CS301 Database Management Systems
Spring 2024
Assignment #4

Relational Algebra and Normalization

Due: March 22 at 11:59 pm

Relational Algebra (10 points each) : Specify the following queries using relational algebra.

1. Find the SSN of employees who work for the Headquarters Department.

$\pi_{SSn}(\sigma_{Dname='Head\ Quarters'}(employee \bowtie Department))$

2. Find the Pnumber and Pname of each project controlled by the department with Dname='Research'.

$\pi_{Pnumber, Pname}(\sigma_{Dname='research'}(project \bowtie Department))$

3. For each department, find the dnumber, dname and the number of employees who work for that department.

$\pi_{Dnum, Dname, count(SSn)}(employee \bowtie Department)\ \gamma_{Dnum, Dname; count(SSn)}$

4. Find the number of employees who have a supervisor role.

$\gamma_{count(SSn)}(\sigma_{Supervisor=''}(employee))$

5. For each employee who has worked on the 'Newbenefits' project for more than 10 hours, list Fname, Lname, and Ssn.

$\pi_{Fname, Lname, SSn}(\sigma_{Pname='New\ Benefits' \wedge hours>10}(employee \bowtie works\_on \bowtie project))$

6. List the ssn, lname of each employee who works on both of the projects ProductX and ProductY.

$\pi_{Ssn, Lname}(\sigma_{Pname='ProductX'}(employee \bowtie works\_on \bowtie Project) \cap \sigma_{Pname='ProductY'}(employee \bowtie works\_on \bowtie Project))$

7. Get Ssn and Salary for each employee who has worked on all the projects.

$\pi_{Ssn, salary}(\sigma_{Pcount=totalP}(employee \bowtie (_{Ssn}\gamma_{count(Pnumber)\rightarrow Pcount}(works\_on) \times \gamma_{count(Pnumber)\rightarrow totalP}(Project))))$

8. Find the project number and name for the projects that involve all the employees in the Department of 'Administration.'

$\pi_{Pnum, Pname}(\sigma_{Ecount=Dcount}(Project \bowtie (_{Dno}\gamma_{count(Ssn)\rightarrow Dcount}(employee \bowtie (_{Dno}\gamma_{count(ssn)\rightarrow Dcount}(employee \bowtie \sigma_{Dname='administration'}(Department)) \times \gamma_{Pnum, count(Ssn)\rightarrow Ecount}(works\_on))))))$

9 **Normalization (20 points) :For each relation and its related function dependencies below, indicate which normal form (1NF, 2NF or 3NF) it is in, and if it is not in 3NF, decompose it into 3NF relations. The primary key is in bold face.**

a). R(**B, C**, D, E, F, G): BC → D, BC → E, B → F, B → G

We need to remove transitive dependency due to B → F, B → G so;

R1(B ,C , D, E) so

BC → D
BC → E
R2(B,F,G) so
B → F
B → G
R1,R2 are both in form of 3NF

b). R(**B**, **C**, D, E, F, G): BC → D, BC → E, D → F, E → G
R is in the form of 3NF