ECE 456 / ECE 656 – Winter 2013
Assignment #3 – SQL APIs Applied

Overview

For the assignment questions below, follow these steps in deriving your solutions.

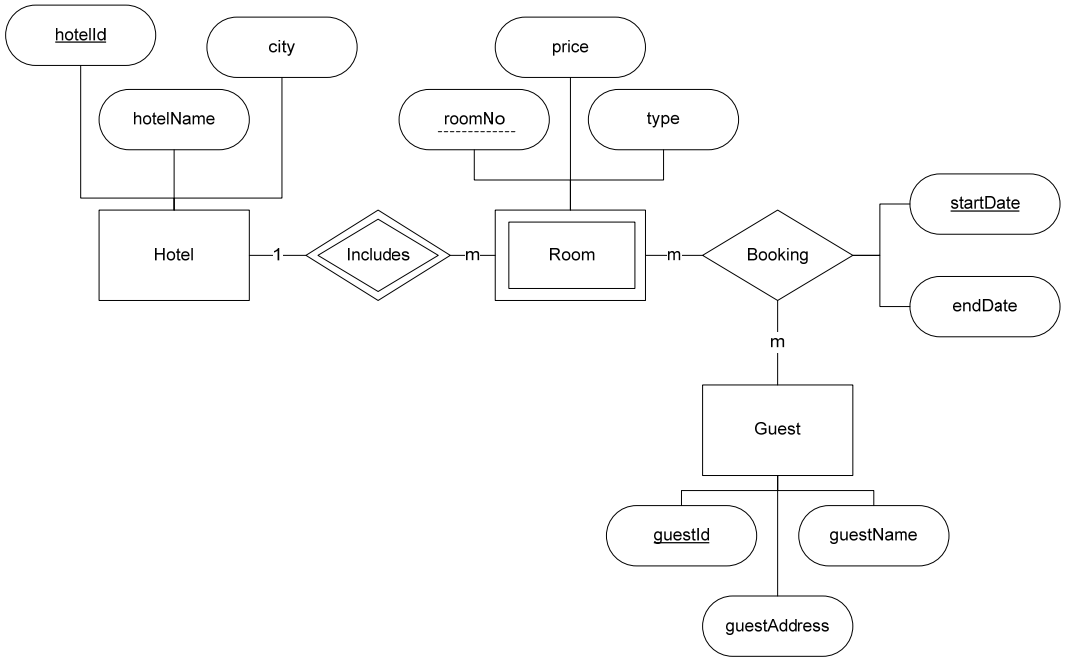
Step 1. Starting with your SQL DDL code from Assignment #2, implement the given ER model in the DBMS of your choice. When selecting the DBMS, select one that supports the API of your choice (e.g., JDBC), and has the corresponding connection drivers available.

Step 2. Implement the specified functionality in the host language using the API of your choice (e.g., implement the functionality in Java using JDBC as the connection API). Include the source code and unit test cases with your submission. **You may find it helpful to use the queries that you have designed for Assignment #2 when writing your API code.**

Step 3. For each module in your report, include the output from your program (e.g., using the Print Screen key) showing a typical execution scenario, where important steps are clearly marked and illustrated.

(Bonus, up to 10 marks) Step 4. Implement a testing harness for your code using a specialized unit-testing framework such as dbUnit or sqlUnit, and use the harness to test your code. In your report, discuss how the harness was constructed, and attach the test cases to your submission.

Hotel Database Revisited [70 marks]



Module 1: Guest Registration [15 marks]

1. A booking agent registers a new guest and enters their information, including the name and address. The guestID is auto-generated, either through the API or by the DBMS itself.
2. The agent can also update the information for an existing guest (except for the guestID), and can also delete the entry for a specific guest if requested.

Module 2: Booking Query [20 marks]

1. Once the guest information has been entered, the agent can then query hotels for available rooms on specified dates. That is, the agent enters one or all of the following: startDate, endDate, hotelName, city, room price, and room type. For any entry that is left blank, the corresponding condition is not applied (e.g., if city is left blank, all cities are considered).
2. The booking query returns all of the rooms that are available based on the information entered, and the query displays the hotelID, hotelName, city, roomNo, price, and type.

Module 3: Booking Registration [20 marks]

1. Once an available room has been found, the agent then registers a new booking by entering the hotelID, roomNo, guestID, startDate, and endDate. If the booking was successful, the system returns the bookingID that the agent can give to the customer as confirmation. The bookingID is auto-generated, either through the API or by the DBMS itself.
2. The booking registration should also ensure that bookings for the same room do not overlap.

Module 4: Room Maintenance and Billing [15 marks]

1. The administrative staff can use a separate interface to list all the arrivals (i.e., the current date equals startDate) and all the departures (i.e., the current date equals endDate) on a specific day to ensure that the rooms are maintained before and after guest arrivals, respectively.
2. Finally, the administrative staff can print bills for the departing guests (display on screen is sufficient), based on the number of days stayed and the applicable room price. Each bill printed should also be logged into the billing log for the accounting purposes.

Deliverables

A printed document submitted in class that includes a solution for each of the assignment questions. For each question, you need to provide your host language code (e.g., your Java code) and the unit test cases along with the output, as described in the Overview section.

You are permitted to work in groups of up to three students. Note that you may be contacted by your course TAs to demonstrate the running of your code if necessary. If applicable, you can also post a running demo of your code that is Web accessible (as a URL) that TAs can access when grading your assignment.

The assignment is due in class on Tue Mar 26th by 10:55am. Late submissions will not be accepted.