CS 540- Database Management Systems
Assignment 1

Submitted by :

Name: Rahul Kumar Nalubandhu , Disha Kanavikar
Email: nalubanr@oregonstate.edu , kanavikd@oregonstate.edu
ONID: nalubanr, kanavikd

Question

Consider the following relational schema:

Emp (eid: integer, ename: string, age: integer, salary: real)
Works (eid: integer, did: integer, pc_time: integer)
Dept (did: integer, dname: string, budge: real, managerid: integer)

Q1. Return names of every employee who works in the "Hardware", "Software", and "Research" departments. (1.5 points)

Answer

a) Datalog:

Q1(b):- Emp(a, b, c, d), Works(a, x, y), Dept(x, ”Hardware”, w, s)
Q2(b):- Emp(a, b, c, d), Works(a, x, y), Dept(x, ”Software”, w, s)
Q3(b):- Emp(a, b, c, d), Works(a, x, y), Dept(x, ”Research”, w, s)
Q(b):- Q1(b), Q2(b), Q3(b)

b) Relational Algebra:

$$\pi_{\text{ename}}\left(\sigma_{\text{dname="Hardware"}}\left(\text{Emp} \bowtie_{\text{Emp.eid=Works.eid}} \text{Works} \bowtie_{\text{Works.did=Dept.did}} \text{Dept}\right)\right) \cap$$
$$\pi_{\text{ename}}\left(\sigma_{\text{dname="Software"}}\left(\text{Emp} \bowtie_{\text{Emp.eid=Works.eid}} \text{Works} \bowtie_{\text{Works.did=Dept.did}} \text{Dept}\right)\right) \cap$$
$$\pi_{\text{ename}}\left(\sigma_{\text{dname="Research"}}\left(\text{Emp} \bowtie_{\text{Emp.eid=Works.eid}} \text{Works} \bowtie_{\text{Works.did=Dept.did}} \text{Dept}\right)\right)$$

c) Relational Calculus:

This expression will be like the one in relational algebra:

E.ename | Emp(E) ∧ (∃.Works(W))(∃.Dept(D)) (E.eid = W.eid ∧ D.did = W.did ∧ (D.dname = "Hardware" ∨ D.dname = "Software" ∨ D.dname = "Research"))

Step by Step Explanation:

E.ename | Emp(E)
→ selecting employee name.

∧ (∃.Works(W))(∃.Dept(D))
→ checking if there exists one relation between works and department.

E.eid = W.eid ∧ D.did = W.did
→ joining Emp, Dept, Works relations

∧ (D.dname = "Hardware" ∨ D.dname = "Software" ∨ D.dname = "Research")
→ department name should be hardware or software or research

Q2. Return the names of every department without any employee. (1.5 points)

a) Datalog:

Q1(a, b, c, d):- Dept(a, b, c, d), Works(p, a, q)
Q(b):- Dept(a, b, c, d), not Q1(a, b, c, d)

b) Relational Algebra:

$\pi_{\text{dname}}(\text{Dept}) - \pi_{\text{dname}}(\text{Dept} \bowtie_{\text{Dept.did=Works.did}} \text{Works})$

c) Relational calculus :

¬ (D.dname | Dept(D) ∧ (∃.Works(W))(W.did = D.did))

Step by Step Explanation:

¬
→ negation (selected all the department name which have at least one employee and applied negation – to get department name without any employee)

D.dname
→ select department name.

Dept(D) ∧ (∃.Works(W)
→ join department and works relations.

W.did = D.did
→ did matches in works and department relations

Q.3 Print the managerid of managers who manage only departments with budgets greater than $1.5 million. (1.5 points)

Dept (<u>did</u>, dname, budge, managerid) => Dept(x, y, z, w)

a) Datalog:

Q1(w): - Dept (x, y, z, w), not Q2(w).

Q2(w): - Dept (x, y, z, w), z < = 1500000.

b) Relational Algebra:

ρ(managerid, D.w) (($\pi_{D.w}$(D) - $\pi_{D.w}$ ($\sigma_{D.c} \leq 1500000$ (D)))

c) Relational Calculus :

Q(w) = $\exists_x, \exists_y, \forall_z, (Dept(x, y, z, w) \wedge z > 1500000) \wedge \neg \exists x' \exists y' \exists z' (\text{Dept}(x' y' z', w) \wedge z' \leq 1500000$

<u>Step by Step Explanation:</u>

→ ∃x, ∃y, (Dept(x, y, z, w) ∧ z > 1500000) - Selecting the department with a budget more than 1.5 million for at least one x and one y.

→ ¬∃x', ∃y', ∃z' (Dept(x', y', z', w) ∧ z' ≤ 1500000) - There should not exist any department with a budget less than or equal to 1.5 million managed by the same manager (w).

Combining the two expressions, we ensure that we only select manager IDs who manage only departments with budgets greater than $1.5 million.

Q.4 Print the name of employees whose salary is less than or equal to the salary of every employee. (1.5 points)

a) DataLog:

Q(b) :- Emp(a, b, c, d), not Q1(b).
Q1(b):- Emp(a, b, p, d), p > c.

b) Relational Algebra:

$$\pi_{ename}(E) - \pi_{ename}(\sigma_{E.salary<E1.salary(E \bowtie E1)})$$

c) Relational Calculus:

Q(q) = ∃p.∃r.∃s.Emp(p,q,r,s) ∧ ∀t.∀u(Emp(p,q,r,t) ∧ Emp(p,q,r,u) => t <= u)

Step by Step Explanation:

∃p.∃r.∃s.Emp(p,q,r,s)
→ Existential (At least one) from the employee relation

∧ ∀t.∀u
→ p,q,r,s is either in selection or in the existential places. t and u are in "for all" condition.

Emp(p,q,r,t) ∧ Emp(p,q,r,u)
→ self-join of employee relation with difference in the amount

=> t <= u
→ "=>" is the "if-then" condition where it checks the salary is less than or equal to every employee.