

Adatbázis rendszerek BSc

7. gyak.

2021. 10. 27.

Készítette:
Kovács Krisztián
Programtervező informatikus
WIQPM2

Miskolc, 2021

1. feladat

Hozza létre az alábbi táblát:

```
CREATE TABLE termek (tkod INT PRIMARY KEY, nev VARCHAR2(50) NOT NULL, ear INT CHECK (EAR > 0), kategoria CHAR(20));
```

Vigyen fel 5 rekordot a táblába. Ha létrehozta a táblát, akkor használja a következő utasítást:
Describe táblanév;

```
INSERT INTO termek VALUES (1,"lapát", 2000,"K1");
```

```
INSERT INTO termek VALUES (2,"seprű", 4000,"K1");
```

```
INSERT INTO termek VALUES (3,"mosogató gél", 1500,"K2");
```

```
INSERT INTO termek VALUES (4,"szappan", 1000,"K2");
```

```
INSERT INTO termek VALUES (5,"pohár", 2400,"K3");
```

Field	Type	Null	Key	Default	Extra
tkod	int(11)	NO	PRI	NULL	
nev	varchar(50)	NO		NULL	
ear	int(11)	YES		NULL	
kategoria	char(20)	YES		NULL	

✓ A(z) 1 sor beszúrása megtörtént. (A lekérdezés 0,0000 másodpercig tartott.)

```
INSERT INTO termek VALUES (1,"lapát", 2000,"K1")
```

✓ A(z) 1 sor beszúrása megtörtént. (A lekérdezés 0,0000 másodpercig tartott.)

```
INSERT INTO termek VALUES (2,"seprű", 4000,"K1")
```

⚠ Warning: #1366 Incorrect string value: 'xC5xB1' for column 'nev' at row 1

✓ A(z) 1 sor beszúrása megtörtént. (A lekérdezés 0,0000 másodpercig tartott.)

```
INSERT INTO termek VALUES (3,"mosogató gél", 1500,"K2")
```

✓ A(z) 1 sor beszúrása megtörtént. (A lekérdezés 0,0000 másodpercig tartott.)

```
INSERT INTO termek VALUES (4,"szappan", 1000,"K2")
```

✓ A(z) 1 sor beszúrása megtörtént. (A lekérdezés 0,0000 másodpercig tartott.)

```
INSERT INTO termek VALUES (5,"pohár", 2400,"K3")
```

2. feladat

Adjon meg az előző táblához olyan felviteli parancsokat, melyet nem fog

```
INSERT INTO termek VALUES (1,'lapát', 2000,'K1');
```

```
INSERT INTO termek VALUES (8,NULL, 4000,'K1');
```

```
INSERT INTO termek VALUES (15,'pohár A', 0,'K3');
```

```
INSERT INTO termek VALUES (15,'pohár A', 20,'K3');
```

```
INSERT INTO termek VALUES (15,'pohár A', '20','K3');
```

Hiba

SQL lekérdezés:

```
INSERT INTO termek VALUES (1,'lapát', 2000,'K1');
```

A MySQL mondta: ?

```
#1054 - A(z) ''lapát'' oszlop ervenytelen 'field list'-ben
```

Hiba

SQL lekérdezés:

```
INSERT INTO termek VALUES (8,NULL, 4000,'K1');
```

A MySQL mondta: ?

```
#1054 - A(z) ''K1'' oszlop ervenytelen 'field list'-ben
```

Hiba

SQL lekérdezés:

```
INSERT INTO termek VALUES (15,'pohár A', 0,'K3');
```

A MySQL mondta: ?

```
#1064 - A Szintaktikai hiba a 'A', 0,'K3')'-hez kozeli a 1 sorban
```

Hiba

SQL lekérdezés:

```
INSERT INTO termek VALUES (15,"pohár A", 20,'K3')
```

A MySQL mondta: ?

#1064 - A Szintaktikai hiba a 'A", 20,'K3')'-hez közeli a 1 sorban

Hiba

SQL lekérdezés:

```
INSERT INTO termek VALUES (15,'pohár A', '20','K3')
```

A MySQL mondta: ?

