CMPSC 431W - Homework 2.

Yanjun Chen yfc5289

1. Declarative query language: describe what you want rather than how you compute it. Relational calculus query is declarative query language.

   Procedural query language: used to represent execution plans. Relational algebra is procedual query language.

2. Assume cid are unique.

(1.) relational algebra:

$$\pi_{fname, lname, branch\_name, num\_employee}((Branch \bowtie_{Branch.eid = PersonalBanker.eid} PersonalBanker) \bowtie Employee)$$

TRC:

$$\{R \mid \exists e \in Employee, \exists p \in PersonalBanker, \exists b \in Branch(e.eid = p.eid \land b.manager\_eid = e.eid \land R.eid = p.eid \land R.fname = e.fname \land R.lname = e.lname \land R.num\_employee = b.num\_employee \land R.branch\_name = b.branch\_name)\}.$$

(2.) relational algebra:

$$\pi_{fname, lname}(\sigma_{birthday > "06/02/2001" \land income > 10000}(Customer))$$

TRC:

$$\{R \mid \exists p \in Customer(p.birthday > "06/02/2001" \land p.fname = R.fname \land R.lname = p.lname \land p.birthday = R.birthday \land p.income > 100000)\}$$

(3.) relational algebra:

$\pi_{cid, fname, lname, birthday}(\sigma_{num\_employee>25}(Customer \bowtie Owns \bowtie Account \bowtie Branch))$

TRC:

$\{R \mid \exists O \in Owns, \exists A \in Account, \exists B \in Branch, \exists C \in Customer(B.num\_employee > 25 \wedge A.accno = O.accno \wedge A.branch\_no = B.branch\_no \wedge O.cid = C.cid \wedge R.cid = O.cid \wedge R.fname = C.fname \wedge R.lname = C.lname \wedge R.birthday = C.birthday)\}$

(4.) relational algebra:

$\pi_{cid}(Owns) - \pi_{cid}(\sigma_{owns.cid \neq d.cid \wedge owns.accno = d.accno}(\rho_d(Owns) \times Owns))$

TRC:

$\{R \mid \forall A \in Owns, \nexists B \in Owns(A.accno = B.accno \wedge A.cid \neq B.cid \wedge R.cid = B.cid)\}$

(For partial): Owners that DO have joint account:

$A.accno = B.accno \wedge A.cid \neq B.cid$.

same account number, but not same cid (different person).

(5.) relational algebra: # I considered "Pittsburgh West" as branch_name. But I think it can also be street

$\pi_{cid}((\sigma_{branch\_name = \text{"Pittsburgh West"}}(Branch \bowtie Employee \bowtie Personal\ Banker)) \cap (\sigma_{branch\_name = \text{"Harrisburg South"}}(Branch \bowtie Employee \bowtie Personal\ Banker)))$

intersect

TRC: same cid (unique). $P_1 . P_2$ should be same person { ① P_1 in Pitts. ② P_2 in Harris.

$\{R \mid \exists C \in Customer, \exists P_1 \in Personal\ Banker, \exists P_2 \in Personal\ Banker, \exists E_1 \in Employee, \exists E_2 \in Employee, \exists B_1 \in Branch, \exists B_2 \in Branch (C.cid = P_1.cid \wedge P_1.cid = P_2.cid \wedge P_1.eid = E_1.eid \wedge E_1.branch\_no = B_1.branch\_no \wedge B_1.branch\_name = \text{"Pittsburgh West"} \wedge P_2.eid = E_2.eid \wedge E_2.branch\_no = B_2.branch\_no \wedge B_2.branch\_name = \text{"Harrisburgh South"} \wedge R.cid = C.cid)\}$

get the cids as required.

