

Sean Gallagher  
Homework 4

1 5.7

It is possible to generate an infinite set in relational calculus. An example would be  $\{S \mid \sim \text{Staff}(S)\}$  which means the set of all tuples that are not in the staff relation. To make it safe you have to add a restriction that all values that appear in the result must be values in the domain.

2 5.12 (b,f) DRC and RA only

RA

b  $\pi \text{ roomNo, hotelNo } \sigma \text{ price} < 20 \wedge \text{type} = \text{single} (\text{Room})$

f  $\pi \text{ roomNo, hotelNo, type, price, guestName } (\sigma \text{ hotelName} = \text{Grosvenor} ( \text{Hotel} \bowtie \text{Booking} \bowtie \text{Room} \bowtie \text{Guest}))$

DRC

b  $\{ \langle \text{roomNo, hotelNo} \rangle \mid \langle \text{roomNo, hotelNo} \rangle \in \text{Room} \wedge \text{price} < 20 \wedge \text{type} = \text{single} \}$

f  $\{ \langle \text{roomNo, hotelNo, type, price, guestName} \rangle \mid (\exists \text{ hotelNo, hotelName})$   
 $(\langle \text{hotelNo, hotelName} \rangle \in \text{Hotel} \wedge \text{hotelName} = \text{Grosvenor} \wedge$   
 $(\exists \text{ roomNo, hotelNo}) (\langle \text{roomNo, hotelNo} \rangle \in \text{Room} \wedge \text{room.hotelNo} = \text{hotel.hotelNo} \wedge$   
 $(\exists \text{ roomNo, hotelNo, guestNo}) (\langle \text{roomNo, hotelNo} \rangle \in \text{Booking} \wedge \text{booking.hotelNo} =$   
 $\text{hotel.hotelNo} \wedge \text{booking.roomNo} = \text{room.roomNo} \wedge$   
 $(\exists \text{ guestNo}) (\langle \text{guestNo, guestName} \rangle \in \text{Guest} \wedge \text{guest.guestNo} = \text{booking.guestNo}) \}$

3 5.16 DRC and RA only

RA

$\pi \text{ fName, lName, address } \sigma \text{ empNo} = \text{mgrEmpNo} (\text{Employee} \bowtie_{\text{employee.deptNo} = \text{Department.deptNo}} \text{Department})$

DRC

$\{ \langle \text{fName, lName, address} \rangle \mid (\exists \text{ empNo}) (\text{empNo} \in \text{Employee} \wedge (\exists \text{ mgrNo}) (\text{mgrNo} \in \text{Department} \wedge$   
 $(\text{Employee}.\text{deptNo} = \text{Department}.\text{deptNo}) \wedge (\text{empNo} = \text{mgrNo})) \}$

4 6.11

Select \* from Booking where dateTo = null;

5 6.13

Select AVG(price) from Room;

6 6.21

Select SUM(Room.price) from Room  
right join Hotel on (Room.hotelNo = Hotel.hotelNo)

where ((GETDATE() < dateFrom) or (GETDATE() > dateTo)) and Hotel.hotelName = 'Grosvenor Hotel';

#### 7 6.23

```
Select Hotel.hotelName COUNT(Room.roomNo) from Room
left join Hotel on (Room.hotelNo = Hotel.hotelNo)
group by Hotel.hotelName
where Hotel.city = 'London';
```

#### 8 7.5

View resolution merges the query on a view with the definition of the view producing a query on the base tables. This is done every time the dbms processes a query on the view. Basically it merges the query that has a view with the defining query. It has to merge each time the dbms runs a query with using that view.

#### 9 7.7

View materialization stores the view as a temporary table in the database which maintains the currency of the view and base tables are updated. This differs from view resolution in that view resolution has to merge every time it is called while materialization saves a temporary view that acts as a table. This saves resources.

#### 10 7.11 (a,c,e) Only create Room and Booking

Create Table Room(

    roomNo        integer

    Contraint roomNo Check (roomNo Between 1 and 100)

    hotelNo        integer

    Type          varchar(7) NOT NULL CHECK (type in ('single', 'double', 'family'))

    Price          decimal(10, 2)

    Primary Key (roomNo)

    Foreign Key (hotelNo) References Hotel(hotelNo)

)

CREATE DOMAIN noDoubleBook AS noDoubleBook

    CHECK(value in (select guestNo from Guest));

Create Table Booking(

    hotelNo        integer

    guestNo        noDoubleBook

    dateFrom      TIMESTAMP

    dateTo        TIMESTAMP

    roomNo        integer

    Primary Key (dateFrom)

    Foreign Key (guestNo)

    Foreign Key (hotelNo)

)

11 7.14

Create View Account

as select Guest.guestNo, Guest.guestName, Guest.guestAddress, Room.price \*  
(Booking.dateTo - Booking.dateFrom)

From Guest, Booking, Hotel, Room

Where Guest.guestNo = Booking.guestNo AND Room.roomNo = Booking.roomNo

And Booking.hotelNo = Hotel.hotelNo and Hotel.hotelName = 'Grosvenor Hotel'

And Booking.dateTo = CURRENT\_DATE;

12 8.6

A trigger defines an action that the database should take when some event occurs. Triggers can be used to enforce different types of constraints including referential integrity constraints, complex enterprise constraints, or to audit changed to data.

13 8.11 a

Create Trigger doubleRoomPrice Insert on Room

For each row when (new.type = 'double' AND new.price < 100)

14  $R(A, B, C, D, E, F, G), B \rightarrow ACDE, E \rightarrow F G$

Compute  $B^+, G^+$

$B^+ = \{A, C, D, E, F, G\}$

$G^+ = \{G\}$

15  $F = \{A \rightarrow D, AC \rightarrow B, C \rightarrow D, A \rightarrow BC\}$

find minimal cover of F

Make singleton right side

$F = \{A \rightarrow D, AC \rightarrow B, C \rightarrow D, A \rightarrow B, A \rightarrow C\}$

Make left side singleton

$C^+ = \{C, D\}$

$A^+ = \{A, D, B, C\}$  includes C

$F = \{A \rightarrow D, A \rightarrow B, C \rightarrow D, A \rightarrow B, A \rightarrow C\}$

Find redundancies

$A^+ = \{A, B, C, D\}$   $C \rightarrow D$

Minimal Cover

$F = \{C \rightarrow D, A \rightarrow B, A \rightarrow C\}$