

Adatbázisok 1.

SQL bevezetés – 1. rész

Select-From-Where záradékok

Több relációt tartalmazó lekérdezések

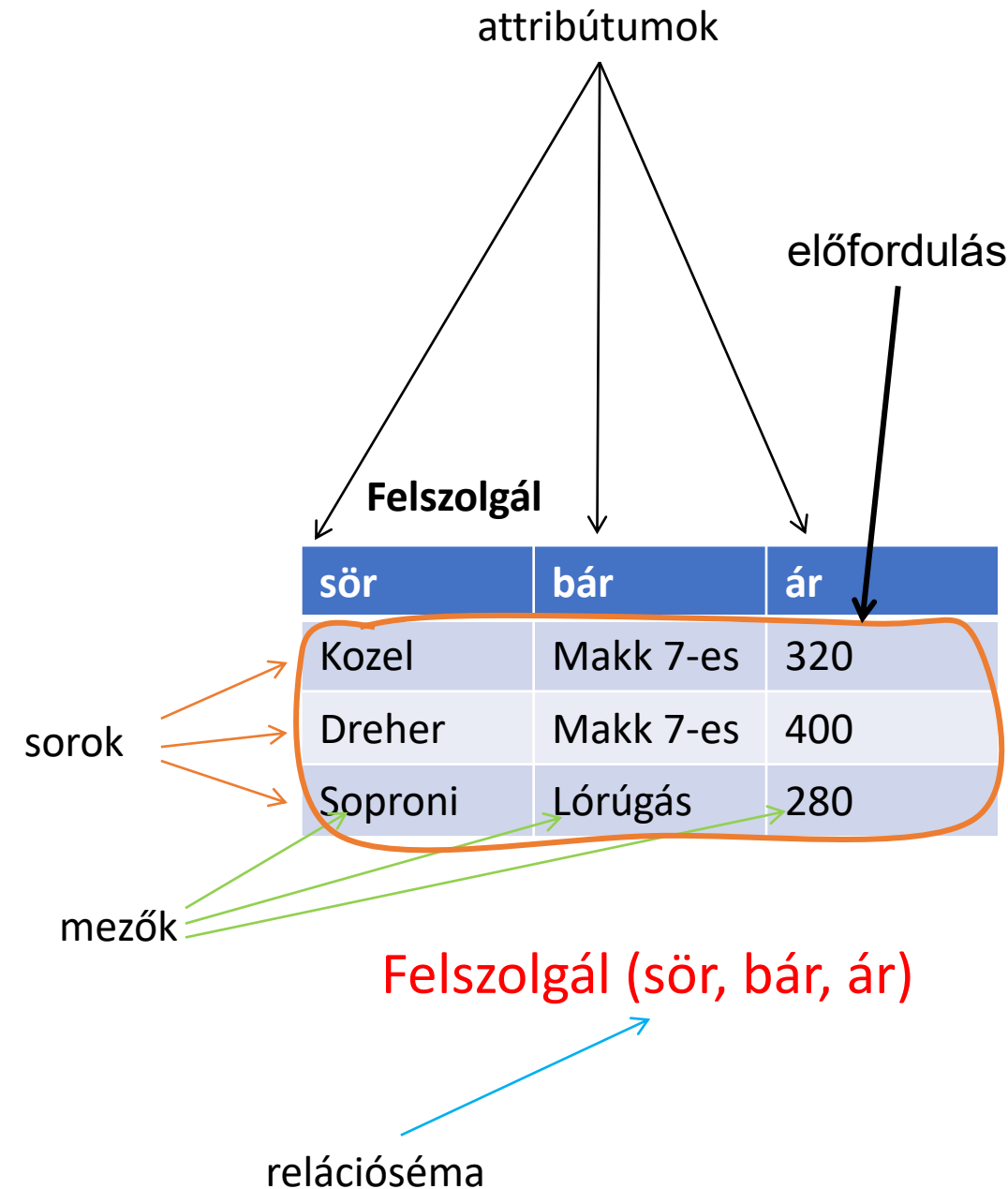
Alkérdezések

Elérhetőségek, információk

- Előadások időpontja, helye:
 - hétfő 8:30-10:00 (1. csop.) és kedd 8:30-10:00 (2. csop.), MS Teams felület
- Előadó
 - Szalai-Gindl János Márk (ELTE IK, Információs Rendszerek Tanszék)
 - Email: szalaigindl@inf.elte.hu
 - Szoba: 2.507 (déli tömb)
 - Honlap: <https://szalaigj.web.elte.hu>
- Az előadás weboldala: <https://canvas.elte.hu/courses/16792>
- Canvas kurzusoknál az előadások diasorai, videólinkjei

Ismétlés I.

