Midterm Exam

October 22nd, 2013
10:00-11:30

CS425 - Database Organization

Results

Please leave this empty!

1.1 1.2 1.3 1.4  

Sum
Instructions

- Try to answer all the questions using what you have learned in class. Keep hard questions until the end.
- **When writing a query, write the query in a way that it would work over all possible database instances and not just for the given example instance!**
- The exam is closed book and closed notes!

Consider the following database schema and example instance for a flight information system:

### airline

<table>
<thead>
<tr>
<th>lCode</th>
<th>lName</th>
<th>country</th>
</tr>
</thead>
<tbody>
<tr>
<td>AE</td>
<td>American Airlines</td>
<td>USA</td>
</tr>
<tr>
<td>DAL</td>
<td>Delta Airlines</td>
<td>USA</td>
</tr>
<tr>
<td>AC</td>
<td>Air Canada</td>
<td>Canada</td>
</tr>
<tr>
<td>DHL</td>
<td>Lufthanse</td>
<td>Germany</td>
</tr>
</tbody>
</table>

### flight

<table>
<thead>
<tr>
<th>lCode</th>
<th>flightNumber</th>
<th>fromAirport</th>
<th>toAirport</th>
<th>lengthHours</th>
</tr>
</thead>
<tbody>
<tr>
<td>AE</td>
<td>367</td>
<td>JFK</td>
<td>DEL</td>
<td>8</td>
</tr>
<tr>
<td>AC</td>
<td>555</td>
<td>ORD</td>
<td>YYZ</td>
<td>2</td>
</tr>
<tr>
<td>AC</td>
<td>73</td>
<td>YYZ</td>
<td>DEL</td>
<td>16</td>
</tr>
</tbody>
</table>

### airport

<table>
<thead>
<tr>
<th>aCode</th>
<th>aName</th>
<th>country</th>
<th>city</th>
</tr>
</thead>
<tbody>
<tr>
<td>JFK</td>
<td>John F Kennedy International</td>
<td>USA</td>
<td>New York</td>
</tr>
<tr>
<td>DEL</td>
<td>Indira Gandhi International Airport</td>
<td>India</td>
<td>Delhi</td>
</tr>
<tr>
<td>ORD</td>
<td>Chicago O’Hare International</td>
<td>USA</td>
<td>Chicago</td>
</tr>
<tr>
<td>YYZ</td>
<td>Lester B. Pearson International</td>
<td>Canada</td>
<td>Toronto</td>
</tr>
</tbody>
</table>

### schedule

<table>
<thead>
<tr>
<th>lCode</th>
<th>flightNumber</th>
<th>date</th>
<th>time</th>
<th>capacity</th>
</tr>
</thead>
<tbody>
<tr>
<td>AE</td>
<td>367</td>
<td>2013-05-04</td>
<td>12:40</td>
<td>245</td>
</tr>
<tr>
<td>AC</td>
<td>555</td>
<td>2013-07-07</td>
<td>02:00</td>
<td>80</td>
</tr>
<tr>
<td>AC</td>
<td>73</td>
<td>2013-08-08</td>
<td>04:00</td>
<td>100</td>
</tr>
<tr>
<td>AC</td>
<td>73</td>
<td>2013-08-09</td>
<td>04:00</td>
<td>100</td>
</tr>
</tbody>
</table>

### booked

<table>
<thead>
<tr>
<th>customer</th>
<th>lCode</th>
<th>flightNumber</th>
<th>date</th>
<th>time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Peter Fullbright</td>
<td>AE</td>
<td>367</td>
<td>2013-05-04</td>
<td>11:00</td>
</tr>
<tr>
<td>Alice Bobsen</td>
<td>AC</td>
<td>555</td>
<td>2013-07-07</td>
<td>02:00</td>
</tr>
</tbody>
</table>

Hints:

- Attributes with grey background form the primary key of a relation (*e.g*, lCode for relation airline)
- The attribute fromAirport of relation flight is a foreign key to aCode of relation airport. The attribute toAirport of relation flight is a foreign key to aCode of relation airport. The attribute lCode of relation flight is a foreign key to lCode in relation airline.
- The attributes lCode and flightNumber of relation schedule form a foreign key to lCode and flightNumber in relation flight.
- The attributes lCode, flightNumber, date, time of relation booked form a foreign key to relation schedule.
- All foreign keys have been created with the **CASCADE** option.
Part 1.1  Relational Algebra (Total: 30 Points)

Question 1.1.1  (7 Points)

Write a relational algebra expression that returns the names of all airlines from Germany.

Solution

\[ \pi_{\text{Name}}(\sigma_{\text{country}=\text{Germany}}(\text{airline})) \]

Question 1.1.2  (7 Points)

Write a relational algebra expression that returns the total capacity of all scheduled flights per airline. Return pairs of airline codes (lCode) and total capacity. For example, if American Airlines flight 555 is scheduled for two times, both with capacity 100, then this would be counted as 200 when calculating the total capacity for American Airlines.

