Databases Exam

TDA357 (Chalmers), DIT620 (University of Gothenburg)

9 January 2018, 14:00-18:00

Department of Computer Science and Engineering

Course responsible Pablo Picazo-Sanchez (EDIT 5472). Markus and Andrea will visit the exam rooms around 15:00 and 17:00.

Exam review See the course web page for time and place:
`http://www.cse.chalmers.se/edu/year/2017/course/TDA357/HT2017/`

Grades Chalmers: 24 for 3, 36 for 4, 48 for 5. GU: 24 for G, 42 for VG.

Help material One cheat sheet, which is an A4 sheet with hand-written notes. You may write on both sides of that sheet. If you bring a sheet, it must be handed in with your answers to the exam questions. One English language dictionary is also allowed.

Specific instructions Answer questions in English. Begin the answer to each question (numbers 1 to 5) on a new page. The a,b,c,... parts with the same number can be on the same page.

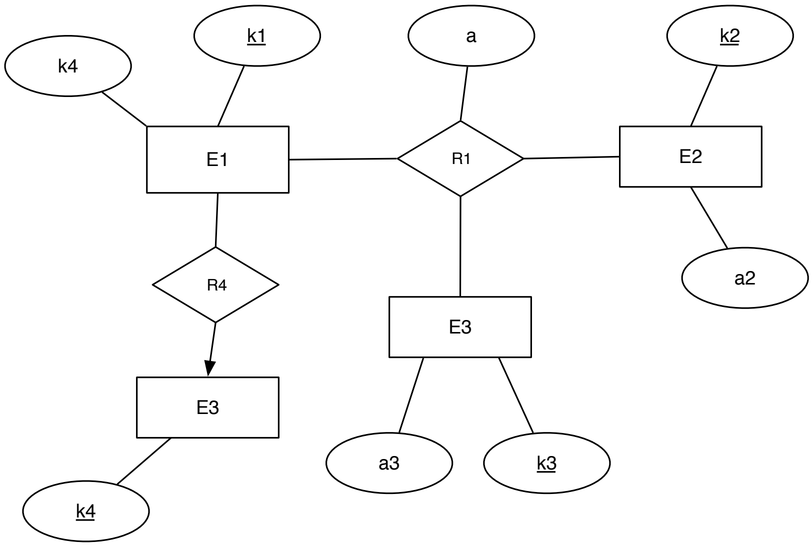
Write clearly unreadable = wrong! Fewer points are given for unnecessarily complicated solutions. Indicate clearly if you make any assumptions that are not given in the question. In SQL questions, use standard SQL or PostgreSQL. If you use any other variant (such as Oracle or MySQL), say this; but full points are not guaranteed since this may change the nature of the question.

Multiple Choice Marks right answer = +1, bad answer = -0,5; blank = 0. The minimum score that you can get for question 1 is zero — if the total points for question 1 turns out to be negative, then a value of zero will be given for this question.

1 Multiple Choice Questions (10p)

For each question, it is sufficient to write the letter corresponding to the correct answer.

1) Given the following Entity-Relationship diagram:



How is relationship R1 represented in the relational database schema?

a) R1(<u>k1</u>,<u>k2</u>,<u>k3</u>)

b) R1(<u>k1</u>,<u>k2</u>,<u>k3</u>,k4,a)

c) R1(<u>k1</u>,<u>k2</u>,<u>k3</u>,a)

d) R1(<u>k1</u>,<u>k2</u>,<u>k3</u>,<u>k4</u>,a)

C

---

2) Given these SQL statements:

```
CREATE TABLE a (
  f1 INTEGER PRIMARY KEY,
  f2 INTEGER);

CREATE TABLE b (
  f2 INTEGER PRIMARY KEY,
  f1 INTEGER REFERENCES a ON DELETE CASCADE);

ALTER TABLE a ADD FOREIGN KEY (f2) REFERENCES b(f2) ON DELETE CASCADE;

INSERT INTO a VALUES(1,null);
INSERT INTO a VALUES(2,null);

INSERT INTO b VALUES(10,1);
UPDATE a SET f2=10 WHERE f1=2;

DELETE FROM a WHERE f1=1;

SELECT * FROM a;
```