- Relációs adatmodell
- Táblák (relációk), attribútumok, sémák, előfordulások, mezők
- Relációs algebra, atomi operandusok és műveletek
- Projekció, szelekció, halmazműveletek, Descartes-szorzat, átnevezés
- Théta- és természetes összekapcsolás
- Műveletek (operátorok) monotonitása
- Megszorítások



Ismétlés II.

- Relációs algebra műveletei:

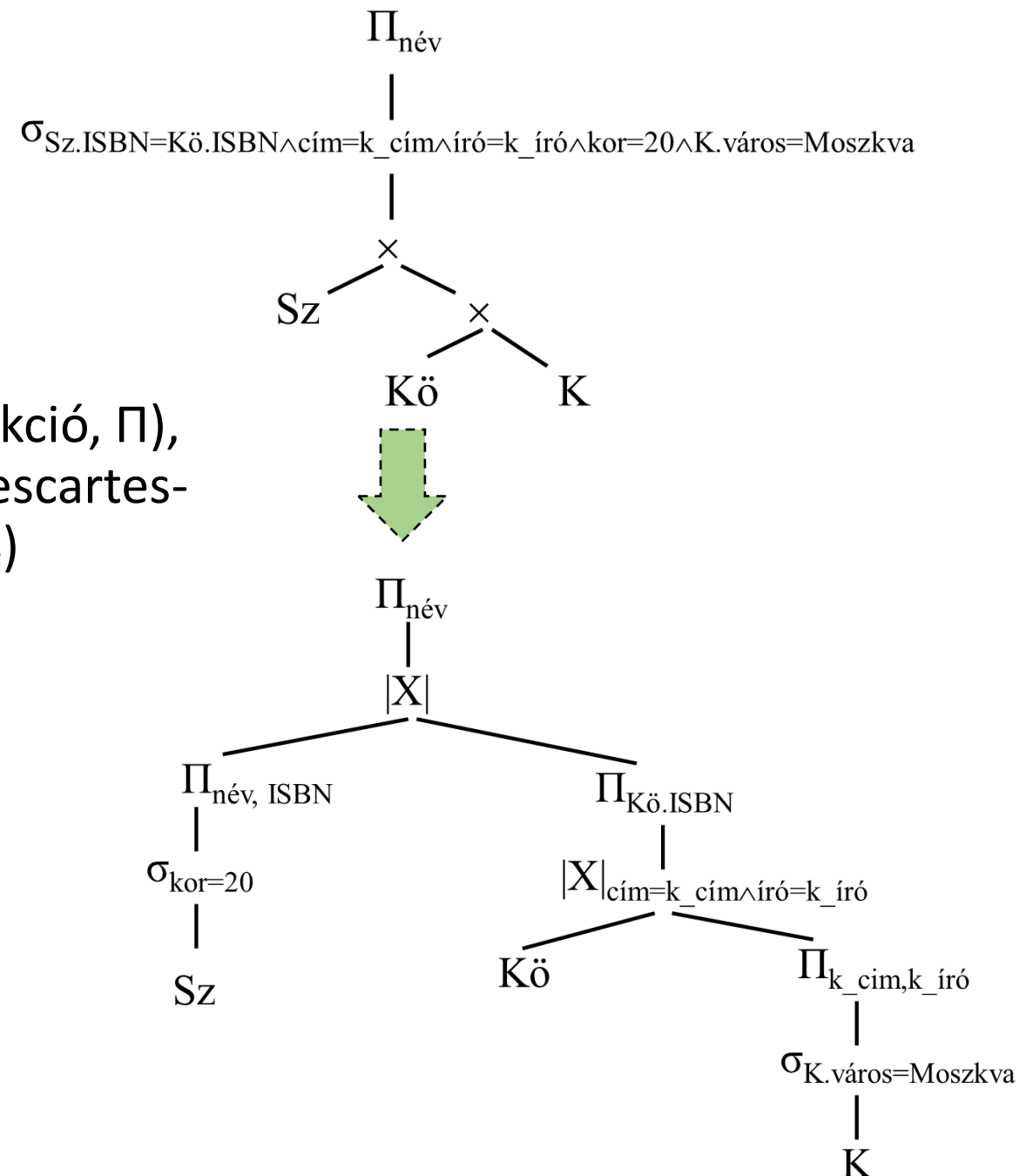
- kiválasztás (szelekció, σ_F), vetítés (projekció, Π),
unió (\cup), metszet (\cap), különbség ($-$), Descartes-
szorzat (\times), összekapcsolások ($|X|$, $|X|_F$)

- Műveletek összeláncolása

- Relációs algebrai lekérdezések

- Ekvivalenciák

- Ekvivalencia-szabályok, átalakítás



Miért az SQL?

