Solutions to Practice Exercises

2.1 a. \( \Pi_{\text{person.name}} ( (\text{employee} \bowtie \text{manages}) \nexists (\text{manager.name} = \text{employee2.person.name} \land \text{employee.street} = \text{employee2.street} \land \text{employee.city} = \text{employee2.city}) (\rho_{\text{employee2}} (\text{employee}))) \)

b. The following solutions assume that all people work for exactly one company. If one allows people to appear in the database (e.g. in \text{employee}) but not appear in \text{works}, the problem is more complicated. We give solutions for this more realistic case later.

\[ \Pi_{\text{person.name}} (\sigma_{\text{company.name} \neq "First Bank Corporation"}(\text{works})) \]

If people may not work for any company:

\[ \Pi_{\text{person.name}} (\text{employee}) - \Pi_{\text{person.name}} (\sigma_{\text{company.name} = "First Bank Corporation"}(\text{works})) \]

c. \( \Pi_{\text{person.name}} (\text{works}) - (\Pi_{\text{works.person.name}} (\text{works}) \nexists (\text{works.salary} \leq \text{works2.salary} \land \text{works2.company.name} = "Small Bank Corporation") (\rho_{\text{works2}} (\text{works}))) \)

2.2 a. The left outer theta join of \( r(R) \) and \( s(S) \) (\( r \bowtie_{\theta} s \)) can be defined as

\( (r \bowtie_{\theta} s) \cup ((r - \Pi_{R}(r \bowtie_{\theta} s)) \times (\text{null}, \text{null}, \ldots, \text{null})) \)

The tuple of nulls is of size equal to the number of attributes in \( S \).

b. The right outer theta join of \( r(R) \) and \( s(S) \) (\( r \bowtie_{\theta} s \)) can be defined as

\( (r \bowtie_{\theta} s) \cup ((\text{null}, \text{null}, \ldots, \text{null}) \times (s - \Pi_{S}(r \bowtie_{\theta} s))) \)

The tuple of nulls is of size equal to the number of attributes in \( R \).
c. The full outer theta join of \( r \) and \( s \) can be defined as:
\[
(r \bowtie_{\theta} s) \cup ((null, null, \ldots, null) \times (s - \Pi_{S}(r \bowtie_{\theta} s))) \cup ((r - \Pi_{R}(r \bowtie_{\theta} s)) \times (null, null, \ldots, null))
\]
The first tuple of nulls is of size equal to the number of attributes in \( R \), and the second one is of size equal to the number of attributes in \( S \).

2.3

a. \( \text{employee} \leftarrow \Pi_{\text{person.name,street}}^{\text{"Newtown"}} \)
\[
\sigma_{\text{person.name}=\text{"Jones"}}(\text{employee})
\]
\[
\cup \left( \text{employee} - \sigma_{\text{person.name}=\text{"Jones"}}(\text{employee}) \right)
\]

b. The update syntax allows reference to a single relation only. Since this update requires access to both the relation to be updated (\( \text{works} \)) and the \( \text{manages} \) relation, we must use several steps. First we identify the tuples of \( \text{works} \) to be updated and store them in a temporary relation (\( t_{1} \)). Then we create a temporary relation containing the new tuples (\( t_{2} \)). Finally, we delete the tuples in \( t_{1} \) from \( \text{works} \) and insert the tuples of \( t_{2} \).
\[
t_{1} \leftarrow \Pi_{\text{works.person.name,company.name,salary}} \sigma_{\text{works.person.name}=\text{manager.name}}(\text{works} \times \text{manages})
\]
\[
t_{2} \leftarrow \Pi_{\text{person.name,company.name,1.1*salary}}(t_{1})
\]
\[
\text{works} \leftarrow (\text{works} - t_{1}) \cup t_{2}
\]