What will be the final output?

a) (2,10)

b) no results are given

c) (2,null)

d) (2,10) and (1,null)

B

---

3) Suppose we have relations r(A,B,C,D) and s(B,D,E). Which is the SQL operation that correspond to $\pi_{r.A,r.B,r.C,r.D,s.E}(\sigma_{r.B=s.B \wedge r.D=s.D}(\mathrm{r} \times \mathrm{s}))$:

a) NATURAL JOIN
b) LEFT JOIN
c) RIGHT JOIN
d) CARTESIAN PRODUCT

A

---

4) Which of the following is used to define the structure of an XML document:

a) DOCTYPE
b) #PCDATA
c) DTD
d) CDATA

C

---

5) To sort the results of an SQL query, we use:

a) SORT BY
b) ASC / DESC
c) GROUP BY
d) ORDER BY

D

---

6) Given this SQL query:

```
UPDATE instructor
_____ salary= salary * 1.05;
```

Fill in with the correct keyword to update the instructor table.

a) WHERE
b) SET
c) IN
d) SELECT

B

---

7) In case of any shut down during the execution of a transaction before commit, which of the following statements is done automatically?

a) Rollback
b) View
c) Commit
d) SavePoint

A

---

8) Given these SQL statements:

```
CREATE TABLE Employee(
  Emp_id        INTEGER PRIMARY KEY NOT NULL,
  name          VARCHAR(100),
  dept_name     VARCHAR(100),
  Salary        INTEGER,
  UNIQUE(Emp_id,name));

INSERT INTO Employee VALUES(102,'Markus', 'CSE', 1000);
INSERT INTO Employee VALUES(106,'Ana','Finance');
INSERT INTO Employee VALUES(102,'Steven','Sales',2000);
```

What will be the final result?

a) All statements executed

b) Error in INSERT INTO Employee VALUES(102,'Steven','Sales',2000);

c) Error in CREATE statement

d) Error in INSERT INTO Employee VALUES(106,'Ana','Finance');

B

---

9) Which of the following statement(s) is used to provide DELETE authorization to instructor?

a)
```
CREATE ROLE instructor;
GRANT DELETE TO instructor;
```

b)
```
CREATE ROLE instructor;
GRANT ALL ON takes;
```

c)
```
CREATE ROLE instructor;
GRANT DELETE ON takes TO instructor;
```

d) All three answers: a), b) and c)

C

---

10) Given the relation R=$\{A, B, C, D, E, F, G\}$ and the following Functional Dependencies: FD = $\{BA \to C; AC \to D; A \to B; E \to A; CDE \to F; AC \to E\}$. Which are the keys?

a) $\{FC, FAB\}$

b) $\{FA, AB, FBC\}$

c) $\{GA, GE\}$

d) $\{FD, FAB, GC\}$

C

---

2 Entity Relationship Modelling (12p)

A national museum manages several art galleries, and wants to use use a database to store information about the galleries and the works of art in their collections.

