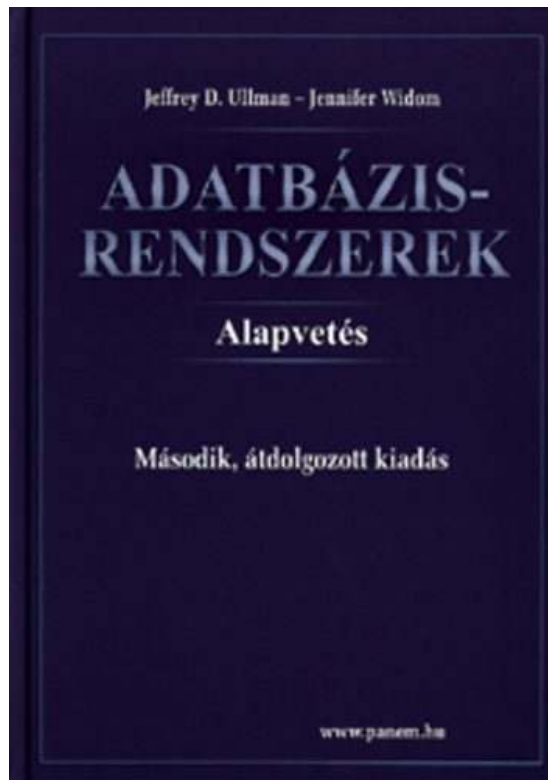


# Az SQL adatbázisnyelv 5.rész



Ullman-Widom: Adatbázisrendszerek  
Alapvetés

Második, átdolgozott kiadás, Panem,  
2009

6.5. Az adatbázis tartalmának  
módosítása (DML utasítások)  
INSERT, DELETE, UPDATE

# SQL - Előzmények

- 2.3. Bevezetés az SQL-be, relációsémák definiálása
- 6.1. Egyszerű lekérdezések az SQL-ben
- 6.2. Több relációra vonatkozó lekérdezések  
Halmaz és multihalmaz műveletek

(SFW)

[UNION | INTERSECT | EXCEPT [ALL] ]

(SFW)

- 6.3. Alkérdezések, korrelált alkérdezések
- 6.4. Relációra vonatkozó műveletek

SELECT DISTINCT ...

FROM ...

[WHERE ...]

[GROUP BY ... [HAVING ...]]

[ORDER BY]

---

# Lekérdezések (később folytatjuk)

- SELECT kiértékelési és hatékonysági kérdésekre köv.félévben Adatbázisok-2 kurzuson fogunk kitérni.
- SELECT-re később a PSM építkezésénél ebben az Adatbázisok-1 kurzusban a **rekurziónál visszatérünk**
- A gyakorlaton vesszük a **hierarchikus** adatszerkezetek lekérdezését az Oracle-ben a CONNECT BY PRIOR
- Most **a tartalom módosító utasításokat** nézzük meg, a gyakorlaton ezt csak az I.ZH után fogjuk kipróbálni.

DML (Data Manipulation Language)

INSERT, DELETE, UPDATE

---

# Adatbázis tartalmának módosítása

- **A módosító utasítások** nem adnak vissza eredményt, mint a lekérdezések, hanem az adatbázis tartalmát változtatják meg.
- 3-féle módosító utasítás létezik:
  - INSERT** - sorok beillesztése, beszúrása
  - DELETE** – sorok törlése
  - UPDATE** – sorok komponensei értékeinek módosítása

---

# Beszúrás

- Ha egyetlen sort szűrünk be:

```
INSERT INTO <reláció>
```

```
VALUES ( <attribútum lista> );
```

- **Példa:** a **Kedvel(név, sör)** táblában rögzítjük, hogy Zsu szereti a Bud sört.

```
INSERT INTO Likes
```

```
VALUES ( 'Zsu' , 'Bud' ) ;
```

# Attribútumok megadása

- A reláció neve után megadhatjuk az attribútumait.
- Ennek alapvetően két oka lehet:
  1. elfelejtettük, hogy a reláció definíciójában, milyen sorrendben szerepeltek az attribútumok.
  2. Nincs minden attribútumnak értéke, és azt szeretnénk, ha a hiányzó értékeket NULL vagy default értékkel helyettesítenék.

Példa:

```
INSERT INTO Kedvel (sör, név)  
VALUES ( 'Bud' , 'Zsu' ) ;
```

---

# Default értékek megadása

- A CREATE TABLE utasításban az oszlopnevet **DEFAULT** kulcsszó követheti és egy érték.
- Ha egy beszúrt sorban hiányzik az adott attribútum értéke, akkor a default értéket kapja.

