

# Adatbázis Rendszerek I.

## BSc

7. gyak

2022. október 25.

**Készítette:**

Martinák Mátyás BSc  
Programtervező informatikus  
alapszak  
KLNSPG

**Miskolc, 2022**

## 1. feladat

Adottak az alábbi tábla:

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

Adja meg az alábbi műveletek relációs algebrai alakját.

1. könyvek címei,
  2. a 2000-nél drágább könyvek adatai,
  3. az 1000-nél olcsóbb könyvek címei,
  4. a tárgyak nevei,
  5. az 'AB' tárgyú könyvek címe és ára.
1.  $\pi_{\text{cím}}(\text{KÖNYV})$
  2.  $\pi_{\text{isbn, cím, tárgy, ár}}^{\sigma(\text{ár} > 2000)}(\text{KÖNYV})$
  3.  $\pi_{\text{isbn, cím, tárgy, ár}}^{\sigma(\text{ár} < 1000)}(\text{KÖNYV})$
  4.  $\pi_{\text{tárgy}}(\text{KÖNYV})$

5.  $\pi$  cím, ár  
 $\sigma(\text{targy} = \text{'AB'}) (\text{KÖNYV})$

## 2. feladat

Adottak az alábbi tábla:

KÖNYV (isbn C(20) PK, cím C(40), targy C(30), ár INT)  
 Adja meg az alábbi műveletek relációs algebrai alakját.

1. a könyvek darabszáma,
  2. a könyvek átlagára,
  3. a legolcsóbb könyv ára,
  4. az 'AB' kategóriájú könyvek darabszáma,
  5. a legdrágább AB kategóriájú könyv ára,
  6. az átlagárnál drágább könyvek címei
  7. az átlagárnál drágább könyvek darabszáma.
1.  $\pi$  COUNT(isbn) (KÖNYV)
  2.  $\pi$  AVG(ar) (KÖNYV)
  3.  $\Gamma$  ar ASC (KÖNYV)
  4.  $\pi$  COUNT(targy)  
 $\sigma(\text{targy} = \text{'AB'}) (\text{KÖNYV})$
  5.  $\Gamma$  ar ASC  
 $\sigma(\text{targy} = \text{'AB'}) (\text{KÖNYV})$
  6.  $\pi$  AVG(ar)  $\rightarrow$  avg, cím  
 $\sigma(\text{ar} > \text{avg}) (\text{KÖNYV})$
  7.  $\pi$  AVG(ar)  $\rightarrow$  avg, COUNT(ar)  
 $\sigma(\text{ar} > \text{avg}) (\text{KÖNYV})$

### 3. feladat

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.

1. az X kategóriájú termékek neve és kódja
  2. a termékek neve és a gyártó neve együtt
  3. az X kategóriájú termékek gyártóinak nevei
  4. az Y-nál drágább termékek darabszáma
  5. az Y-nál drágább termékek gyártóinak darabszáma
1.  $\pi_{\text{nev, tkod}}(\text{termek})$   
 $\sigma(\text{kategória} = 'X')$
  2.  $\pi_{\text{termek.nev, gyarto.nev}}$   
 $(\text{termek} \bowtie \text{termek.tkod} = \text{gyarto.adoszam gyarto})$
  3.  $\pi_{\text{termek.nev, gyarto.nev}}$   
 $\sigma(\text{kategória} = 'X')(\text{termek} \bowtie \text{termek.tkod} = \text{gyarto.adoszam gyarto})$
  4.  $\pi_{\text{COUNT(tkod)}}$   
 $\sigma(\text{ear} > 'Y')$
  5.  $\pi_{\text{COUNT(termek.tkod), COUNT(gyarto.adoszam)}}$   
 $\sigma(\text{ear} > 'Y')(\text{termek} \bowtie \text{termek.tkod} = \text{gyarto.adoszam gyarto})$

