Consider the following employee database, where the primary keys are underlined. Give an expression in SQL for each of the following queries.

employee (employee-name, street, city)

works (employee-name, company-name, salary)

company (company-name, city)

SQL Queries for Employee Database:

a. Employees working for First Bank Corporation:

SQL

```
SELECT employee-name
FROM works
WHERE company-name = 'First Bank Corporation';
```

b. Names and cities of employees of First Bank Corporation:

SQL

```
SELECT employee-name, city
FROM employee
JOIN works ON employee.employee-name = works.employee-name
WHERE company-name = 'First Bank Corporation';
```

c. Employees of First Bank Corporation earning more than 10,000:

SQL

```
SELECT employee-name, street, city
FROM employee
JOIN works ON employee.employee-name = works.employee-name
WHERE company-name = 'First Bank Corporation' AND salary > 10000;
```

d. Employees living in same city as their company:

SQL

```
SELECT employee-name
FROM employee
JOIN works ON employee.employee-name = works.employee-name
JOIN company ON works.company-name = company.company-name
WHERE employee.city = company.city;
```

e. Employees not working for First Bank Corporation:

SQL

```
SELECT employee-name
FROM employee
WHERE employee-name NOT

IN (
    SELECT employee-name
    FROM works
    WHERE company-name =

'First Bank Corporation'
);
```

f. Employees earning more than any Small Bank Corporation employee:

SQL

```
SELECT employee-name
FROM employee
JOIN works ON employee.employee-name = works.employee-name
WHERE salary > (
    SELECT MAX(salary)
    FROM works
    WHERE company-name = 'Small Bank Corporation'
);
```

g. Companies located in all Small Bank Corporation cities:

SQL

```
SELECT company-name
FROM company
WHERE city IN (
    SELECT DISTINCT city
    FROM works
    WHERE company-name = 'Small Bank Corporation'
);
```

h. Employees earning more than their company's average salary:

SQL

```
SELECT employee-name
FROM employee
JOIN works ON employee.employee-name = works.employee-name
WHERE salary > (
    SELECT AVG(salary)
    FROM works
    WHERE company-name = works.company-name
);
```

i. Company with the most employees:

SQL

```
SELECT company-name
FROM works
GROUP BY company-name
ORDER BY COUNT(*) DESC
LIMIT 1;
```

j. Company with the smallest payroll:

SQL

```
SELECT company-name
FROM works
GROUP BY company-name
ORDER BY SUM(salary) ASC
LIMIT 1;
```

k. Companies with higher average employee salary than First Bank Corporation:

SQL

```
SELECT company-name
FROM works
GROUP
```

```
BY company-name
HAVING

AVG(salary) > (
    SELECT

AVG(salary)
    FROM works
    WHERE company-name

=

'First Bank Corporation'
);
```

l. Update Jones' city to Newtown:

SQL

```
UPDATE employee
SET city = 'Newtown'
WHERE employee-name = 'Jones';
```

m. Give First Bank Corporation employees a 10% raise:

SQL

```
UPDATE works
SET salary = salary * 1.1
WHERE company-name = 'First Bank Corporation';
```

n. Give First Bank Corporation managers a 10% raise:

SQL

```
UPDATE works
SET salary = salary * 1.1
WHERE company-name = 'First Bank Corporation' AND EXISTS (
    SELECT *
    FROM employee
    WHERE employee-name = works.employee-name AND job = 'Manager'
);
```

o. Delete Small Bank Corporation employee records from works:

SQL

```
DELETE FROM works
WHERE company-name = 'Small Bank Corporation';
```