

Adatbázisok az iskolában 2012

SQL

Dr. Balázs Péter

Relációs algebra

- *Direkt szorzat*: $T1 \times T2$ ($T1$ minden eleméhez hozzárendeli $T2$ minden elemét)
- *Szelekció*: $\sigma_f(T)$ (T tábla f kifejezésnek eleget tevő sorainak leválogatása)
- *Projekció*: T táblára alkalmazott projekció: $\Pi_J(T)$ (T tábla J halmazban levő oszlopainak leválogatása)

Példa 1.

- **V:** vasarlok(vkod, vnev, vnem(F/N))
- **A:** arucikk(akod, anev, ear)
- **F:** forgalom(*vkod*, *akod*, menny, datum, sszam)
- Mit jelent a $\Pi_J(\sigma_f(V \times A \times F))$ relációs algebrai kifejezés, ha
 - f : $A.akod = F.akod$ AND $V.vkod = F.vkod$
 - J : $vnev, anev, menny * ear$

Példa 2.

- **V:** vasarlok(vkod, vnev, vnem(F/N))
- **A:** arucikk(akod, anev, ear)
- **F:** forgalom(*vkod*, *akod*, menny, datum, sszam)
- Írjuk fel azt a relációs algebrai kifejezést, mely a következő táblát adja: Azon árucikkek nevei és egységárai, melyeket 1996. decemberében női vásárló vett.

Megoldás

- $\Pi_J(\sigma_f(V \times A \times F))$, ahol
- f : $A.akod=F.akod$ AND $V.vkod=F.vkod$
AND $vnem='N'$ AND $datum \geq 1996.12.01$
AND $datum \leq 1996.12.31$
- J : $anev, ear$

SQL lekérdezés

SELECT ... FROM ... WHERE ...

GROUP BY ... HAVING ORDER BY ...

SELECT: Projekció

FROM: Direkt szorzat

WHERE: Szelekció

GROUP BY: Csoportosítás

HAVING: Csoportok közötti válogatás

ORDER BY: Rendezés

SQL – Direkt szorzat

Aruk(akod, nev)

Forgalom(*akod*, menny)

- **SELECT * FROM aruk, forgalom**

SQL – Projekció

Aruk(akod, nev)

Forgalom(*akod*, menny)

- **SELECT** nev, menny **FROM** aruk, forgalom

SQL – Szelekció

Aruk(akod, nev)

Forgalom(*akod*, menny)

- **SELECT** nev, menny **FROM** aruk, forgalom
WHERE aruk.akod=forgalom.akod

SQL – Rendezés

Aruk(akod, nev)

Forgalom(*akod*, menny)

- **SELECT** nev, menny **FROM** aruk, forgalom
WHERE aruk.akod=forgalom.akod **ORDER**
BY nev

SQL – Összesítő függvények

Aruk(akod, nev)

Forgalom(*akod*, menny)

- SUM, AVG, MAX, MIN, COUNT
- SELECT nev, SUM(menny) FROM aruk, forgalom WHERE aruk.akod=forgalom.akod
- **HIBÁS! Meg kell mondani, hogy hogyan csoportosítsunk!**
- SELECT nev, SUM(menny) FROM aruk, forgalom WHERE aruk.akod=forgalom.akod GROUP BY aruk.akod, **nev**
- A csoportosításba minden olyan attribútumot be kell venni, amire nem alkalmaztuk az összesítőfüggvényt (itt: nev).

SQL – Csoportok közötti válogatás

Aruk(akod, nev)

Forgalom(*akod*, menny)

- **SELECT** nev, SUM(menny) FROM aruk,
forgalom WHERE
aruk.akod=forgalom.akod GROUP BY
aruk.akod, nev HAVING SUM(menny)>20

További lehetőségek

- Duplikátumok kiszűrése: `SELECT DISTINCT`
- Multihalmaz műveletek: `UNION`, `EXCEPT`, `INTERSECT`

Mintaillesztés feltételben - LIKE

EGYEZÉS JELLEGE	MINTA	EGYEZIK (TRUE EREDMÉNYT AD)	NEM EGYEZIK (FALSE EREDMÉNYT AD)
Több karakter	a*a	aa, aBa, aBBBa	aBC
	ab	abc, AABb, Xab	aZb, bac
Különleges karakter	a[*]a	a*a	aaa
Több karakter	ab*	abcdefg, abc	cab, aab
Egyetlen karakter	a?a	aaa, a3a, aBa	aBBBa
Egyetlen számjegy	a#a	a0a, a1a, a2a	aaa, a10a
Karaktertartomány	[a-z]	f, p, j	2, &
Tartományon kívüli	[!a-z]	9, &, %	b, a
Nem számjegy	[!0-9]	A, a, &, ~	0, 1, 9
Kombinált	a[!b-m] #	An9, az0, a99	abc, aj0

Törlés, beszúrás, módosítás

- Törlés: `DELETE FROM aruk WHERE nev=„Kifli”`
- Beszúrás: `INSERT INTO aruk (akod, nev) VALUES(5, 'tej')`
- Módosítás: `UPDATE aruk SET nev='nagykifli' WHERE nev='kifli'`