

<https://www.mysqltutorial.org/>  
<https://dev.mysql.com/>  
[http://www.php-editors.com/articles/sql\\_phpmyadmin.php](http://www.php-editors.com/articles/sql_phpmyadmin.php)  
<https://docs.phpmyadmin.net/hu/latest/>

# ALAPFOGALMAK

## Feladat 1

Tároljuk el egy kisebb cég alkalmazottainak adatait táblázatos formában!

### ALKALMAZOTT

| azonosító | név   | cím   | fkód   | foglalkozás   |
|-----------|-------|-------|--------|---------------|
| az_1      | név_1 | cím_1 | fkód_1 | foglalkozás_1 |
| az_2      | név_2 | cím_2 | fkód_2 | foglalkozás_2 |
| az_3      | név_3 | cím_3 | fkód_3 | foglalkozás_3 |

### Redundancia

#### Anomáliák

- bővítési anomália
- módosítási anomália
- törlési anomália

### Megoldás

#### ALKALMAZOTT

| <u>azonosító</u> | név   | cím   | <i>fkód</i> |
|------------------|-------|-------|-------------|
| az_1             | név_1 | cím_1 | fkód_1      |
| az_2             | név_2 | cím_2 | fkód_2      |
| az_3             | név_2 | cím_3 | fkód_3      |

| FOGLALKOZÁS |             |
|-------------|-------------|
| <u>fkód</u> | foglalkozás |
| fkód_1      | foglalkozás |
| fkód_2      | foglalkozás |

ALKALMAZOTT(azon, nev, cím, *fkod*)    FOGLALKOZAS(fkod, foglalkozas)



## Feladat 2

Tároljuk el ismerőseink nevét, lakóhelyének városát, az utca nevét, a házszámot és az irányítószámot! (Tételezzük fel, hogy nincs két azonos nevű ismerősünk és azt is, hogy minden városnak csak egy irányítószáma van.)

### ISMERŐSÖK

| név   | város   | utca, hsz. | irányítószám |
|-------|---------|------------|--------------|
| név_1 | város_1 | cím_1      | írszám_1     |
| név_2 | város_1 | cím_1      | írszám_2     |

### Problémák

redundancia, anomáliák

Össze nem tartozó adatokat sűrítettünk egy táblába.

### Megoldás

| <u>Név</u> | város   | cím   |
|------------|---------|-------|
| név_1      | város_1 | cím_1 |
| név_2      | város_2 | cím_2 |

| <u>város</u> | irányítószám |
|--------------|--------------|
| város_1      | írszám_1     |

ISMERŐSÖK(nev, varos, cim)      VÁROS(varos, irányítószám)

### Példa 1

Tegyük fel, hogy a város nevének tárolására 32 betűnyi, míg az irányítószám tárolására 4 betűnyi helyet tartunk fenn. Hány ismerősünknek kell átlagosan egy városban lakni, hogy a fenti elrendezés kisebb helyet foglaljon el, mint az eredeti?

Eredeti táblázat első adat:  $32 + 4 = 36$ .

Két táblázat első adat:  $32 + 32 + 4 = 68$ .

Adattárolási többlet az első adathál:  $t = 68 - 36 = 32$ .

További adatok esetén a két táblázatos rendszerrel elegendő a várost eltárolni, ami 32 betű szemben az eredeti táblázat 36 betűjével. A megtakarítás tehát:  $m = 36 - 32 = 4$  betű.

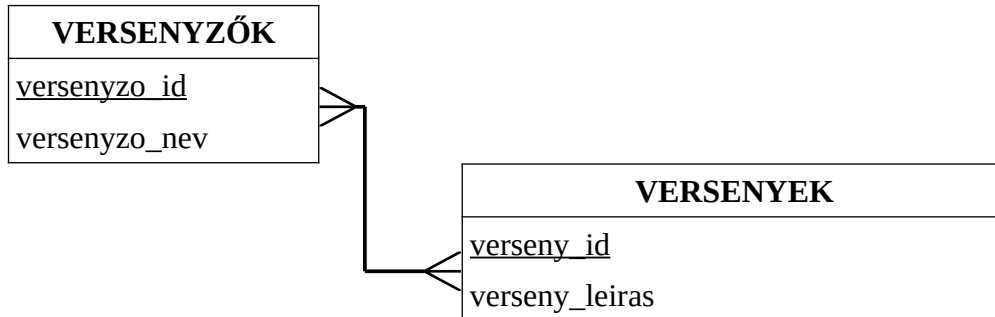
Megoldás:

$t / m = 32 / 4 = 8$ , azaz ha átlagosan 9 lakos lakik ugyanabban a városban, akkor a két módszer ugyanannyi helyet foglal.