- Az SQL magas szintű programozási nyelv.
 - A “hogyan” helyett “mit”
 - Így elkerülünk egy csomó macerát a procedurális nyelvekhez (pl. C++ vagy Java) képest

Miért az SQL?

- Az SQL magas szintű programozási nyelv.
 - A “hogyan” helyett “mit”
 - Így elkerülünk egy csomó macerát a procedurális nyelvekhez (pl. C++ vagy Java) képest
- Az adatbázis-kezelő rendszer kitalálja a leggyorsabb végrehajtási módot.
 - Ezt nevezik “lekérdezés optimalizációnak.”

Select-From-Where záradékok

SELECT az érdekes attribútumok

FROM egy vagy több tábla

WHERE a táblák soraira vonatkozó feltételek

A példa, amit használunk

- Minden SQL lekérdezést a következő adatbázisséma fölött hajtunk végre.
 - Az aláhúzás a kulcsattribútumokat jelöli.

Sörök(név, gyártó)

Kocsmák(név, cím, engedélySzám)

Alkeszek(név, cím, telefon)

Szeret(alkesz, sör)

Felhasználó(kocsmák, sör, ár)

Látogat(alkesz, kocsmák)

Példa

Sörök(név, gyártó)
Kocsmák(név, cím, engedélySzám)
Alkeszek(név, cím, telefon)
Szeret(alkesz, sör)
Felszolgál(kocsma, sör, ár)
Látogat(alkesz, kocsma)

- A Sörök(név, gyártó) táblában mely söröket gyártotta az Anheuser-Busch?

```
SELECT név
```

```
FROM Sörök
```

```
WHERE gyártó = 'Anheuser-Busch';
```

Sörök(név, gyártó)
Kocsmák(név, cím, engedélySzám)
Alkeszek(név, cím, telefon)
Szeret(alkesz, sör)
Felszolgál(kocsma, sör, ár)
Látogat(alkesz, kocsma)

A lekérdezés eredménye

név
Bud
Bud Lite
Michelob
...

Az eredmény egyetlen attribútumot (név) tartalmaz a sorok Anheuser-Busch által gyártott söröket adják.

A lekérdezés jelentése

- Kezdjük a FROM záradékban megadott relációval.

A lekérdezés jelentése

- Kezdjük a FROM záradékban megadott relációval.
- Alkalmazzuk a WHERE záradékban megadott kiválasztási feltételt.

A lekérdezés jelentése

- Kezdjük a FROM záradékban megadott relációval.
- Alkalmazzuk a WHERE záradékban megadott kiválasztási feltételt.
- Levetítjük az eredményt a SELECT záradékban megadott oszlopokra.

Szemantika (a példában)

Sörök(név, gyártó)
Kocsmák(név, cím, engedélySzám)
Alkeszek(név, cím, telefon)
Szeret(alkesz, sör)
Felszolgál(kocsma, sör, ár)
Látogat(alkesz, kocsma)

név	gyártó
Bud	Anheuser-Busch

A *t* sorváltóval
a sorokat vesszük
egymás után.

Szemantika (a példában)

Sörök(név, gyártó)
Kocsmák(név, cím, engedélySzám)
Alkeszek(név, cím, telefon)
Szeret(alkesz, sör)
Felszolgál(kocsma, sör, ár)
Látogat(alkesz, kocsma)

név	gyártó
Bud	Anheuser-Busch

Ellenőrizzük, hogy
Anheuser-Busch-e.

A t sorváltóval
a sorokat vesszük
egymás után.

Sörök(név, gyártó)
Kocsmák(név, cím, engedélySzám)
Alkeszek(név, cím, telefon)
Szeret(alkesz, sör)
Felszolgál(kocsma, sör, ár)
Látogat(alkesz, kocsma)

Szemantika (a példában)

név	gyártó
Bud	Anheuser-Busch

t.név bekerül az
eredménybe, ha igen.

Ellenőrizzük, hogy
Anheuser-Busch-e.

A *t* sorváltóval
a sorokat vesszük
egymás után.

