

# **Adatbázis rendszerek 1. BSc**

**7.Gyak**

**2022.10.25.**

**Készítette:**

Nyíri Levente

F023QC

## **1. feladat**

**Mentés: neptunkod\_7.1.txt Adottak**

**az alábbi tábla:**

**KÖNYV (isbn C(20) PK, cim C(40), targy C(30), ar INT)**

**Adja meg az alábbi műveletek relációs algebrai alakját.**

- könyvek címei,**
- a 2000-nél drágább könyvek adatai,**
- az 1000-nél olcsóbb könyvek címei,**

- a tárgyak nevei,
- az 'AB' tárgyú könyvek címe és ára.

```

Πcím(könyv)
σar<2000(könyv)
Πcím(σar<1000(könyv))
Πtárgy(könyv)
Πcím,ár(σtárgy='AB'(könyv))

```

## 2. feladat

Adottak az alábbi tábla:

Mentés: neptunkod\_7.2.txt

KÖNYV (isbn C(20) PK, cím C(40), tárgy C(30), ár INT)

Adja meg az alábbi műveletek relációs algebrai alakját.

- a könyvek darabszáma,
- a könyvek átlagára,
- a legolcsóbb könyv ára,
- az 'AB' kategóriájú könyvek darabszáma,
- a legdrágább AB kategóriájú könyv ára,
- az átlagárnál drágább könyvek címei,
- az átlagárnál drágább könyvek darabszáma.

```

Γ count(*) (könyv)
Γ avg(ár) (könyv)
Γ min(ár) (könyv)
Γcount(*) (σtárgy='AB' (könyv))
Γmax(ár) (σtárgy='AB' (könyv))
Πcím(σ(ar> (Γ{avg(ár)} (könyv)))) (könyv))
Γcount(*) (σ(ar > (Γ{avg(ár)} (könyv)))) (könyv))

```

## 3. feladat

Mentés: neptunkod\_7.3.txt

Adott az alábbi relációs adatbázis:.

```
CREATE TABLE gyarto (adoszam INT PRIMARY KEY, nev
VARCHAR2(30) NOT NULL, varos VARCHAR2(200));
CREATE TABLE termék (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

```
πtkod,nev(σkategória = x(termek))
πtermeknev,gyartonev(termek t ⋈ termékgyarto = gyartoadoszam gyarto gy)
πgyartonev(σkategória = x(termek)t ⋈ termékgyarto =gyarto adoszam gyarto gy)
Γcount(*) (σear>Y (termek))
Γcount(*) (gyarto gy ⋈ termékgyarto=gyartoadoszam σear>Y (termek) t )
```

#### 4. feladat

Mentés: neptunkod\_7.4.txt

Adott az alábbi relációs adatbázis:

```
CREATE TABLE gyarto (adoszam INT PRIMARY KEY, nev
VARCHAR2(30) NOT NULL, varos VARCHAR2(200));
CREATE TABLE termék (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 termék, db**

**INT CHECK (db > 0)) ;**

**CREATE TABLE komponens (termek INT REFERENCES termék,  
alkatrész INT REFERENCES**

**alkatrész);**

**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**

```
Γkategoriakategoria,avg(ar)((termek))  
Γtermektermek,count(*) (komponens)  
Γtermeknevtermeknev,count(*) (komponens k ⋈tkod =k.termek termék t)  
Π.a.* ( ⋈ nev= x (termek) t ⋈ t.tkod = k.termek komponens k ⋈ a.akod= k.alkatresz alkatresz a)  
Π kategoria ( ⋈ av > x (Γ kategoria kategoria, avg(ar) av ( (termek)))
```

## **5. feladat**

**Mentés: neptunkod\_7.5.txt**

**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)**

**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

```
Γ sum(befizetes), sum(Eur) ( ε befizetes/370 Eur (befizetes))  
Γ r..tajszam max(r.nev),sum(b.befizetes) (befizetes b ⋈ b.diak= r.tajszam resztvevo r)  
Γ t.tkod max(r.megnevezes),count(*) (befizetes b ⋈ b.kurzus= t.tkod tanfolyam t)  
Γ t.tkod max(r.megnevezes),count(b.kurzus) (befizetes b ⋈ + b.kurzus= t.tkod tanfolyam t)  
Π megnevezes ( ⋈ tkod not ∈ (Π {kurzus} (befizetes)) (tanfolyam))  
Γ t.tkod max(r.megnevezes) mn,count(*) db (befizetes b ⋈ b.kurzus= t.tkod tanfolyam t)  
Π mn ( ⋈ db = ( Γ {max(db)} (X) ) (X) )
```

## 6. feladat

**Mentés: neptunkod\_7.6.txt 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**

```
CREATE TABLE termek (tkod INT PRIMARY KEY,  
nev VARCHAR2(50) NOT NULL,  
ear INT CHECK (EAR > 0),  
kategoria CHAR(20));  
  
INSERT INTO termek VALUES (11,'bab',2000,'Z');  
  
DELETE FROM termek WHERE ear < X;  
  
UPDATE termek SET ear = ear*.9 WHERE kategoria = Y;
```

## **7. feladat**

**Mentés: neptunkod\_7.7.txt 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**

```
CREATE TABLE termekek (tkod INT PRIMARY KEY,  
nev VARCHAR2(50) NOT NULL,  
ear INT CHECK (EAR > 0),  
kategoria CHAR(20));  
  
ALTER TABLE termekek ADD (szin CHAR(20));  
  
DROP TABLE termekek;  
  
DELETE FROM termekek WHERE kategoria IS NULL;
```