

Database Concepts

Exercise 11

1. Given the following example database from the appendix. Formulate following queries using relational algebra:

- (a) Query the names of employees, who work on all projects that "John Smith" is working on.
- (b) Query name and address of all employees, who work for the "Research" department.
- (c) Query the project number of the project that is located in "Stafford". Moreover, you should retrieve the number of the department that controls the project as well as as the responsible manager's name, address and birth day.
- (d) Query the names of employees, who work on all projects controlled by department 5.
- (e) List all project numbers of projects that involve an employee (including managers) whose last name is "Smith".
- (f) Find the names of all employees who have two or more dependents.
- (g) Find the names of all employees who have no dependents.
- (h) Find the names of all managers who have at least one dependent.

2. Given following relational schema:

| | |
|-----------------|--|
| Station: | (Name: string) |
| Train: | (<u>Train_number</u> : integer) |
| Local_train: | (<u>Train_number</u> →Train, Bikes_allowed: boolean) |
| Distance_train: | (<u>Train_number</u> →Train, Dining_car: boolean, Label: string) |
| Car: | (<u>Car_number</u> : integer, <u>Train_number</u> →Train, Position: integer) |
| Seat: | (<u>Car_number</u> : integer→Car, <u>Seat_number</u> : integer, Category: integer, Smoker: boolean, Window:boolean) |
| Connection: | (Arrival: time, Departure: time, Day: date, starts_at: string→Station, goes_to: string→Station, <u>Train_number</u> : integer→Train) |
| Ticket: | (Price: integer, <u>Ticket_nubmer</u> : integer) |
| Reservation: | (<u>Ticket_number</u> →Ticket, (Arrival, Departure, Day, starts_at, |

| | |
|-----------------|--|
| | $\frac{\text{goes_to, Train_number} \rightarrow \text{Connection, (Car_number, Seat_number)} \rightarrow \text{Seat, Price: integer}}{((\text{Arrival, Departure, Date, starts_at, goes_to, Train_number}) \rightarrow \text{Connection, Ticket_number} \rightarrow \text{Ticket})}$ |
| Valid for: | |
| Adds_discount: | $\frac{\text{(Label: string, Unit: string, Amount: integer)}}{\text{requires: string} \rightarrow \text{Adds_Discount}}$ |
| Imputation: | $\frac{\text{(Ticket_number} \rightarrow \text{Ticket, Label} \rightarrow \text{Adds_discount})}{\text{(Excluder: string} \rightarrow \text{Adds_discount, Excluded: string} \rightarrow \text{Adds_discount)}}$ |
| Excludes: | |

Formulate following queries using the tuple calculus:

- (a) Find all stations.
- (b) Find the labels of all discounts and additions.
- (c) Find all tickets that cost more than 100€.
- (d) Find all departure times of all connections that go from Munich to Augsburg before noon (12 o'clock).
- (e) Find all trains that have a connection from Munich to Augsburg.
- (f) Find all discounts and additions that do not depend on others and do not exclude others.

3. Formulate the queries from task 2 using the domain calculus.

4. Trigger and Integrity:

- (a) Explain the ACID principle.
- (b) What are triggers used for in DBMS?
- (c) What possibilities do you know to ensure integrity using triggers?

5. Create the following views in SQL:

- (a)
 - i. Create a view *employee_view* on relation *employee* that only shows the name, the address and the job of all employees. Rename the columns *name* into *empview_name* and *adresse* into *empview_adresse*!
 - ii. Now, the view *employee_view* should only list employees which earn more than 5000€ .
 - iii. Is it possible to insert new employees into view *employee_view*?
- (b) Given following relation:

exams (*course_of_studies, course, student, examiner, date, mark*)

