Laboratory work 1

Please write your answers to the pdf file for defense:

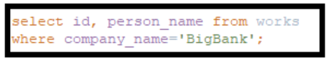
1. Consider the employee database of figure below. Give an expression in the relational algebra to express each of the following queries:

employee (person_name, street, city)
works (person_name, company_name, salary)
company (company_name, city)

Figure

- Find the ID and name of each employee who works for “BigBank”.

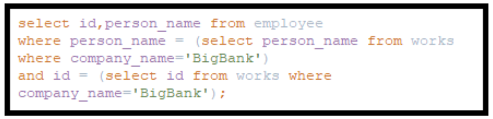
$$\Pi_{id,name}\left(\sigma_{company\_name\ =\ \text{"BigBank"}}(works)\right)$$



```
select id, person_name from works
where company_name='BigBank';
```

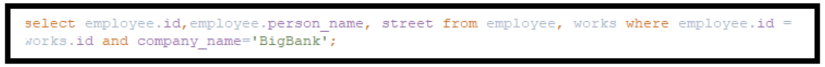
- Find the ID, name, and city of residence of each employee who works for “BigBank”.

$$\Pi_{id,person\_name,\ street}\left(\sigma_{company\_name\ =\ \text{"BigBank"}}\left(employee \bowtie_{employee.id=works.id} works.\right)\right)$$



```
select id,person_name from employee
where person_name = (select person_name from works
where company_name='BigBank')
and id = (select id from works where
company_name='BigBank');
```

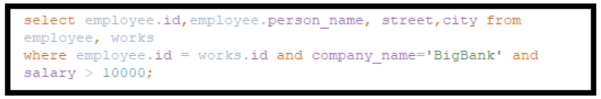
OR



```
select employee.id,employee.person_name, street from employee, works where employee.id =
works.id and company_name='BigBank';
```

- Find the ID, name, street address, and city of residence of each employee who works for “BigBank” and earns more than $10000.

$$\Pi_{id,person\_name,street,\ city}\left(\sigma_{company\_name\ =\ \text{"BigBank"} \wedge\ salary>10000}\left(employee \bowtie_{employee.id=works.id} works.\right)\right)$$



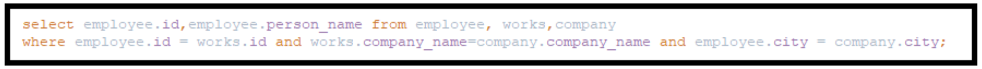
```
select employee.id,employee.person_name, street,city from
employee, works
where employee.id = works.id and company_name='BigBank' and
salary > 10000;
```

- Find the ID and name of each employee in this database who lives in the same city as the company for which she or he works.

$$\Pi_{id,person\_name}\ ((employee \bowtie_{employee.id=works.id} works.) \wedge (works \bowtie_{works.company\_name=company.company\_name} Company.) \wedge (employee \bowtie_{employee.city\ =\ company.city} Company)$$

OR

$$\Pi_{id,person\_name}\left(\sigma_{employee.id=works.id\ \wedge\ works.company\_name=company.company\_name\ \wedge\ employee.city=company.city}\left(employee \times works \times company\right)\right)$$

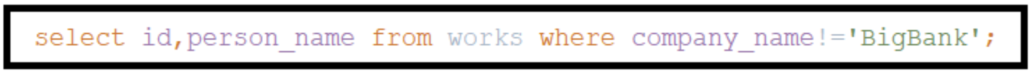


```
select employee.id,employee.person_name from employee, works,company
where employee.id = works.id and works.company_name=company.company_name and employee.city = company.city;
```

2. Consider the employee database of figure above. Give an expression in the relational algebra to express each of the following queries:

- Find the ID and name of each employee who does not work for “ BigBank”.