Solution

\[ \text{lCode} \bowtie \text{sum(capacity)}(\text{schedule}) \]
Question 1.1.3  (7 Points)

Write a relational algebra expression that returns cities from the USA that have an airport.

Solution

\[ \pi_{\text{city}}(\sigma_{\text{country}='\text{USA'}}( \text{airport})) \]

Question 1.1.4  (8 Points)

Write a relational algebra expression that returns the names of all customers together with the destination city of flights they have booked.

Solution

\[
\begin{align*}
\text{bookCity} & \leftarrow \text{booked} \bowtie \text{flight} \bowtie \text{toAirport=aCode} \text{ airport} \\
q & \leftarrow \pi_{\text{customer,city}}(\text{bookCity})
\end{align*}
\]
Part 1.2 SQL - DDL (Total: 16 Points)

Question 1.2.1 (8 Points)
Write an SQL statement that creates a new relation `flightdelay` that records delay times (delay in minutes) for scheduled flights. For example, we would like to be able to store that flight AC 555 scheduled for 2013-07-07 at 02:00 is delayed by 15 minutes.

Solution

```sql
CREATE TABLE flightdelay (  
    lCode VARCHAR(3),  
    flightNumber INT,  
    date DATE,  
    time TIME,  
    delay INT,  
    PRIMARY KEY (lCode, flightNumber, date, time),  
    FOREIGN KEY (lCode, flightNumber, date, time) REFERENCES schedule  
);  
```

Question 1.2.2 (8 Points)
Write a relation `securityAlert`. Each alert is for a fixed airport. An alert consist of a message, a severity level (one of 'RED', 'ORANGE', 'GREEN'), and a time when the alert was issued. Alerts are identified by their time and the airport they have been issued for.

Solution

```sql
CREATE TABLE securityAlert (  
    aCode VARCHAR(3) FOREIGN KEY REFERENCES airport,  
    message VARCHAR(4000),  
    severity VARCHAR(6),  
    issueTime TIMESTAMP,  
    PRIMARY KEY (aCode, issueTime),  
    CHECK (severity IN ('RED', 'ORANGE', 'GREEN'))  
);  
```
Part 1.3  SQL - Queries (Total: 39 Points)

Question 1.3.1  (7 Points)

Write an SQL query that returns airline code (lCode) of flights that are scheduled between 10:00 and 11:00.

Solution

```
SELECT lCode
FROM schedule
WHERE time BETWEEN '10:00' AND '11:00';
```

We did not ask for it, but DISTINCT would be ok.

Question 1.3.2  (7 Points)

Write an SQL query that returns the airline name (lName) and flight number for flights that are less than 6 hours long.

Solution

```
SELECT lName, flightNumber
FROM airline NATURAL JOIN flight
WHERE lengthHours < 6;
```
Question 1.3.3  (9 Points)

Write an SQL query that returns flights (airline code and flight number) that start from an airport in USA and land in an airport in India.

Solution

```sql
SELECT lCode, flightNumber
FROM airport s, flight f, airport d
WHERE s.country = 'USA' AND f.fromAirport = s.aCode
    AND f.toAirport = d.aCode AND d.country = 'India';
```

Question 1.3.4  (8 Points)

Write an SQL query that returns for each time and date, the number of flights starting at that date and their maximum capacity. For example, if there are two flights that start at July, 7th 2013 at 10:00, one with capacity 60 and one with capacity 200, then one result tuple of the query would be (2,200,2013-07-07,10:00).

Solution

```sql
SELECT count(*) AS numFlights, max(capacity) AS maxC, date, time
FROM schedule
GROUP BY date, time
```
Question 1.3.5  (9 Points)

Write an SQL query that returns flights that are over-booked. A flight is considered over-booked if the number of bookings for this flight exceeds the capacity of this flight.

Solution

```sql
SELECT *
FROM schedule s
WHERE capacity < (SELECT count(*)
    FROM booked b
    WHERE s.lCode = b.lCode AND s.flightNumber = b.flightNumber
    AND s.data = b.date AND s.time = b.time);
```

Or

```sql
SELECT lCode, flightNumber, date, time, capacity
FROM schedule
NATURAL JOIN booked
GROUP BY lCode, flightNumber, date, time, capacity
HAVING capacity < count(*);
```
Part 1.4  SQL - Updates (Total: 15 Points)

Question 1.4.1  (7 Points)

Write an SQL statement that deletes all flights from JFK to DEL. Recall that all foreign key constraints were created with the CASCADE option.

Solution

DELETE FROM flight
WHERE fromAirport = 'JFK' AND toAirport = 'DEL';

Question 1.4.2  (8 Points)

Write an SQL statement that reduces the capacity of flights to 200 if they are scheduled for a time in the morning (before 12:00 midday) and currently have a capacity above 200.

Solution

UPDATE schedule SET capacity = 200
WHERE capacity > 200 AND time < 12:01;