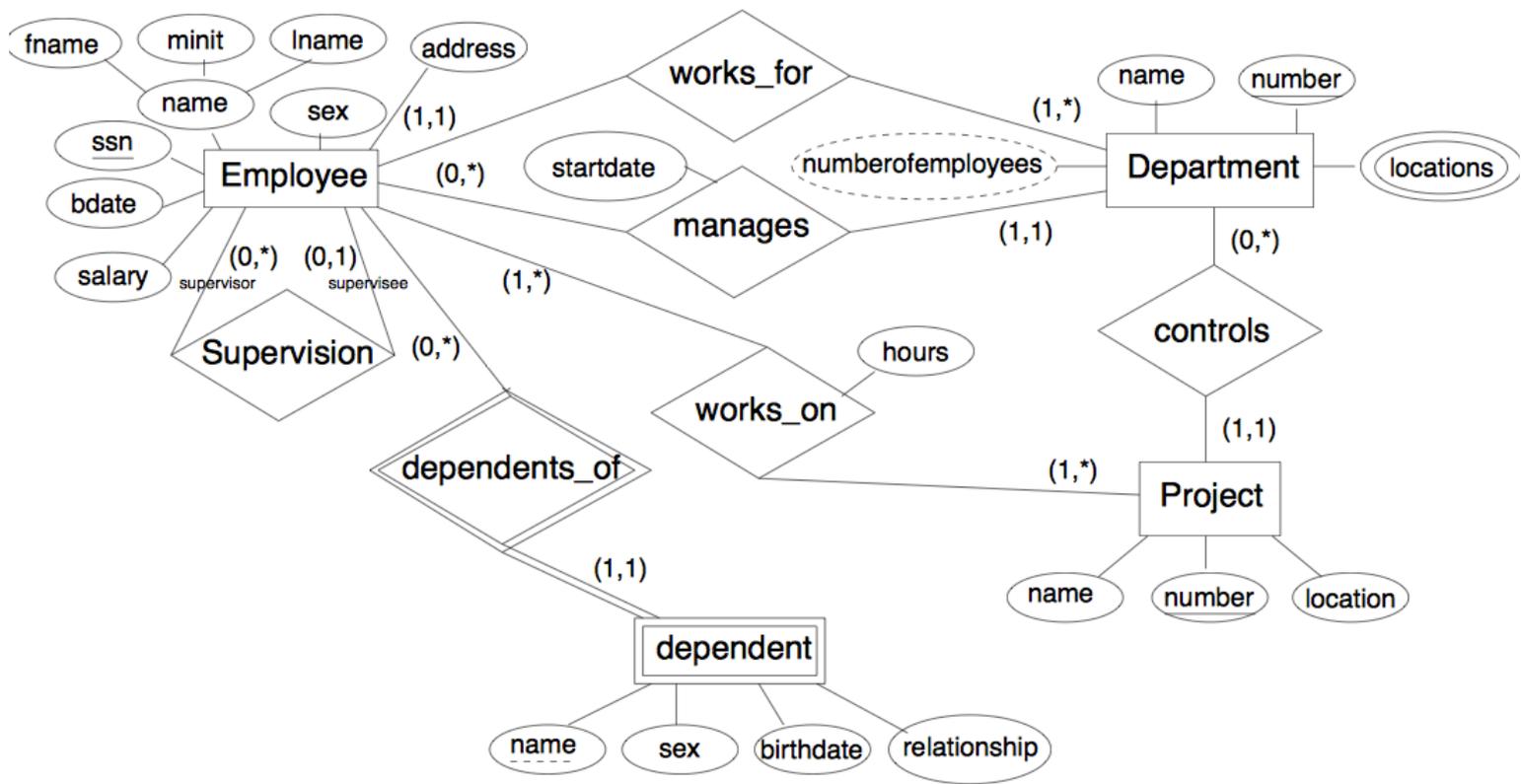
Define following views using SQL:

- i. The computer science faculty can only view data of students that are registered in *computerscience*.

- ii. The examination office can view all data.
- iii. The scholarship commission can only view average marks of every student.
- iv. The dean can only view data about exams of the last year for statistical purposes (i.e., the relationship to students and examiners must be removed).

Good Luck!

