Warmup #1  (Textbook Problem 6.5)

Let the following relational schemas be given:

\[ R = (A, B, C) \]
\[ S = (D, E, F) \]

Let relations \( r(R) \) and \( s(S) \) be given. Give an expression in the tuple relational calculus that is equivalent to each of the following:

a. \( \Pi_A(r) \)

b. \( \sigma_{B=17}(r) \)

c. \( r \times s \)

d. \( \Pi_{A,F}(\sigma_{C=D}(r \times s)) \)

Warmup #2

Consider the following database:

\[ \text{employee} = (\text{person\_name}, \text{street}, \text{city}) \]
\[ \text{works} = (\text{person\_name}, \text{company\_name}, \text{salary}) \]
\[ \text{company} = (\text{company\_name}, \text{city}) \]
\[ \text{manages} = (\text{person\_name}, \text{manager\_name}) \]

Give expressions in tuple relational calculus and SQL for each of the following queries:

a. Find the names of all employees who work for First Bank Corporation.

b. Find the names and cities of residence of all employees who work for First Bank Corporation.

c. Find the names, cities of residence, and street address of all employees who work for First Bank Corporation and earn more than $10,000.

d. Find the names of all employees in this database who live in the same city as that in which the company for which they work is located.

Warmup #3  (Textbook Problem 3.6)

The SQL like operator is case sensitive, but the lower() function on strings can be used to perform case insensitive matching. To show how, write a query that finds departments whose names contain the string sci as a substring, regardless of the case. Assume that the relation name is “department” and the attribute name is “dept\_name”.

1
**Warmup #4  (Textbook Problem 3.7)**

Consider the SQL query

```sql
select distinct p.a1
from p, r1, r2
where p.a1 = r1.a1 or p.a1 = r2.a1
```

Under what conditions does the preceding query select values of p.a1 that are either in r1 or in r2? Examine carefully the cases where one of r1 or r2 may be empty.

**Problem 5 (To Be Graded)**

Consider the following simplified university registrar database:

<table>
<thead>
<tr>
<th>Student</th>
<th>Course</th>
<th>Enrollment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name</td>
<td>Title</td>
</tr>
<tr>
<td>Amy</td>
<td>CS33</td>
<td>2015F</td>
</tr>
<tr>
<td>Ben</td>
<td>CS127</td>
<td>2015F</td>
</tr>
<tr>
<td>Carl</td>
<td>CS195</td>
<td>2013F</td>
</tr>
<tr>
<td>Dan</td>
<td>CS127</td>
<td>2014F</td>
</tr>
<tr>
<td>Eliza</td>
<td>CS136</td>
<td>2012S</td>
</tr>
</tbody>
</table>

The keys for each relation are as follows:

- **Student**: Name (all student names are assumed to be unique)
- **Course**: Title and Semester
- **Enrollment**: Name, Title, and Semester

For each of the following, give the equivalent (a) TRC formula and (b) SQL query

1. Which students graduating in 2015 have ever gotten an A in CS127?

2. Which of Professor Doeppner's courses is Eliza taking this semester (2015F)?

3. \( \sigma_{semester=2015F}(Courses) \bowtie Enrollment \bowtie \sigma_{gpa>3.5}(Student) \)

4. \( \pi_{name,title}(\sigma_{gradyear=2015}(Student) \bowtie Enrollment) \div \pi_{title}(\sigma_{semester=2015F \land instructor=Doeppner}(Course) \bowtie Enrollment) \)