## 4. feladat

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, alkat-  
rész INT REFERENCES  
alkatrész);
```

Adja meg az alábbi műveletek relációs algebrai alakját!

1. kategóriánként a termékek átlagára
2. termékenként az alkatrészek darabszáma (termékkód, darabszám)
3. termékenként az alkatrészek darabszáma (terméknév, darabszám)
4. az X nevű termékben szereplő alkatrészek listája
5. azon kategóriák, ahol a termékek átlagára nagyobb mint X

1.  $\Gamma$  AVG(ar), kategoria(termek)
2.  $\Gamma$  tkod, COUNT(tkod)(termek)
3.  $\Gamma$  nev, COUNT(tkod)(termek)
4.  $\Pi$  alkatresz.akod, alkatresz.nev  
 $\sigma(\text{kategoria} = 'X')(\text{termek} \bowtie \text{termek.tkod} = \text{komponens.termek}$   
 $\text{komponens} \bowtie \text{komponens.alkatresz} = \text{alkatresz.akod})$

5.  $\pi$  AVG(ar)  $\rightarrow$  avg, kategoria  
 $\sigma(\text{ar} > \text{avg})(\text{termek})$

## 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:

1. a befizetések összege Ft-ban és Euróban
  2. a befizetések összege résztvevőnként (név) egy adott tanfolyamra
  3. mennyien vesznek részt az egyes tanfolyamokon
  4. mennyien vesznek részt az egyes tanfolyamokon, a 0 érték is jelenjen meg
  5. mely tanfolyamokra nem jelentkeztek még
  6. a legtöbb jelentkezőt vonzó tanfolyam neve
1.  $\pi$  befizetes, befizetes \* 410(befizetes)
  2.  $\pi$  SUM(befizetes.befizetes), résztvevo.nev  $\gamma$  tanfolyam.tipus  
(befizetes  $\infty$  befizetes.diak = résztvevo.tajszam résztvevo)  
(befizetes  $\infty$  befizetes.kurzus = tanfolyam.tkod)
  3.  $\pi$  SUM(resztvevo), tanfolyam.tipus  
(befizetes  $\infty$  befizetes.kurzus = tanfolyam.tkod tanfolyam)  
(befizetes  $\infty$  befizetes.diak = résztvevo.tajszam résztvevo)

4.  $\Pi$  SUM(resztvevo), tanfolyam.tipus  
 (befizetes  $+\infty$  befizetes.kurzus = tanfolyam.tkod +tanfolyam)  
 (befizetes  $+\infty$  befizetes.diak = resztvevo.tajszam +resztvevo)
5.  $\Pi$  tanfolyam.tipus  
 $\sigma$  (tanfolyam.tipus IS NULL) (befizetes  $\infty$  befizetes.kurzus = tanfolyam.tkod  
 tanfolyam)  
 (befizetes  $\infty$  befizetes.diak = resztvevo.tajszam resztvevo)
6.  $\Pi$  tanfolyam.tipus  
 (befizetes  $\infty$  befizetes.kurzus = tanfolyam.tkod tanfolyam)  
 (befizetes  $\infty$  befizetes.diak = resztvevo.tajszam resztvevo)

## 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:

1. új rekord felvitele
  2. az X-nél olcsóbb termékek törlése
  3. az Y kategóriájú termékek árának csökkentése 10%-kal
1. INSERT INTO termek(3, 'Mizo tej', 2, 'Tejtermek');
  2. DELETE FROM termek  
WHERE ear < 'X';
  3. UPDATE TABLE termek SET ar = ar \* 0.1  
WHERE kategoria = 'Y';

## 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:

1. új **szín** mező felvitele a sémába
  2. a termek tábla megszüntetése
  3. azon rekordok megszüntetése, ahol nincs kitöltve a kategória
1. ALTER TABLE termek ADD szín VARCHAR(30);
  2. DELETE FROM termek;
  3. DELETE FROM  
WHERE kategoria IS NULL;