For the above schema (the primary key for each relation is denoted by the underlined attribute), provide relational algebra expressions for the following queries:

**Note:** For notational convenience, I am using pname instead of person-name, cname instead of company-name, and mname instead of manager-name. The SQL queries provided here are 'translation' of the relational algebra (RA) queries - there may be better/efficient ways to write them.

1. Find the name of all employees (i.e., persons) who work for the City Bank company (which is a specific company in the database).

   \[ \pi_{\text{person-name}} (\sigma_{\text{cname}=\text{'City Bank'}}(\text{works})) \]

   ```sql
   SELECT person-name
   FROM works
   WHERE cname= 'City Bank' ;
   ```

2. Find the name and city of all employees who work for City Bank. Similar to previous query, except we have to access the lives table to extract the city of the employee. The join condition is the same person name in the two tables Lives and Works.

   \[ \pi_{\text{lives.pname, city}} (\sigma_{((\text{cname}=\text{'City Bank'})\land(\text{lives.pname}=\text{works.pname}))}(\text{works } \times \text{lives})) \]

   ```sql
   SELECT lives.pname, city
   FROM works, lives
   WHERE cname='City Bank' AND works.pname=lives.pname;
   ```

3. Find the name, street and city of all employees who work for City Bank and earn more than $50,000. Similar to previous query except an additional condition on salary attribute.

   \[ \pi_{\text{lives.pname, street, city}} (\sigma_{((\text{cname}=\text{'City Bank'})\land(\text{lives.pname}=\text{works.pname})\land(\text{salary}>50000))}(\text{works } \times \text{lives})) \]

   ```sql
   SELECT lives.pname, city
   FROM works, lives
   WHERE cname='City Bank' AND works.pname=lives.pname AND salary > 50000 ;
   ```

4. Find all employees who live in the same city as the company they work for. For this query we need to access the lives table to get city of the employee and the located-in table to get city of the company; plus the works table to associate employee with their company. The selection condition is then that the two cities are the same.
\[ \pi_{\text{lives.pname}} \\
\quad \left( \sigma \left( \left( \text{locatedin.cname} = \text{works.cname} \right) \land \left( \text{located.in.city} = \text{lives.city} \right) \land \left( \text{lives.pname} = \text{works.pname} \right) \right) \right) \\
\quad \left( \text{works} \times \text{lives} \times \text{locatedin} \right) \]

```
SELECT lives.pname 
FROM lives, works, locatedin 
WHERE lives.pname = works.pname AND works.cname = locatedin.cname 
\quad AND lives.city = locatedin.city;
```

5. Find all employees who live in the same city and on the same street as their manager. This requires accessing lives table twice – once for finding city of employee and a second time for finding city of manager. Therefore we need the rename operator – so access and rename lives as mlives to indicate this is for information about where the manager lives. To find manager of employee we need to look up the manages table.

\[ \pi_{\text{lives.pname}} \\
\quad \left( \sigma \left( \left( \text{lives.city} = \text{mlives.city} \right) \land \left( \text{lives.street} = \text{mlives.street} \right) \right) \land \left( \text{manages.pname} = \text{lives.pname} \right) \land \left( \text{mname} = \text{mlives.pname} \right) \right) \\
\quad \left( \text{lives} \times \text{manages} \times \left( \rho_{\text{mlives}} \left( \text{lives} \right) \right) \right) \]

```
SELECT E.pname 
FROM lives E, lives M, manages 
WHERE E.pname = manages.pname AND manages.mname = M.pname 
\quad AND E.city = M.city AND E.street = M.street;
```