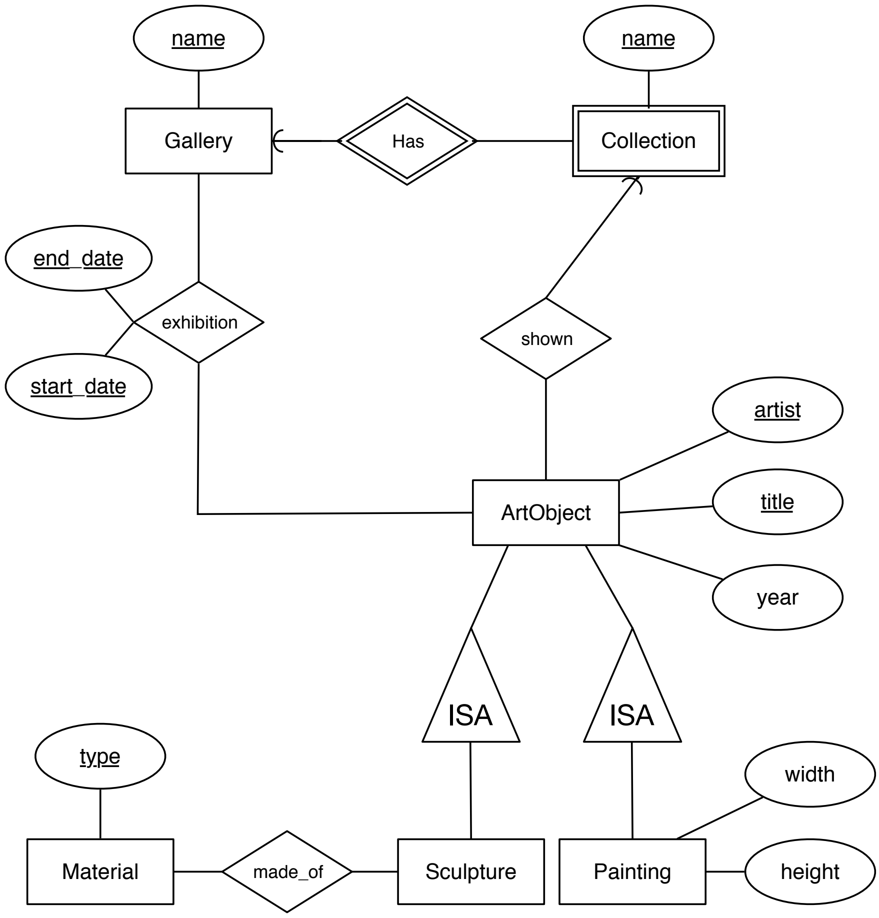
Each art object is identified by a combination of its title and the name of the artist who created the object. The year that the art object was created should also be stored in the database. Paintings and sculptures are two different kinds of art objects. The width and height of paintings should be stored in the database. For sculptures, the material (e.g. wood, bronze) should be stored.

A gallery is identified by its name. Each gallery can have several collections (e.g. "17th Century", "Cubism", etc.). Since different galleries can have collections with the same name, a collection name is not sufficient on its own to identify a collection. Each art object belongs to exactly one collection.

The museum organises exhibitions. An exhibition consists of a set of art objects that can be gathered together from one or more collections. The exhibition will be displayed at a gallery from a start date to an end date. The exhibition might move from one gallery to another (i.e. the same exhibition can be displayed at different galleries, with different start and end dates). Over time, the same art object can be in several different exhibitions.

2a. Draw an Entity-Relationship diagram for this domain listing any assumptions you make. (8p)

SOLUTION:



* Material will also be considered as valid solution as an attribute instead of an entity.
** Exhibitions can also be modelled as an entity.

2b. Translate the ER diagram into a set of relations, clearly marking keys and references in your answer. (4p)

Solution:

Gallery (name)

Collection (name, gallery_name)

    gallery_name→Gallery.name

ArtObject (title, artist, name, gallery_name, year)

    (name, gallery_name)→Collection(name, gallery_name)

Painting (artist, title, width, height)

    (artist, title)→ArtObject(artist,title)

Sculpture (artist, title)

    (artist, title)→ArtObject(artist,title)

Material (type)

made_of (type, artist, title)

    type→Material.type

    (artist, title)→ArtObject(artist,title)

exhibition (name, artist, title, start_date, end_date)

    (artist, title)→ArtObject(artist,title)

    gallery_name→Gallery.name

3 Functional Dependencies and Normal Forms (12p)

Suppose we have relation R(A, B, C, D, E, F) and functional dependencies
FD={AB → CD, C → E, E → CD, A → BCF, E → B, AC → E, AE → F}.

