# CS143: Database Systems
# Homework #1 SOLUTION

1. $(R - S) \cup (S - R)$ is:

| A | B | C |
|---|---|---|
| 1 | 2 | 6 |
| 2 | 5 | 4 |
| 4 | 5 | 6 |

2. $R \bowtie_{R.A<S.C \wedge R.B<S.D} S$ is:

| A | R.B | S.B | C | D |
|---|---|---|---|---|
| 1 | 2 | 2 | 4 | 6 |
| 3 | 4 | 2 | 4 | 6 |
| 1 | 2 | 8 | 6 | 8 |
| 3 | 4 | 8 | 6 | 8 |
| 5 | 6 | 8 | 6 | 8 |
| 1 | 2 | 7 | 5 | 9 |
| 3 | 4 | 7 | 5 | 9 |

3. (a)

$$\pi_{customer-name}(\sigma_{branch-name='Region12'}(Account))$$

(b)

$$\pi_{customer-name}(\sigma_{A.city<>B.city \wedge A.branch-name=B.branch-name}(\rho_B(Branch)\times \\ \rho_A(Customer \bowtie Account)))$$

(c)

$$\pi_{branch-name}(Branch) - \pi_{branch-name}(Account)$$

(d)

$$\pi_{customer-name}(Customer) - \pi_{customer-name}(\sigma_{branch-name='Region12'}(Account))$$

(e)

$$\pi_{customer-name}(Customer) - \\ \pi_{customer-name}(\pi_{customer-name}(Customer) \times \pi_{branch-name}(\sigma_{city='LosAngeles'}(Branch)) - \\ \pi_{customer-name,branch-name}(Account))$$

(f)

$$\pi_{customer-name}(Customer) - \\ \pi_{A.customer-name} \\ (\sigma_{A.branch-name<>B.branch-name \vee A.account-number<>B.account-number) \wedge A.customer-name=B.customer-name} \\ (\rho_A(Account) \times \rho_B(Account)))$$

4. $\pi_{sid}(Student) - \pi_{A.sid}(\sigma_{A.GPA>B.GPA \wedge A.sid<>B.sid}(\rho_A(Student) \times \rho_B(Student)))$

5. *Write the querie of Exercises 3(e) without using division*

$$\pi_{customer-name}(Customer) - \pi_{customer-name}(\pi_{customer-name}(Customer) \times \pi_{branch-name}(\sigma_{city='LosAngeles'}(Branch)) - \pi_{customer-name,branch-name}(Account))$$

6. *How did relational division (div) get that name?*

**Answe: Let** R(A, B) div S(B) = Q(A). Then Q(A) is the largest (w.r.t. set ordering) relation that satisfies the following property: Q(A) x S(B) is a subset of R(A, B).

This is similar to the integer division operator Z= X div Y, where given two integers X and Y, t their quotient Z is the largest integer where Y x Z <= X.

*[SQL] 4. The relation* ***Student(sid, GPA)*** *captures the student-GPA information, where* ***sid*** *is the id of a student and GPA is the student's GPA. Write a relational algebra that finds the ids of the students with the lowest GPA.*

*(Hint: When a query is difficult to write, think of its complement.)*

| **EXCEPT** | **NOT IN** |
|---|---|
| SELECT sid<br>FROM Student<br>EXCEPT<br>SELECT A.sid<br>FROM Student A,<br>    Student B<br>WHERE A.GPA > B.GPA<br>  AND A.sid <> B.sid | SELECT DISTINCT sid<br>FROM Student<br>WHERE sid NOT IN<br>   (SELECT A.sid<br>    FROM Student A,<br>        Student B<br>    WHERE A.GPA > B.GPA<br>      AND A.sid <> B.sid) |

This is much easier with aggregates.

```
SELECT sid
FROM Student
WHERE GPA = (SELECT MIN(GPA)  FROM Student )
```