Tutorial 1: Solutions

Nathan Sharp | s1869292

Schema :

Customer C : (k) ID, Name, City
Account A: (k) Number, Branch, CustId, Balance

Problem 1

1) Return the (value of attribute) *Number* of all accounts owned by customers called "John Doe".

```
SELECT Account.Number
FROM Customer, Account
WHERE Customer.ID = Account.CustID
AND Customer.Name = 'John Doe';
```

2) Return the *Number* and *Branch* of all accounts owned by a customer with *ID* "xyz123", only if the customer there is such a customer in the Customer table

```
SELECT Account.Number, Account.Branch
FROM Customer, Account
WHERE Customer.ID = Account.CustID
AND Customer.Id ='xyz123'
```

3) Return the *Number* and *Balance* of all overdrawn accounts in the "London" branch.

```
SELECT Account.Number, Account.Balance
FROM Account
WHERE Account.Branch = 'London'
AND Account.Balance < 0
```

4) Return all pairs (*Name,Number*) where *Name* is the name of a customer and *Number* is the number of an account owned by that customer, such that the branch of the account is in a different city than the one where the customer lives.

```
SELECT Customer.Name, Acccount.Number
FROM Customer, Account
WHERE Customer.ID = Account.CustID
AND Customer.City != Account.Branch
```

Problem 2

Write the following queries in relational algebra

1) *ID and name of customers who own an account in a branch in their city*

$$\pi_{(C.Id,C.Name)}\sigma_{(C.City=A.Branch \wedge C.Id=A.CustId)}(C \times A)$$

2) *ID and name of customers who do not own any account*

$$\pi_{(C.Id,C.Name)}(C) - \pi_{(C.Id,C.Name)}\sigma_{(C.Id=A.Cust)}(C \times A)$$

3) *ID and name of customers who own an account in every branch*

$$\pi_{C.Id,C.Name}\pi_{A.Branch}(A)$$

4) *ID and name of customers who won an account with a balance which is no less than the balance of an other account*

$\pi_{(C.Id,C.Name)}\sigma_{(C.Id=A.CustId)}(C\times$
$(A - \rho_{(A.Balance2\rightarrow A.Balance)}(\pi_{(A.CustID,A.Balance2)}(\rho_{(A.Balance\rightarrow A.Balance1)}(A) \bowtie_{(A.Balance1\leq A.balance2)}$
$\rho_{(A.Balance\rightarrow A.Balance2)}(A)))))$

Problem 3

Can query (4)) of Problem 2 ever return more than one tuple? If yes show a database (over the given schema) on which that happens; otherwise, explain why in cannot happen.

Yes, If two customers have equal highest account balances.

Problem 4

Compute a query on a given database

Id	Name	City
1	John	London
3	Jeff	London