Sörök(név, gyártó)
Kocsmák(név, cím, engedélySzám)
Alkeszek(név, cím, telefon)
Szeret(alkesz, sör)
Felszolgál(kocsma, sör, ár)
Látogat(alkesz, kocsma)

* a SELECT záradékban

- 1 reláció a FROM-ban, a * SELECT-ben az összes attribútum helyett
- Példa: Sörök(név, gyártó):

```
SELECT *
```

```
FROM Sörök
```

```
WHERE gyártó = 'Anheuser-Busch';
```

Sörök(név, gyártó)
Kocsmák(név, cím, engedélySzám)
Alkeszek(név, cím, telefon)
Szeret(alkesz, sör)
Felszolgál(kocsma, sör, ár)
Látogat(alkesz, kocsma)

A válasz:

név	gyártó
Bud	Anheuser-Busch
Bud Lite	Anheuser-Busch
Michelob	Anheuser-Busch
...	...

Azaz a **Sörök** reláció összes attribútuma szerepel.

Sörök(név, gyártó)
Kocsmák(név, cím, engedélySzám)
Alkeszek(név, cím, telefon)
Szeret(alkesz, sör)
Felszolgál(kocsma, sör, ár)
Látogat(alkesz, kocsma)

Attribútumok átnevezése

- Az attribútumok átnevezéséhez “AS <new name>” utasítást használhatjuk.
- Példa: Sörök(név, gyártó):

```
SELECT név AS sör, gyártó
```

```
FROM Sörök
```

```
WHERE gyártó = 'Anheuser-Busch';
```

Az eredmény:

Sörök(név, gyártó)
Kocsmák(név, cím, engedélySzám)
Alkeszek(név, cím, telefon)
Szeret(alkesz, sör)
Felszolgál(kocsma, sör, ár)
Látogat(alkesz, kocsma)

sör	gyártó
Bud	Anheuser-Busch
Bud Lite	Anheuser-Busch
Michelob	Anheuser-Busch
.

Sörök(név, gyártó)
Kocsmák(név, cím, engedélySzám)
Alkeszek(név, cím, telefon)
Szeret(alkesz, sör)
Felszolgál(kocsma, sör, ár)
Látogat(alkesz, kocsma)

A SELECT záradék kifejezései

- Minden kifejezés, ami „értelmesnek tűnik” megjelenhet a SELECT záradékban.
- **Példa:** Felszolgál(kocsm, sör, ár):

```
SELECT kocsm, sör,  
        ár*114 AS árJenben  
FROM Felszolgál;
```

Az eredmény

Sörök(név, gyártó)
Kocsmák(név, cím, engedélySzám)
Alkeszek(név, cím, telefon)
Szeret(alkesz, sör)
Felszolgál(kocsma, sör, ár)
Látogat(alkesz, kocsma)

kocsm	sör	ár
Joe's	Bud	285
Sue's	Miller	342
...

Konstansok

Sörök(név, gyártó)
Kocsmák(név, cím, engedélySzám)
Alkeszek(név, cím, telefon)
Szeret(alkesz, sör)
Felszolgál(kocsma, sör, ár)
Látogat(alkesz, kocsma)

- Szeret(alkesz, sör):

```
SELECT alkész,  
       'szereti a Budot' AS BudIvó  
FROM Szeret  
WHERE sör = 'Bud';
```

Az eredmény

Sörök(név, gyártó)
Kocsmák(név, cím, engedélySzám)
Alkeszek(név, cím, telefon)
Szeret(alkesz, sör)
Felszolgál(kocsma, sör, ár)
Látogat(alkesz, kocsma)

alkesz	Budlvó
Sally	szereti a Budot
Fred	szereti a Budot
...	...

Információ integráció

- Adatbázisok sok forrásból épülhetnek fel (adattárházak).
- Tfh. minden kocsmának van egy saját **Menü(sör, ár)** táblája.
- A **Felhasznál(kocsmá, sör, ár)** tábla elkészítéséhez minden ilyen táblát fel kell dolgoznunk és a kocsmá nevét konstansként kell beszúrnunk.

Információ integráció --- (2)

- Pl. Joe bárjánál:

```
SELECT 'Joe bárja' AS koccsma, sör, ár  
FROM Menü;
```

Összetett feltételek a WHERE záradékban

- Logikai műveletek: AND, OR, NOT.
- Összehasonlítások =, <>, <, >, <=, >=.

