2.7 Consider the relational database of Figure **??**. Give an expression in the relational algebra to express each of the following queries:

a. Find the names of all employees who live in city "Miami".

b. Find the names of all employees whose salary is greater than $100,000.

c. Find the names of all employees who live in "Miami" and whose salary is greater than $100,000.

Answer:

a. $\Pi_{person\_name}(\sigma_{city="Miami"}(employee))$

b. $\Pi_{person\_name}(\sigma_{salary>100000}(works))$

c. $\Pi_{person\_name}(\sigma_{city="Miami" \wedge salary>100000}(employee \bowtie works))$

2.10 Consider the *advisor* relation shown in Figure 2.8, with *s_id* as the primary key of *advisor*. Suppose a student can have more than one advisor. Then, would *s_id* still be a primary key of the *advisor* relation? If not, what should the primary key of *advisor* be?
Answer: No, *s_id* would not be a primary key, since there may be two (or more) tuples for a single student, corresponding to two (or more) advisors. The primary key should then be *s_id, i_id*.

2.13 Consider the bank database of Figure 2.15. Give an expression in the relational algebra for each of the following queries:

a. Find all loan numbers with a loan value greater than $10,000.

b. Find the names of all depositors who have an account with a value greater than $6,000.

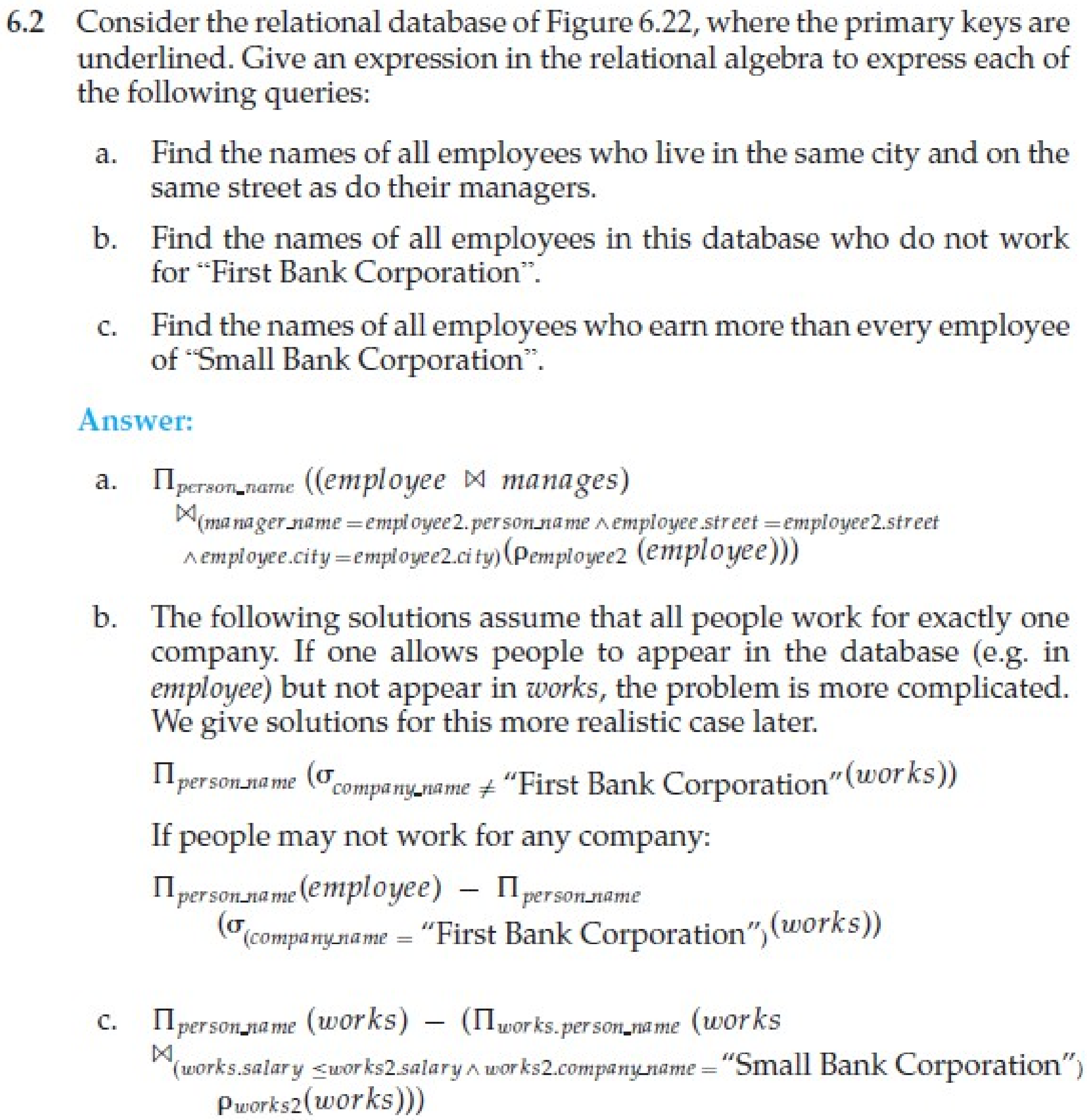
c. Find the names of all depositors who have an account with a value greater than $6,000 at the "Uptown" branch.

Answer:

a. $\Pi_{loan\_number}\ (\sigma_{amount>\ 10000}(loan)$

b. $\Pi_{customer\_name}\ (\sigma_{balance>\ 6000}\ (depositor \bowtie\ account))$

c. $\Pi_{customer\_name}\ (\sigma_{balance>\ 6000 \wedge branch\_name=\text{``Uptown''}}\ (depositor \bowtie\ account))$



6.2 Consider the relational database of Figure 6.22, where the primary keys are underlined. Give an expression in the relational algebra to express each of the following queries:

a. Find the names of all employees who live in the same city and on the same street as do their managers.

b. Find the names of all employees in this database who do not work for "First Bank Corporation".

c. Find the names of all employees who earn more than every employee of "Small Bank Corporation".

Answer:

a. $\Pi_{person\_name}\ ((employee\ \bowtie\ manages)$
$\bowtie_{(manager\_name = employee2.person\_name \land employee.street = employee2.street}$
$_{\land employee.city = employee2.city)}(\rho_{employee2}\ (employee)))$

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 *employee*) but not appear in *works*, the problem is more complicated. We give solutions for this more realistic case later.

$\Pi_{person\_name}\ (\sigma_{company\_name \neq \text{“First Bank Corporation”}}(works))$

If people may not work for any company:

$\Pi_{person\_name}(employee)\ -\ \Pi_{person\_name}$
$(\sigma_{(company\_name = \text{“First Bank Corporation”})}(works))$

c. $\Pi_{person\_name}\ (works)\ -\ (\Pi_{works.person\_name}\ (works$
$\bowtie_{(works.salary \leq works2.salary \land works2.company\_name = \text{“Small Bank Corporation”})}$
$\rho_{works2}(works)))$

NOTE: There are many ways to solve this last problem, especially depending on what you assume about the relationship between employees and companies.