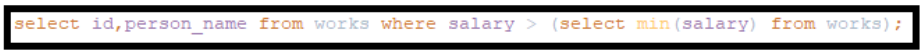
$$\Pi_{\text{id,name}} \left(\sigma_{\text{company\_name} \neq \text{BigBank"}} (\text{works}) \right)$$



```
select id,person_name from works where company_name!='BigBank';
```

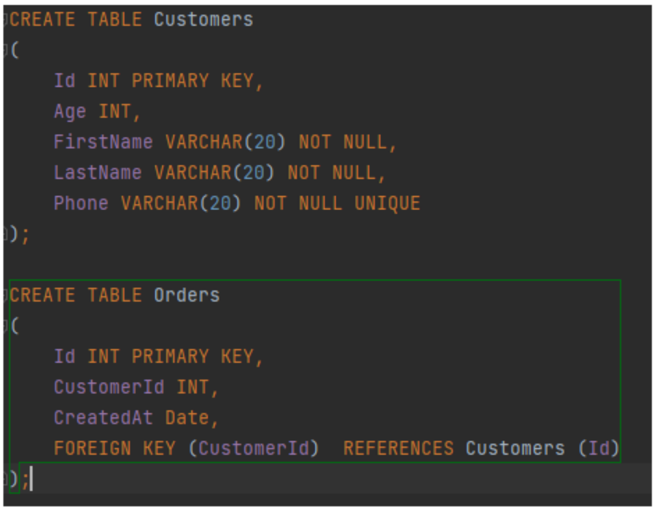
- Find the ID and name of each employee who earns at least as much as every employee in the database.

$$\Pi_{\text{id,name}} \left(\sigma_{\text{salary>min(salary)}} (\text{works}) \right)$$



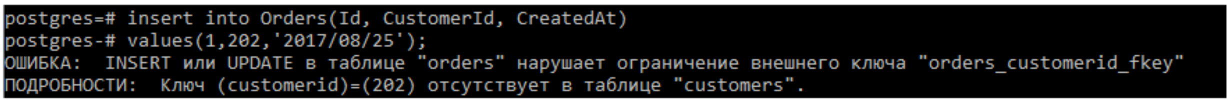
```
select id,person_name from works where salary > (select min(salary) from works);
```

3. Consider the foreign-key constraint from the *dept_name* attribute of instructor to the *department* relation. Give examples of inserts and deletes to these relations that can cause a violation of the foreign-key constraint.



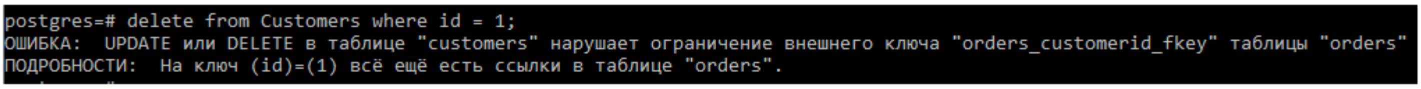
```
CREATE TABLE Customers
(
    Id INT PRIMARY KEY,
    Age INT,
    FirstName VARCHAR(20) NOT NULL,
    LastName VARCHAR(20) NOT NULL,
    Phone VARCHAR(20) NOT NULL UNIQUE
);

CREATE TABLE Orders
(
    Id INT PRIMARY KEY,
    CustomerId INT,
    CreatedAt Date,
    FOREIGN KEY (CustomerId)  REFERENCES Customers (Id)
);
```



```
postgres=# insert into Orders(Id, CustomerId, CreatedAt)
postgres-# values(1,202,'2017/08/25');
ОШИБКА:  INSERT или UPDATE в таблице "orders" нарушает ограничение внешнего ключа "orders_customerid_fkey"
ПОДРОБНОСТИ:  Ключ (customerid)=(202) отсутствует в таблице "customers".
```

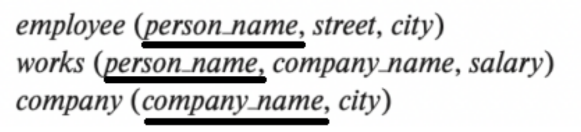
Create tuple in Orders when CustomerID its not created in Customers



```
postgres=# delete from Customers where id = 1;
ОШИБКА:  UPDATE или DELETE в таблице "customers" нарушает ограничение внешнего ключа "orders_customerid_fkey" таблицы "orders"
ПОДРОБНОСТИ:  На ключ (id)=(1) всё ещё есть ссылки в таблице "orders".
```

Delete from Customers primary key(which is referenced by multiple foreign keys)

4. Consider the employee database of figure above. What are the appropriate primary keys?



$employee\ (\underline{person\_name}, street, city)$

$works\ (\underline{person\_name}, company\_name, salary)$

$company\ (\underline{company\_name}, city)$