employee (*ID*, *person_name, street, city*)
company (*company_name, city*)
works (ID, *company_name, salary*)
manages (ID, manager_id)

Figure 1

Consider the database in Figure 1, where the primary keys are underlined. The *manages* relation describes the manager (manager_id) of a certain employee (ID). Each manager is also an employee himself (herself). Construct the following SQL queries for this relational database.

1. Find the ID, name, and manager of each employee who works for "FirstBank".

1. A:

```
select distinct ID,person_name,manager_id
from employee,works,manages
where employee.ID = works.ID and employee.ID = manage.ID and
works.company_name = 'FirstBank';
```

2. Find the ID of each employee who does not work for "FirstBank".

A:

```
select distinct e.ID
from employee as e
where ID not in (select ID from works where company_name = 'FirstBank');
```

3. Find the ID and name of each employee who lives in the same city as the location of the company for which the employee works.

A:

```
select distinct ID,person_name
from employee,works,company
where employee.ID = works.ID and employee.city = company.city and
works.company_name = company.company_name ;
```

4. Find the ID of each employee who earns more than at least one employee of "SmallBank".

(1) Please use "tuple variable".

A:

```
select distinct T.ID
from works as T, works as S
```

where S.company_name = ‘SmallBank’ **and** S.salary < T.salary

(2) Please use “nested subquery” in the WHERE clause.

A:

```
select distinct ID
from works
where salary > some( select salary
                     from works
                     where company_name=’SmallBank’);
```

5. Find the name of each company whose employees earn a higher salary, on average, than the average salary at “FirstBank”.

(1) Please use “having”

A:

```
select company_name
from works
group by company_name
having avg (salary) > (select avg (salary)
                       from works
                       where company_name = ‘FirstBank’);
```

(2) Please use “with”.

A:

```
with firstbank_avg(value) as ( select avg(salary)
                               from works
                               where company_name = ‘FistBank’)
select company_name
from works, firstbank_avg
group by company_name
where avg(salary)>small_avg.salary;
```

6. Delete all tuples in the *works* relation for employees of “SmallBank”.

A:

```
delete from works
where company_name = 'SmallBank';
```

7. Add a new employee with the ID as “E01” and the name as “John”, but the address is currently unknown.

A:

```
insert into employee
    Values ('E01', John , null,null);
```

8. Give each employee of "FirstBank" a 10-percent raise of salaries unless the salary becomes greater than $100000; in such cases, give only a 3-percent raise.

   A:

```
update works
        set salary = salary * 1.1
        where salary > 100000;
update works
        set salary = salary * 1.03
       where salary <= 100000;
```

Note:

1. Please submit your homework in a single PDF file to Tronclass before 2023/10/25 23:59 (星期三)

2. We do NOT accept late submission for this homework.