2.1
The Primary keys are underlined:

\[
\begin{align*}
\text{employee (person\_name, street, city)} \\
\text{works (person\_name, company\_name, salary)} \\
\text{company (company\_name, city)}
\end{align*}
\]

2.2
- Inserting a tuple:
  
  \[(10111, \text{Ostrom, Economics, } 110,000)\]

  into the instructor table, where the department table does not have the
department Economics, would violate the foreign key constraint.

- Deleting the tuple:
  
  \[(\text{Biology, Watson, 90000})\]

  from the department table, where at least one student or instructor
tuple has dept\_name as Biology, would violate the foreign key con-
straint.

6.1
a)

\[\pi_{title}(\sigma_{\text{dept\_name} = '\text{Comp. Sci}' \land \text{credits} = 3}(\text{course}))\]

b)

\[\pi_{ID}(\sigma_{ID = 'Einstein'}(\text{takes} \Join \rho_{\text{HIL}, course, section, semester, year}(\text{teaches})))\]

Assuming the set version of the relational algebra is used, there is no need to explicitly remove duplicates. If the multiset version is used, the grouping operator can be used without any aggregation to remove duplicates. For example given relation \(r(A, B)\) possibly containing duplicates, \(A, B \ G(r)\) would return a duplicate free version of the relation.

c)

\[\pi_{\text{salary}}(\text{instructor}) - \pi_{\text{instructor\_salary}}(\sigma_{\text{instructor\_salary} < \text{department\_salary}}(\text{instructor} \times \rho_{\text{department}}(\text{instructor})))\]

d)
\[ \pi_{\text{name}}(\text{instructor}) - \pi_{\text{instructor.name}}(\sigma_{\text{instructor.salary} < \text{department.salary}}(\text{instructor} \times \rho_d(\text{instructor}))) \]

6.2

a) 
\[ \Pi_{\text{person.name}} ((\text{employee} \times \text{manages}) \setminus (\text{manager.name} = \text{employee2.person.name} \land \text{employee.street} = \text{employee2.street} \land \text{employee.city} = \text{employee2.city})(\rho_{\text{employee2}}(\text{employee}))) \]

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

If people may not work for any company:
\[ \Pi_{\text{person.name}}(\text{employee}) - \Pi_{\text{person.name}} (\sigma_{\text{company.name} = \text{"First Bank Corporation"}}(\text{works})) \]

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

6.10

a) 
\[ \Pi_{\text{name}} (\text{student} \times \text{takes} \times \Pi_{\text{course.id}} (\sigma_{\text{dept.name} = \text{'Comp.Sci.'}}(\text{course}))) \]

b) 
\[ \Pi_{\text{ID.name}} (\text{student}) - \Pi_{\text{ID.name}} (\sigma_{\text{year} < 2009}(\text{student} \times \text{takes})) \]

6.11

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

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

c)

\[ \Pi_{\text{person-name}, \text{street}, \text{city}} \\
(\sigma_{\text{company-name} = "First Bank Corporation" \land \text{salary} > 10000} \text{works} \Join \text{employee}) \]

d)

\[ \Pi_{\text{person-name}} (\text{employee} \Join \text{works} \Join \text{company}) \]