Sörök(név, gyártó)
Kocsmák(név, cím, engedélySzám)
Alkeszek(név, cím, telefon)
Szeret(alkesz, sör)
Felszolgal(kocsma, sör, ár)
Látogat(alkesz, kocsma)

Példa összetett feltételre

- A Felszolgal(kocsm, sör, ár) táblában keressük meg Joe bárjában mennyit kérnek a Bud sörért:

```
SELECT ár
FROM Felszolgal
WHERE kocsm = 'Joe bárja' AND
       sör = 'Bud' ;
```

Minták

- A feltételekben a szavakat mintákra illeszthetjük
 - <Attribútum> LIKE <minta> vagy <Attribútum> NOT LIKE <minta>
- *Minta* aposztrófok közötti szöveg az alábbi jelekkel: % = “akármennyi karakter”; _ = “tetszőleges karakter, pontosan egy.”

Sörök(név, gyártó)
Kocsmák(név, cím, engedélySzám)
Alkeszek(név, cím, telefon)
Szeret(alkesz, sör)
Felszolgál(kocsma, sör, ár)
Látogat(alkesz, kocsma)

Példa: LIKE

- Az **Alkeszek(név, cím, telefon)** keressük a budapestieket.

```
SELECT név  
FROM Alkeszek  
WHERE cím LIKE '%Budapest%';
```

Sörök(név, gyártó)
Kocsmák(név, cím, engedélySzám)
Alkeszek(név, cím, telefon)
Szeret(alkesz, sör)
Felszolgál(kocsma, sör, ár)
Látogat(alkesz, kocsma)

Példa: LIKE

- Az **Alkeszek(név, cím, telefon)** keressük azokat az alkeszeket, akiknek a nevük második karaktere 'a', és a nevük 'y'-ra végződik!

```
SELECT név  
FROM Alkeszek  
WHERE név LIKE '_a%y';
```

NULL értékek

- A sorok mezői az SQL relációkban NULL értékeket is tartalmazhatnak.
- A jelentés a kontextustól függően változhat.
Általában:
 - *hiányzó érték* : pl. nem ismerjük Joe bárja címét.
 - *értelmetlen* : egy szingli esetében a házastárs neve.

NULL összehasonlítás

- Az SQL 3-értékű logikája: TRUE, FALSE, UNKNOWN.
- Egy érték NULL-lal hasonlítva az eredmény UNKNOWN
- Egy sor akkor és csak akkor kerül be az eredménybe, ha a WHERE záradék TRUE értéket ad.

3-értékű logika

- Tegyük fel a következőt: TRUE = 1, FALSE = 0, és UNKNOWN = $\frac{1}{2}$.
- Ekkor: AND = MIN; OR = MAX, NOT(x) = $1-x$.

3-értékű logika

- Tegyük fel a következőt: TRUE = 1, FALSE = 0, és UNKNOWN = $\frac{1}{2}$.
- Ekkor: AND = MIN; OR = MAX, NOT(x) = $1-x$.

- Példa:

$$\begin{aligned}\text{TRUE AND (FALSE OR NOT(UNKNOWN))} &= \text{MIN}(1, \text{MAX}(0, (1 - \tfrac{1}{2}))) \\ &= \text{MIN}(1, \text{MAX}(0, \tfrac{1}{2})) = \text{MIN}(1, \tfrac{1}{2}) = \tfrac{1}{2}.\end{aligned}$$

Sörök(név, gyártó)
Kocsmák(név, cím, engedélySzám)
Alkeszek(név, cím, telefon)
Szeret(alkesz, sör)
Felszolgál(kocsma, sör, ár)
Látogat(alkesz, kocsma)

Meglepetés!

- Az alábbi Felszolgál tábla esetén:

kocsm	sör	ár
Joe bárja	Bud	NULL

SELECT kocsm

FROM Felszolgál

WHERE ár < 2.00 OR ár >= 2.00;

Sörök(név, gyártó)
Kocsmák(név, cím, engedélySzám)
Alkeszek(név, cím, telefon)
Szeret(alkesz, sör)
Felszolgál(kocsma, sör, ár)
Látogat(alkesz, kocsma)

Meglepetés!

- Az alábbi Felszolgál tábla esetén:

kocsm	sör	ár
Joe bárja	Bud	NULL

SELECT kocsm

FROM Felszolgál

WHERE ár < 2.00 OR ár >= 2.00;

←→
UNKNOWN

←→
UNKNOWN

Sörök(név, gyártó)
Kocsmák(név, cím, engedélySzám)
Alkeszek(név, cím, telefon)
Szeret(alkesz, sör)
Felszolgál(kocsma, sör, ár)
Látogat(alkesz, kocsma)

Meglepetés!

- Az alábbi Felszolgál tábla esetén:

kocsm	sör	ár
Joe bárja	Bud	NULL

SELECT kocsm

FROM Felszolgál

WHERE ár < 2.00 OR ár >= 2.00;