Appendix:



1. Employee(fname, minit, lname, ssn, bdate, address, sex, salary, superssn → Employee, dno → Department)
2. Department(dname, dnumber, mgrssn → Employee, mgrstartdate)
3. Dept_locations(dnumber → Department, dlocation)
4. Project(pname, pnumber, plocation, dnum → Department)
5. Works_on(essn → Employee, pno → Project, hours)
6. Dependent(essn → Employee, dependent_name, sex, bdate, relationship)

| EMPLOYEE | | | | | | | | | |
|----------|-------|---------|-----------|------------|--------------------------|-----|--------|-----------|-----|
| FNAME | MINIT | LNAME | SSN | BDATE | ADDRESS | SEX | SALARY | SUPERSSN | DNO |
| John | B | Smith | 123456789 | 1965-01-09 | 731 Fondren, Houston, TX | M | 30000 | 333445555 | 5 |
| Franklin | T | Wong | 333445555 | 1955-12-08 | 638 Voss, Houston, TX | M | 40000 | 888665555 | 5 |
| Alicia | J | Zelaya | 999887777 | 1968-07-19 | 3321 Castle, Spring, TX | F | 25000 | 987654321 | 4 |
| Jennifer | S | Wallace | 987654321 | 1941-06-20 | 291 Berry, Bellaire, TX | F | 43000 | 888665555 | 4 |
| Ramesh | K | Narayan | 666884444 | 1962-09-15 | 975 Fire Oak, Humble, TX | M | 38000 | 333445555 | 5 |
| Joyce | A | English | 453453453 | 1972-07-31 | 5631 Rice, Houston, TX | F | 25000 | 333445555 | 5 |
| Ahmad | V | Jabbar | 987987987 | 1969-03-29 | 980 Dallas, Houston, TX | M | 25000 | 987654321 | 4 |
| James | E | Borg | 888665555 | 1937-11-10 | 450 Stone, Houston, TX | M | 55000 | null | 1 |

| DEPARTMENT | | | |
|----------------|---------|-----------|--------------|
| DNAME | DNUMBER | MGRSSN | MGRSTARTDATE |
| Research | 5 | 333445555 | 1988-05-22 |
| Administration | 4 | 987654321 | 1995-01-01 |
| Headquarters | 1 | 888665555 | 1981-06-19 |

| DEPT_LOCATIONS | |
|----------------|-----------|
| DNUMBER | DLOCATION |
| 1 | Houston |
| 4 | Stafford |
| 5 | Bellaire |
| 5 | Sugarland |
| 5 | Houston |

| WORKS_ON | | |
|-----------|-----|-------|
| ESSN | PNO | HOURS |
| 123456789 | 1 | 32,5 |
| 123456789 | 2 | 7,5 |
| 666884444 | 3 | 40,0 |
| 453453453 | 1 | 20,0 |
| 453453453 | 2 | 20,0 |
| 333445555 | 2 | 10,0 |
| 333445555 | 3 | 10,0 |
| 333445555 | 10 | 10,0 |
| 333445555 | 20 | 10,0 |
| 999887777 | 30 | 30,0 |
| 999887777 | 10 | 10,0 |
| 987987987 | 10 | 35,0 |
| 987987987 | 30 | 5,0 |
| 987654321 | 30 | 20,0 |
| 987654321 | 20 | 15,0 |
| 888665555 | 20 | null |

| DEPENDENT | | | | |
|-----------|----------------|-----|------------|--------------|
| ESSN | DEPENDENT_NAME | SEX | BDATE | RELATIONSHIP |
| 333445555 | Alice | F | 1986-04-05 | DAUGHTER |
| 333445555 | Theodore | M | 1983-10-25 | SON |
| 333445555 | Joy | F | 1958-05-03 | SPOUSE |
| 987654321 | Abner | M | 1942-02-28 | SPOUSE |
| 123456789 | Michael | M | 1988-01-04 | SON |
| 123456789 | Alice | F | 1988-12-30 | DAUGHTER |
| 123456789 | Elizabeth | F | 1967-05-05 | SPOUSE |

| PROJECT | | | |
|-----------------|---------|-----------|------|
| PNAME | PNUMBER | PLOCATION | DNUM |
| ProductX | 1 | Bellaire | 5 |
| ProductY | 2 | Sugarland | 5 |
| ProductZ | 3 | Houston | 5 |
| Computerization | 10 | Stafford | 4 |
| Reorganization | 20 | Houston | 1 |
| Newbenefits | 30 | Stafford | 4 |