(6.) relational algebra:

$$\pi_{branch\_name}(\sigma_{lname = \text{"Jackson"}}(Employee \bowtie Branch) \cap \sigma_{salary < 20000}(Employee \bowtie Branch))$$

TRC:

$$\{R \mid \exists E_1 \in Employee, \exists E_2 \in Employee, \exists B \in Branch(E_1.branch\_no = E_2.branch\_no \wedge E_1.lname = \text{"Jackson"} \wedge E_2.salary < 20000 \wedge E_1.branch\_no = B.branch\_no \wedge R.branch\_name = B.branch\_name)\}$$

work in the same branch.

no matter if $E_1 = E_2$.

(7.) relational algebra:

$$\pi_{cid}(\sigma_{txn\_type = \text{"check"} \wedge |amount| > 10000}(Transaction \bowtie Owns))$$

absolute value.

TRC:

$$\{R \mid \exists T \in Transaction, \exists O \in Owns(T.txn\_type = \text{"check"} \wedge T.accno = O.accno \wedge |T.amount| > 10000 \wedge R.cid = O.cid)\}$$

absolute value

(8.) relational algebra:

select all custome meet the birthday requirement.

$$\pi_{fname, lname, birthday}(\sigma_{birthday < \text{"08/21/1950"}}(Customer) - \sigma_{A.income > B.income}(\rho_A(\sigma_{birthday < \text{"08/21/1950"}}(Customer)) \times \rho_B(\sigma_{birthday < \text{"08/21/1950"}}(Customer))))$$

all possible combination of customer

(8.) TRC =

$\{R \mid \exists C_1 \in Customer, \forall C_2 \in Customer (C_1.birthday < "08/21/1950" \wedge$
$\underline{C_2.birthday < "08/21/1950" \wedge C_1.income < C_2.income} \wedge$
$R.fname = C_1.fname \wedge R.lname = C_1.lname \wedge$
$R.birthday = C_1.birthday)\}$

select all (by $\forall$) customer with required birthday.

(9.) relational algebra:

$\pi_{cid, salary}(\sigma_{Employee.salary > M.salary}(Employee \bowtie (\rho_M(Employee \bowtie Branch)$
$\bowtie_{eid = manager\_eid} Employee)))$

TRC = regular employee, manager

$\{R \mid \exists \underline{E \in Employee}, \exists \underline{M \in Employee}, \exists B \in Branch (E.branch\_no =$
$M.branch\_no \wedge M.eid = B.manager\_eid \wedge E.salary > M.salary \wedge$
$R.eid = E.eid \wedge R.salary = E.salary)\}$

get full info   get "real" id.   get id of "fake info".   all possible combination (with wrong info)

10. relational algebra:

$\pi_{fname, lname, income}((\pi_{cid}(Customer) - \pi_{cid}(\pi_{cid,branch\_no}(\underline{Customer \times Branch})$
$\underline{\pi_{cid,branch\_no}(Owns \bowtie Account)})) \bowtie Customer)$

that actual have the account

"fake info".

TRC =

$\{R \mid \exists C \in Customer, \exists O \in Owns, \exists A \in Account, \forall B \in Branch$
$(C.cid = O.cid \wedge O.accno = A.accno \wedge A.branch\_no = B.branch\_no$
$\wedge R.fname = C.fname \wedge R.lname = C.lname \wedge R.income = C.income)\}$

11. relational algebra:

$$\pi_{cid}(Customer) - \pi_{cid}\Big(\pi_{cid,branch\_no}\big(\underbrace{Customer \times (\sigma_{20000<budget<200000}(Branch))}_{\text{all possible combination.}}\big) - \big(\underbrace{\pi_{cid,branch\_no}(Owns \bowtie Account)}_{\text{all cid + branch\_no that do exist}}\big)\Big)$$

TRC =

$$\{R \mid \exists C \in Customer, \exists O \in Owns, \exists A \in Account, \forall B \in Branch\big(\underline{(20000 < B.budget < 200000 \rightarrow (C.cid = O.cid \land O.accno = A.accno \land A.branch\_no = B.branch\_no))} \land R.cid = C.cid\big)\}$$