

Adatbázisrendszerek I.

BSc

6.gyak.

2021. 11. 27.

**Készítette: Oravecz
Áron**

Mérnökinformatikus

Neptunkód:

D3U3EE

Miskolc, 2021.27.

1. feladat

```
1 • CREATE DATABASE D3U3EE;
2 • USE D3U3EE;
3 • CREATE TABLE termek (tkod INT PRIMARY KEY, nev VARCHAR(50) NOT NULL, ear INT CHECK (EAR > 0), kategoria CHAR(20));
4 • INSERT INTO termek VALUES(0, 'billentyűzet', 5000, 'elektronika');
5 • INSERT INTO termek VALUES(1, 'ásó', 2000, 'kertészet');
6 • INSERT INTO termek VALUES(2, 'lámpa', 3000, 'elektronika');
7 • INSERT INTO termek VALUES(3, 'monitor', 15000, 'elektronika');
8 • INSERT INTO termek VALUES(4, 'szőnyeg', 10000, 'egyéb');
```

2. feladat

INSERT INTO termek VALUES (1, valami, 2000, 'K1'); // létező kulcs

INSERT INTO termek VALUES (8, NULL, 4000, 'K1'); // üres név;

INSERT INTO termek VALUES (15, 'pohár A', 0, 'K3'); // hibás ár

3. feladat

Mentés: neptunkod_7.3.docx

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),
kategoria 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

4. feladat

Mentés: neptunkod_7.4.docx

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),
```

```
kategoria CHAR(10), gyarto INT REFERENCES GYARTO);
```

```
CREATE TABLE alkatresz (akod INT PRIMARY KEY,
nev VARCHAR2(50) NOT NULL);
```

```
CREATE TABLE egysgek (aru INT REFERENCES termek,
db INT CHECK (db > 0));
```

```
CREATE TABLE komposzt (termek INT REFERENCES termek, alkatresz INT REFERENCES
alkatresz);
```

Adja meg az alábbi műveletek relációs algebrai alakját!

- kategóriánként a termékek listája
- 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

1. Π név, kód (δ kategória = 'X'(termek))
2. Π t.nev, gy.nev (termek t \bowtie t.gyarto = gy.adoszam gyarto gy)
3. Π gy.nev (δ kategória = X (termek) t \bowtie t.gyarto = gy.adoszam gyarto gy)
4. Γ count(*) (δ ear > Y (termek))
5. Γ count(*) (gyarto gy \bowtie t.gyarto = gy.adoszam δ ear > Y (termek) t)

1. Γ categoria categoria, avg(ear)) ((termek))
2. Γ termék termék, count(*) (komponens)
3. Γ t.nev t.nev, count(*) (komponens k \bowtie t.tkod = k.termék termék t)
4. Π .a.* (δ nev= X (termek) t \bowtie t.tkod = k.termék komponens k \bowtie
a.akod= k.alkatresz alkatresz a)
5. Π categoria (δ av > X (Γ categoria categoria, avg(ar) av ((termek)))

5. feladat

Mentés: neptunkod_7.5

Adott az alábbi relációs adatbázis:

```
CREATE TABLE tanfolyam (tkod INT, ar INT, tipus CHAR(30), megnevezes
VARCHAR2(100), PRIMARY KEY (tkod));
```

```
CREATE TABLE resztvevo (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 resztvevo, FOREIGN KEY (kurzus)
REFERENCES (tanfolyam)
```

1. Γ sum(befizetes), sum(Eur) (ϵ befizetes/370 Eur (befizetes))
2. Γ ..tajszam max(r.nev),sum(b.befizetes)
(befizetes b \bowtie b.diak= r.tajszam resztvevo r)
3. Γ t.tkod max(r.megnevezes),count(*) (befizetes b \bowtie b.kurzus= t.tkod tanfolyam t)
4. Γ t.tkod max(r.megnevezes),count(b.kurzus) (befizetes b \bowtie + b.kurzus= t.tkod tanfolyam t)
5. Π t.megnevezes (tanfolyam t \bowtie t.tkod= k.tkod (Π tkod (tanfolyam) \ Π kurzus (jelentkezes)) k)
6. Π mn (δ db = (Γ {max(db)} (X)) (X))

6. feladat

```
1 • CREATE DATABASE D3U3EE;
2 • USE D3U3EE;
3 • CREATE TABLE termekek (tkod INT PRIMARY KEY, nev VARCHAR(50) NOT NULL, ear INT CHECK (EAR > 0), kategoria CHAR(20));
4 • INSERT INTO termekek VALUES(0,'ásó',1500,'k1');
5 • INSERT INTO termekek VALUES(1,'kapa',2000,'k2');
6 • INSERT INTO termekek VALUES(2,'harang',2500,'k3');
7 • DELETE FROM termekek WHERE ear < 1500;
8 • UPDATE termekek SET ear = ear*0.9 WHERE kategoria = 'Y';
```

7. feladat

```
1 • CREATE DATABASE D3U3EE;
2 • USE D3U3EE;
3 • CREATE TABLE termekek (tkod INT PRIMARY KEY, nev VARCHAR(50) NOT NULL, ear INT CHECK (EAR > 0), kategoria CHAR(20));
4 • ALTER TABLE termekek ADD (szin CHAR(20));
5 • DROP TABLE termekek;
6 • DELETE FROM termekek WHERE kategoria IS NULL;
```