## Példa: Default értékek

```
CREATE TABLE Sörivók (  
    név CHAR(30) PRIMARY KEY,  
    cím CHAR(50) DEFAULT 'Sesame St'  
    telefon CHAR(16) );  
INSERT INTO Sörivók(név)  
VALUES ('Zsu');
```

Az eredmény sor:

név	cím	telefon
Zsu	Sesame St	NULL



## Több sor beszúrása

- Egy lekérdezés eredményét is beszúrhatjuk a következő módon alkérdéssel:

```
INSERT INTO <reláció>  
( <alkérdés> );
```

- A **Látogat(név, bár)** tábla felhasználásával adjuk hozzá a **LehetBarát(név)** táblához Zsu „lehetséges barátait”, vagyis azokat a sörivőket, akik legalább egy olyan barát látogatnak, ahova Zsu is szokott járni.  
(megoldás a köv.oldalon)

# Megoldás

```
INSERT INTO LehetBarát
(SELECT I2.név
FROM Látogat I1, Látogat I2
WHERE I1.név = 'Zsu' AND
      I2.név <> 'Zsu' AND
      I1.bár = I2.bár
);
```

(SELECT) a másik sörivó

(FROM) névpárok:  
az első Zsu,  
a második nem Zsu,  
de van olyan bár,  
amit mindketten  
látogatnak.

---

## Tk.Példa INSERT INTO utasításra

- A lekérdezést teljesen ki kell értékelni, mielőtt a sorokat beszúrnánk.
- Tankönyv 6.36 példa: új stúdiók beszúrása

```
INSERT INTO Stúdió (név)
SELECT DISTINCT stúdióNév
FROM Filmek
WHERE stúdióNév NOT IN
      (SELECT név FROM Stúdió);
```

# Törlés

- A törlendő sorokat egy WHERE feltétel segítségével adjuk meg:

```
DELETE FROM <reláció>  
WHERE <feltétel>;
```

- **Példa:**

```
DELETE FROM Kedvel  
WHERE nev = 'Zsu' AND  
sör = 'Bud' ;
```

- **Az összes sor törlése:**

```
DELETE FROM Kedvel ;
```

---

## Példa: Több sor törlése

- A **Sörök(név, gyártó)** táblából töröljük azokat a söroket, amelyekhez létezik olyan sör, amit ugyanaz a cég gyártott.  
(Megoldás a köv.oldalon)

## Példa: Több sor törlése

- A **Sörök(név, gyártó)** táblából töröljük azokat a söroket, amelyekhez létezik olyan sör, amit ugyanaz a cég gyártott.

```
DELETE FROM Sörök s
WHERE EXISTS (
  SELECT név FROM Sörök
  WHERE gyártó = s.gyártó
  AND név <> s.név);
```

(WHERE) azok a sörök, amelyeknek ugyanaz a gyártója, mint az s éppen aktuális sorának, a nevük viszont különböző.

---

# A törlés szemantikája

- Tegyük fel, hogy az Anheuser-Busch csak Bud és Bud Lite söröket gyárt.
- Tegyük fel még, hogy s sorai közt a Bud fordul elő először.
- Az alkérdés nem üres, a későbbi Bud Lite sor miatt, így a Bud törlődik.
- Kérdés, hogy a Bud Lite sor törlődik-e?  
(válasz a köv. oldalon)

---

# A törlés szemantikája

- **Válasz:** igen, a Bud Lite sora is törlődik.
- A törlés ugyanis két lépésben hajtódik végre.
  1. Kijelöljük azokat a sorokat, amelyekre a WHERE feltétele teljesül.
  2. Majd töröljük a kijelölt sorokat.



---

# Módosítás

- Bizonyos sorok bizonyos attribútumainak módosítása.

**UPDATE** <reláció>

**SET** <attribútum értékadások listája>

**WHERE** <sorokra vonatkozó feltétel>;

---

## Példa: Módosítás

- Fecó telefonszámát 555-1212-re változtatjuk (Fecó itt egy sörivó neve):

```
UPDATE Sörivók  
SET telefon = '555-1212'  
WHERE név = 'Fecó' ;
```

---

## Példa: Több sor módosítása

- Legfeljebb 4 dollárba kerülhessenek a sörök:

```
UPDATE Felszolgal  
SET ár = 4.00  
WHERE ár > 4.00;
```

---

# Tk.Példa UPDATE utasításra

- Tankönyv 6.39 példa:

**UPDATE** Gyártáslrányító

SET név = 'lg.' || név

WHERE azonosító IN

(SELECT elnökAzon FROM Stúdió)