

Baze de date

Universitatea “Transilvania” din Brasov

Lect.dr. Costel Aldea
costel.aldea@gmail.com

Baze de date – Lab.08

1. Teorie

- a) Ce este integritatea relationala?
- b) Definiti integritatea referentiala.
- c) Ce este un view?
- d) Descrieti tipuri de chei.
- e) Definire forma normala. Forme normale.
- f) Anomalii de actualizare date.
- g) Dependente functionale.
- h) Axiomele lui Armstrong
- i) Dependentele funcționale și cheile relațiilor
- j) Teorema lui Ullman
- k) Forma normala Boyce-Codd (FN3.5)
- l) Algoritmi de normalizare

2. Interogari (bd. firma)

- 1. Sa se afiseze angajatii care locuiesc in acelasi oras cu firma la care lucreaza
- 2. Sa se afiseze angajatii care locuiesc in acelasi oras, pe aceeași strada cu managerul lor
- 3. Sa se afizeze angajatii care castiga peste medie
- 4. Sa se identifice acele firme la care angajatii castiga in medie un salariu mai mare decat angajatii de la 'FaraSoft SRL'
- 5. Daca firmele sunt localizate in mai multe orase afisati firmele care sunt in acelasi oras cu 'Banca FaraSoft'
- 6. Afisati angajatii care castiga mai mult decat oricare dintre angajatii firmei 'Banca FaraSoft'
- 7. Sa se creeze cate un view pentru interogarile 1-6.

3. Interogări (bd. restaurant)

- a) Creati un view prin care sa afisati lista restaurantelor alaturi de oras.
- b) Creati un view cu lista clientilor si restaurantul la care au fost in ultimele doua luni ordonata alfabetic dupa numele clientului.

4. Proiect

- a) Pentru o relatie din problema modelata propuneti multimea de dependente functionale si aratati in ce forma normala este relatia (slide 32 / curs 09)

bd. firma

- Se da urmatoarea schema de baza de date
 - Employee(employee-name, street, city)
 - Works(employee-name, company-name, salary)
 - Company(company-name, city)
 - Manages(employee-name, manager-name)

Solutii sql

- ❑ SELECT employee-name FROM employee, works WHERE employee.employee-name = works.employee-name AND works.companyname = company.company-name AND employee.city = company.city;
- ❑ SELECT e1.employee-name FROM employee e1, employee e2, manages WHERE e1.employee-name = manages.employee-name AND e2.employee-name = manages.manager-name AND e1.street = e2.street AND e1.city = e2.city;
- ❑ SELECT employee-name FROM works w1, (SELECT AVG(salary) AS avg-salary, company-name FROM works GROUP BY company-name) w2 WHERE w1.company-name = w2.company-name AND w1.salary > w2.avg-salary
- ❑ SELECT company-name FROM works GROUP BY company-name HAVING AVG(salary) > (SELECT AVG(salary) FROM works GROUP BY company-name HAVING company-name = 'FaraSoft SRL')

-
- ❑ SELECT company-name FROM(SELECT c1.company-name,c1.city FROM company c1, company c2 WHERE c1.city=c2.city AND c2.company-name = 'Banca FaraSoft') R GROUP BY company-name HAVING COUNT(DISTINCT city) = (SELECT COUNT(DISTINCT city) FROM company GROUP BY company.company-name HAVING company-name = 'Banca FaraSoft');
 - ❑ Select employee-name From employee Where employee-name NOT IN (SELECT employee-name FROM works w1,works w2 WHERE w1.salary < w2.salary AND w2.companyname = 'Banca FaraSoft');