3a. Extract the keys of the relation `R` and give the Normal Form of R. (6p)

SOLUTION: We have to eliminate those weird attributes in the LHS:

- We can get rid off B in $AB \to C$ because $\{A\}^+ = \{ABCDEF\}$ so we have $A \to C$
- We can get rid off B in $AB \to D$ because $\{A\}^+ = \{ABCDEF\}$ so we have $A \to D$
- We can get rid off C in $AC \to E$ because $\{A\}^+ = \{ABCDEF\}$ so we have $A \to E$
- We can get rid off E in $AE \to F$ because $\{A\}^+ = \{ABCDEF\}$ so we have $A \to F$

The resulting FD set is $\{A \to CD,\ C \to E,\ E \to CD,\ A \to BCF,\ E \to B,\ A \to E,\ A \to F\}$

Now we can delete both $A \to C$ and $A \to F$ because they are redundant so we have:
FD=$\{A \to D,\ C \to E,\ E \to CD,\ A \to BCF,\ E \to B,\ A \to E\}$

At this point, if we delete the redundant FD ($A \to D$, $A \to B$ and $A \to C$), the resulting FD are:
FD=$\{C \to E,\ E \to CD,\ A \to F,\ E \to B,\ A \to E\}$
And it is easy to see that the only key is A. Also, $E \to BCD$ violates 3NF since this FD is not trivial, E is not a superkey of `R`, and BCD are not prime in `R`.

SOLUTION 2:
Since no FD has A on its RHS, attribute A must be in any superkeys of R. $\{A\}^+$ is ABCDEF, *i.e.*, all attributes of R, so $\{A\}$ is a superkey of R. No subset of $\{A\}$ is a superkey of R, so $\{A\}$ must be a key of R. And since we know that all superkeys of R must contain A, there can be no other keys of R.

Keys = {A}
Normal Form = 2NF

3b. Decompose relation `R` to BCNF. Show each step in the normalisation process, and at each step indicate which functional dependency is being used. (6p)

SOLUTION:
The first element we choose to normalise is $E \to BCD$ because it violates BCNF:

- $R_1 =< T_1, L_1 >$: $T_1 = \{A, F, E\}; L_1 = \{A \to FE\}$ where A is the key and it is in BCNF
- $R_2 =< T_2, L_2 >$: $T_2 = \{B, C, D, E\}; L_2 = \{C \to E, E \to BCD\}$. Note that there is a redundancy: $C \to E$ and $E \to C$, so the final $L_2 = \{E \to BCD\}$ where E is the only key and it is in BCNF.

To conclude, the schemas in BCNF are: $R_1 =< T_1, L_1 >$ and $R_2 =< T_2, L_2 >$

4 Relational algebra (12p)

A relational database used by an environmental company to monitor fires in South Europe has the following relations: Municipality, Fire and Extent. Each municipality has a name and is identified by a code. Regarding fires, we want to know when the fire started and ended as well as the total area of the fire (in hectares) and also the cause of the fire (e.g. "natural", "arson", etc.). The same fire can damage land which is used in different ways (e.g. "grain", "scrubland", etc.). Also, the fire can spread over several municipalities. The area of each land use that has been destroyed in each municipality is recorded in the relation Extent.

We are provided with the following relational database schema:

```
Municipality(mcode, name, country)
Fire(fcode, start_date, end_date, total_area, cause)
Extent(mcode, fcode, land_use, area)
```

4a. Write a relational algebra expression that finds countries where there have been fires that were caused by "arson", and for these countries count how many municipalities have been affected by fires caused by "arson". (6p)

SOLUTION:
$\gamma_{country,COUNT(mcode)}(\sigma_{cause='arson'}(\text{Extent} \bowtie \text{Municipality} \bowtie \text{Fire}))$

4b. Write a relational algebra expression that finds the names of those municipalities which have **only** been affected by fires where land use type is "grain" or "scrubland". (6p)