#1064 - A Szintaktikai hiba a 'A', '20','K3')'-hez közeli a 1 sorban

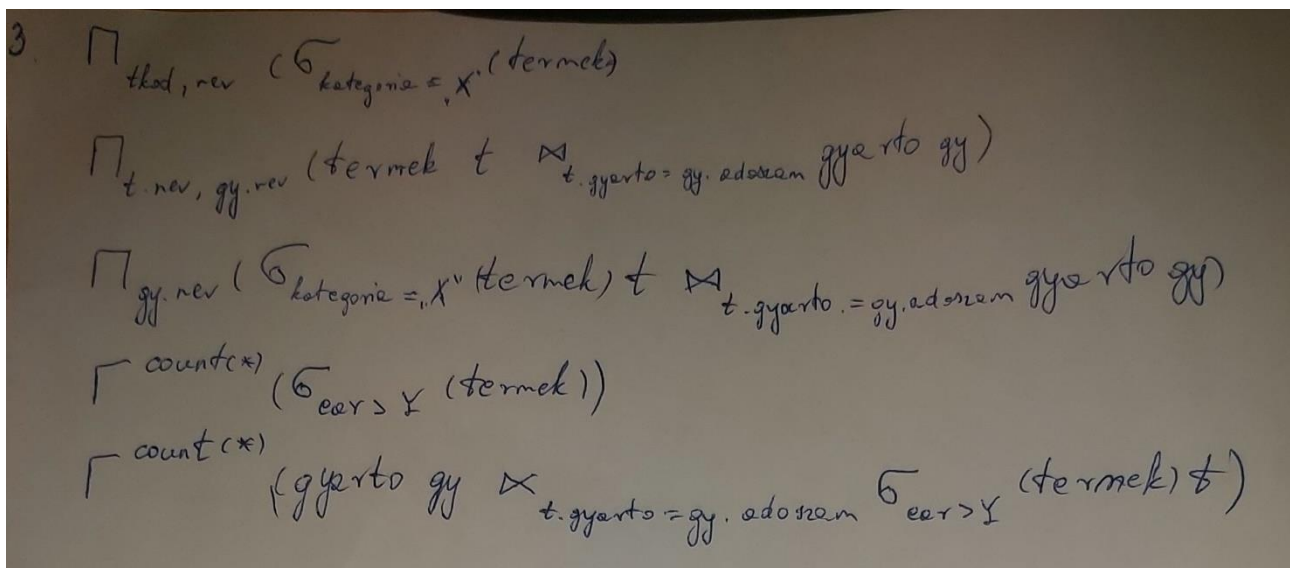
3. feladat

Adott az alábbi relációs adatbázis:.

```
CREATE TABLE gyarto (adoszam INT PRIMARY KEY,  
nev VARCHAR2(30) NOT NULL, varos VARCHAR2(200));  
CREATE TABLE termek (tkod INT PRIMARY KEY,  
nev VARCHAR2(50) NOT NULL, ear INT CHECK (EAR > 0),  
kategória CHAR(10), gyarto INT REFERENCES GYARTO);
```

Adja meg az alábbi műveletek relációs algebrai alakját.

- az X kategóriájú termékek neve és kódja
- a termékek neve és a gyártó neve együtt
- az X kategóriájú termékek gyártóinak nevei
- az Y-nál drágább termékek darabszáma
- az Y-nál drágább termékek gyártóinak darabszáma



4. feladat

Adott az alábbi relációs adatbázis:

CREATE TABLE gyarto (adoszam INT PRIMARY KEY,

nev VARCHAR2(30) NOT NULL, varos VARCHAR2(200));

CREATE TABLE termek (tkod INT PRIMARY KEY,

nev VARCHAR2(50) NOT NULL, ear INT CHECK (EAR > 0),

kategória CHAR(10), gyarto INT REFERENCES GYARTO);

CREATE TABLE alkatresz (akod INT PRIMARY KEY,

nev VARCHAR2(50) NOT NULL);

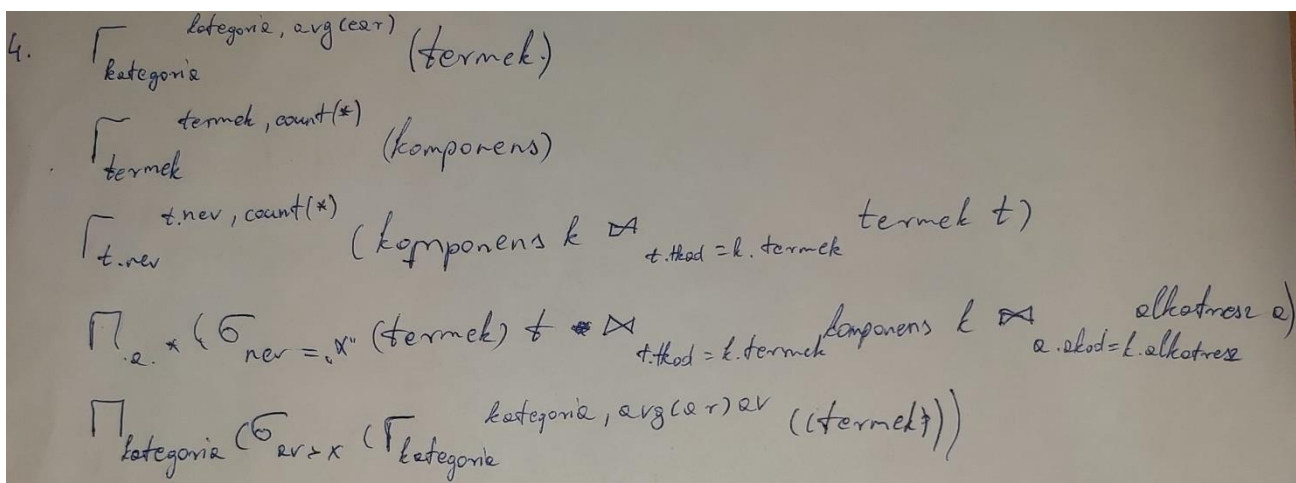
CREATE TABLE egysegek (aru INT REFERENCES termek,

db INT CHECK (db > 0)) ;

CREATE TABLE komponens (termek INT REFERENCES termek, alkatresz INT REFERENCES alkatresz);

Adja meg az alábbi műveletek relációs algebrai alakját!