### Feladat 3

Tároljuk el versenyzők és versenyek adatait:

versenyzo\_nev, versenyzo\_id, verseny\_id, verseny\_leiras

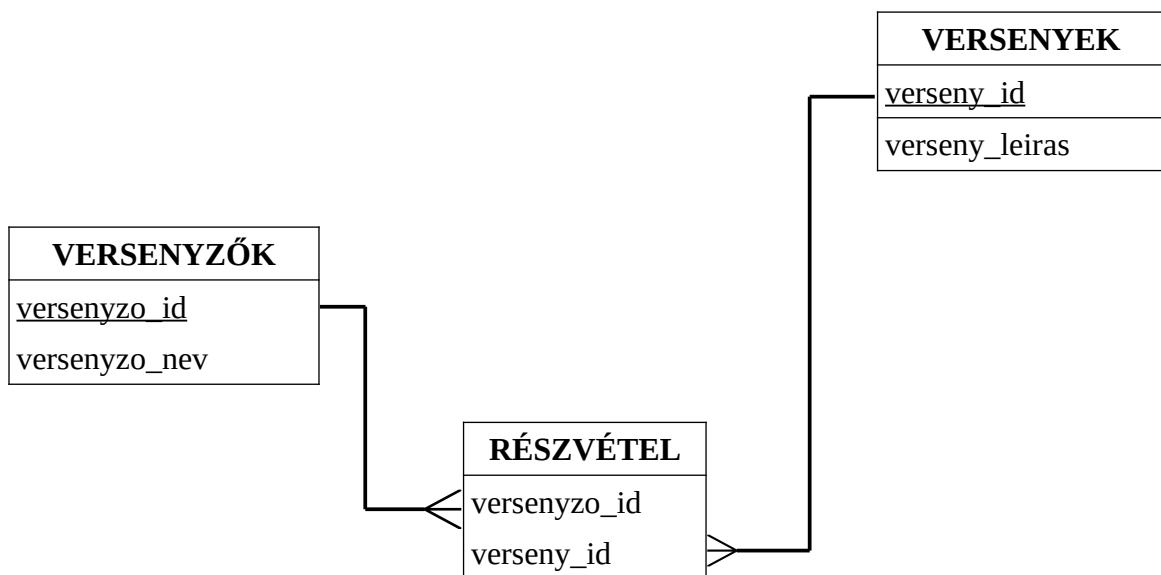


### Problémák

- Több-több kapcsolat: az egyik tábla kulcsértéke többször megjelenhet a másik táblában, a másik tábla kulcsértéke szintén sokszor előfordulhat az első táblában.
- Redundancia

### Megoldás

- Felbontás egy-több kapcsolatokra.
- Egy közbülső tábla létrehozása a két tábla között, amely összerendeli az összetartozó adatokat.



A relációs modellen alapuló adatbázis-kezelő rendszerek a következő elnevezéseket használják a rendszer egy-egy elemére:

## **Reláció**

Egy táblázat, amely sorokba és oszlopokba rendezve tartalmazza az adatokat. (Halmazok tagjaiból képzett párok, hármasok stb. halmaza reláció, akkor is, ha nem tartalmaz minden lehetséges kombinációt:  $\rho \subseteq H_1 \times \dots \times H_n$ )

## **Attribútum**

A táblázat egy oszlopa. Formailag azonos elemeket tartalmaznak: szám, szöveg stb.. Szokásos elnevezésük még az oszlop és a mező.

## **Sor**

A sorok formailag különböző, de valamiképp együvé tartozó adatokat tartalmaznak. A sor elnevezés mellett használatos még a rekord kifejezés is.

## **Adatbázis**

A relációkból felépített teljes rendszer. Az adatbáziskezelő rendszerben található adatok halmaza.

## **Elsődleges kulcs**

A táblázat rekordjait egymástól egyértelműen megkülönböztető mező(k).

## **Idegen kulcs**

Kapcsolatot biztosító mező. (A hivatkozott mező a másik táblában elsődleges kulcs.)

## **Kapcsolatok**

1 - 1, 1 - N, N - N

## **Anomáliák**

törlési, módosítási, bővítési

## **Első normál forma<sup>1</sup>**

A táblázat első normálformában van, ha minden sora különböző és nincsenek többértékű tulajdonságok (A kulcsmezőtől funkcionálisan függ minden további mező.)

## **Második normál forma**

Az összetartozó adatok külön táblákban foglalnak helyet. (Az adatbázis első normálformában van, és a kulcsmezőtől való függés teljes, a kulcs minimális.)

---

1 **normalizálás** lat 4. inf adatbázisokban az adatok egyértelmű, ellentmondásmentes tárolása (Bakos Ferenc: Idegen szavak és kifejezések szótára)