SOLUTION:
$S1 = \pi_{fcode}(\sigma_{land\_use='grain'}(\text{Extent})) \bigcup \pi_{fcode}(\sigma_{land\_use='scrubland'}(\text{Extent}))$
$S2 = \pi_{fcode}(\sigma_{land\_use<>'grain'}(\text{Extent})) \bigcup \pi_{fcode}(\sigma_{land\_use<>'scrubland'}(\text{Extent}))$
$S3 = (S1–S2) \bowtie \text{Extent}$
$\pi_{name}((S3) \bowtie (\text{Municipality}))$

5 SQL, Triggers (14p)

Assume the description provided in Question 4 and the same relational database schema:

```
Municipality(mcode, name, country)
Fire(fcode, start_date, end_date, total_area, cause)
Extent(mcode, fcode, land_use, area)
```

5a. Write an SQL query which returns those countries with fires during 2017. (3p)

SOLUTION:

```
SELECT DISTINCT(country)
FROM Municipality,Extent,Fire
WHERE Extent.mcode = Municipality.mcode
  AND Extent.fcode=Fire.fcode
  AND start_date>='1/1/2017'
  AND end_date<'1/1/2018';
```

SOLUTION 2:

```
SELECT DISTINCT(country)
FROM Municipality
WHERE mode IN (
  SELECT mcode
  FROM Extent
  WHERE fcode IN (
    SELECT fcode
    FROM Fire
    WHERE start_date>='1/1/2017' AND end_date<'1/1/2018'));
```

SOLUTION 3:

```
SELECT DISTINCT(country)
FROM Municipality
NATURAL JOIN Extent
NATURAL JOIN Fire
WHERE start_date>='1/1/2017' AND end_date<'1/1/2018';
```

5b. Assuming you have the following function already defined: DATE_PART('day', end - start). Write an SQL query to get the values of fcode and the number of days that the fire was active for those fires where the land use type is "scrubland". Your solution must use a subquery and the keyword **IN**. (5p)

SOLUTION:

```
SELECT fcode, DATE_PART('day', end_date - start_date) AS Number_of_days
FROM Fire
WHERE fcode  IN (SELECT fcode
  FROM Extent
  WHERE land_use='scrubland');
```

5c. Write the corresponding triggers for automatically updating the total_area value when the content of the Extent table is modified. Note that a modification implies INSERT, UPDATE and DELETE operations. (6p)

Solution:

```
CREATE OR REPLACE FUNCTION InsertTotalExtent() RETURNS TRIGGER  AS $$
  BEGIN
    UPDATE Fire
     SET total_area = total_area + NEW.area
     WHERE fcode = NEW.fcode;
    RETURN NEW;
  END
$$ LANGUAGE 'plpgsql';

CREATE TRIGGER InsertTotalExtent AFTER INSERT ON Extent
FOR EACH ROW EXECUTE PROCEDURE InsertTotalExtent();

CREATE OR REPLACE FUNCTION UpdateTotalExtent() RETURNS TRIGGER AS $$
  BEGIN
    IF OLD.fcode <> NEW.fcode THEN
      UPDATE Fire
       SET total_area = total_area + NEW.area - OLD.area
       WHERE fcode = NEW.fcode;
      RETURN NEW;
    END IF;
END
$$ LANGUAGE 'plpgsql';

CREATE TRIGGER UpdateTotalExtent AFTER UPDATE ON Extent
FOR EACH ROW EXECUTE PROCEDURE UpdateTotalExtent();

CREATE OR REPLACE FUNCTION DeleteTotalExtent() RETURNS TRIGGER AS $$
  BEGIN
    UPDATE Fire
     SET total_area = total_area - OLD.area
     WHERE fcode = OLD.fcode;
    RETURN OLD;
  END
$$ LANGUAGE 'plpgsql';

CREATE TRIGGER DeleteTotalExtent AFTER DELETE ON Extent
FOR EACH ROW EXECUTE PROCEDURE DeleteTotalExtent();
```