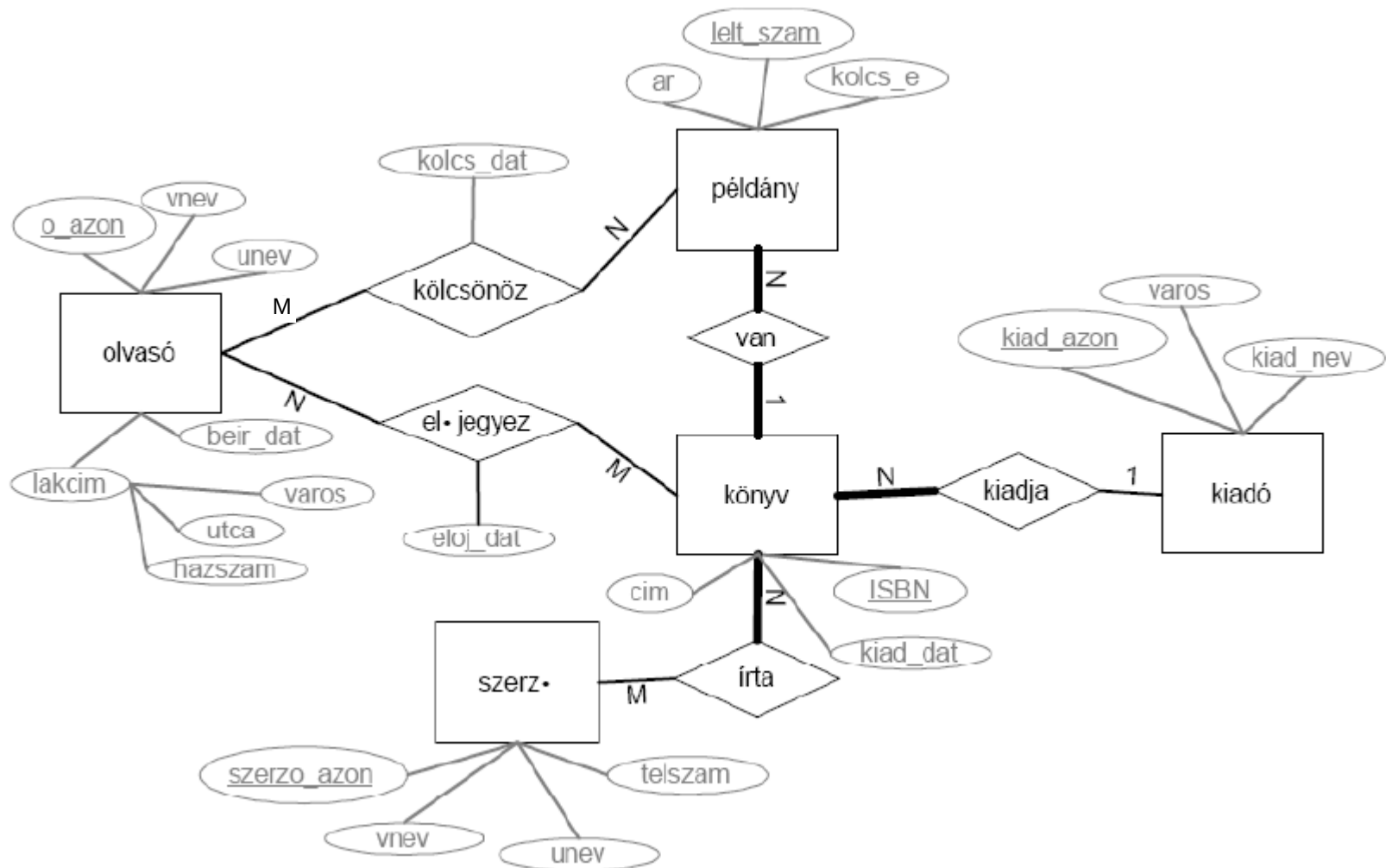
filmek (film száma, film címe)

rendelések (rendelés száma, rendelés dátuma)

Suli könyvtár

- Készítsük el egy iskolai könyvtár adatbázisát
 - Egyedek: olvasó, könyv, példány, szerző, kiadó
 - Kapcsolatok:
 - egy olvasó több könyvet is kölcsönözhet
 - a könyvekre több olvasó is adhat le előjegyzést
 - egy könyvnek több szerzője is lehet
 - egy könyvnek csak egy kiadója van
 - egy könyv több példányban is meglehet
 - Az attribútumokat egyelőre ne tüntessük fel

EK-diagramm



Egyedekre vonatkozó sémák

- **olvasó** (oazon, vnev, unev, varos, utca, hazszam, beir_dat)
- **könyv** (ISBN, cim, kiad_dat)
- **példány** (lelt_szam, kolcs_e, ar)
- **kiadó** (kiadazon, varos, kiad_nev)
- **szerző** (szerzoazon, vnev, unev, telszam)

Kapcsolatok átírása

- **kiad: könyv** (ISBN, cim, kiad_dat, *kiad_azon*)
- **van: példány** (lelt_szam, kolcs_e, ar, *ISBN*)
- **kölcsönöz: kölcsönöz** (*lelt_szam*, *o_azon*, kolcs_dat)
- **írta: írta** (*szerzo_azon*, *ISBN*)
- **előjegyez: előjegyez** (*o_azon*, *ISBN*, eloj_dat)

Tehát a sémák (3NF-ben)

- **olvasó** (oazon, vnev, unev, varos, utca, hazszam, beir_dat)
- **könyv** (ISBN, cim, kiad_dat, *kiadazon*)
- **példány** (leltszam, kolcs_e, ar, *ISBN*)
- **kiadó** (kiadazon, varos, kiad_nev)
- **szerző** (szerzoazon, vnev, unev, telszam)
- **írta** (szerzoazon, ISBN)
- **előjegyez** (oazon, ISBN, eloj_dat)
- **kölcsönöz** (leltszam, oazon, kolcs_dat)