- kategóriánként a termékek átlagára
- termékenként az alkatrészek darabszáma (termékkód, darabszám)
- termékenként az alkatrészek darabszáma (terméknév, darabszám)
- az X nevű termékben szereplő alkatrészek listája
- azon kategóriák, ahol a termékek átlagára nagyobb mint X



5. feladat

Adott az alábbi relációs adatbázis:

CREATE TABLE tanfolyam (tkod INT, ar INT, tipus CHAR(30), megnevezes

VARCHAR2(100), PRIMARY KEY (tkod));

CREATE TABLE résztvevo (tajszam CHAR(13), nev CHAR(30), lakcim

VARCHAR2(100), PRIMARY KEY (tajszam));

CREATE TABLE befizetes (diak INT, kurzus INT, befizetes INT,

FOREIGN KEY (diak) REFERENCES résztvevo, FOREIGN KEY (kurzus)

REFERENCES (tanfolyam))

Adja meg az alábbi műveletek relációs algebrai alakját.

- a befizetések összege Ft-ban és Euróban
- a befizetések összege résztvevőnként (név) egy adott tanfolyamra
- mennyien vesznek részt az egyes tanfolyamokon
- mennyien vesznek részt az egyes tanfolyamokon, a 0 érték is jelenjen meg
- mely tanfolyamokra nem jelentkeztek még
- a legtöbb jelentkezőt vonzó tanfolyam neve

5.

$\pi_{\text{sum(befizetes)}, \text{sum(Euro)}} (\sigma_{\text{befizetes} \geq 1305 \text{ Euro}} (\text{befizetes}))$

$\pi_{r.\text{tajszam}} (\sigma_{\text{max}(r.\text{nev}), \text{sum}(b.\text{befizetes})} (\text{befizetes } b \bowtie_{b.\text{diak} = r.\text{tajszam}} \text{resztvevo } r))$

$\pi_{t.\text{tkod}} (\sigma_{\text{max}(r.\text{megn.}), \text{count}(*)} (\text{befizetes } b \bowtie_{b.\text{kurzus} = t.\text{tkod}} \text{tanfolyam } t))$

$\pi_{t.\text{tkod}} (\sigma_{\text{max}(r.\text{megn.}), \text{count}(b.\text{kurzus})} (\text{befizetes } b \bowtie_{b.\text{kurzus} = t.\text{tkod}} \text{tanfolyam } t))$

$\pi_{\text{megnevezes}} (\text{tanfolyam } t \bowtie_{t.\text{tkod} = k.\text{tkod}} (\pi_{t.\text{tkod}} (\text{tanfolyam})) \setminus \pi_{kurzus} (\text{jelentkezes}))$

6. feladat

Adott az alábbi tábla:

CREATE TABLE termek (tkod INT PRIMARY KEY,

nev VARCHAR2(50) NOT NULL, ear INT CHECK (EAR > 0), kategoria CHAR(20));

Végezze el az alábbi SQL műveleteket:

- új rekord felvitele
- az X-nél olcsóbb termékek törlése
- az Y kategóriájú termékek árának csökkentése 10%-kal

INSERT INTO termek VALUES(6,"villa", 123, "V3");

✓ A(z) 1 sor beszúrása megtörtént. (A lekérdezés 0,00

INSERT INTO termek VALUES(6,"villa", 123, "V3")

DELETE FROM termek WHERE ear<500;

✓ 1 sor érintett. (A lekérdezés 0,0000

DELETE FROM termek WHERE ear<500

UPDATE termek SET ear=ear*0.9 WHERE kategoria="K1";

✓ 2 sor érintett. (A lekérdezés 0,0000 másodpercig t

UPDATE termek SET ear=ear*0.9 WHERE kategoria="K1"

7. feladat

Adott az alábbi tábla:

```
CREATE TABLE termek (tkod INT PRIMARY KEY,  
nev VARCHAR2(50) NOT NULL, ear INT CHECK (EAR > 0),  
kategoria CHAR(20));
```

Végezze el az alábbi SQL műveleteket:

- új szín mező felvitele a sémába
- a termek tábla megszüntetése
- azon rekordok megszüntetése, ahol nincs kitöltve a kategória

```
ALTER TABLE termek ADD (szin VARCHAR(50));
```

✓ A MySQL üres eredményhalmazt adott vissza (pl. null).

```
ALTER TABLE termek ADD (szin VARCHAR(50));
```

```
DELETE FROM termek WHERE kategoria IS NULL;
```

✓ 0 sor érintett. (A lekérdezés 0,0000 másodpercet vett igénybe.)

```
DELETE FROM termek WHERE kategoria IS NULL;
```

```
DROP TABLE termek;
```

✓ A MySQL üres eredményhalmazt adott vissza.

```
DROP TABLE termek;
```