

Laboratory work 1

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

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

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

$$\Pi_{ID, person_name} (\sigma_{company_name = \text{“BigBank”}}(works))$$

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

$$\Pi_{ID, person_name, city} (employee \bowtie_{employee.id=works.id} (\sigma_{company_name = \text{“BigBank”}}(works)))$$

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

$$\Pi_{ID, person_name, street, city} (employee \bowtie_{employee.id=works.id} (\sigma_{company_name = \text{“BigBank”} \wedge salary > 10000}(works)))$$

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:

$$\Pi_{ID, person_name} (employee \bowtie_{employee.city=company.city} company)$$

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”:

$$\Pi_{ID, person_name} (\sigma_{company_name \neq \text{“BigBank”}}(works))$$

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

$\Pi_{ID, person_name} (works \bowtie_{works.salary \geq copy.salary} (\rho_{copy}(works)))$

ID	name	dept_name	salary
22222	Einstein	Physics	95000
12121	Wu	Finance	90000
32343	El Said	History	60000
45565	Katz	Comp. Sci.	75000
98345	Kim	Elec. Eng.	80000
76766	Crick	Biology	72000
10101	Srinivasan	Comp. Sci.	65000
58583	Califleri	History	62000
83821	Brandt	Comp. Sci.	92000
15151	Mozart	Music	40000
33456	Gold	Physics	87000
76543	Singh	Finance	80000

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.

So, foreign key (*dept_name*) references *department*

Insert: (123456, Jake, Database, 98000)

(123123, Nuke, FiNance, 12000)

Delete: (654321, Lake, Physics, 89000)

(321321, Pawl, Music, 21000)

4. Consider the employee database of figure above. What are the appropriate primary keys?
Employee - primary key (ID)

Works – primary key(ID)

Company – primary key(company_name)