NYU Tandon School of Engineering
Computer Science and Engineering
CS6083, Fall 2023

October 13, 2023

Problem Set #2

(due 10/24 at 11:59 pm)

Problem 1: In this problem, you have to write SQL and RA queries for a database that manages information about comedy shows. Here are the tables:

Comedians (comedian_id, first_name, last_name, bio, phone)
Venues (venue_id, name, address, contact_info, capacity)
Events (event_id, venue_id, date, start_time, end_time)
Events_Comedians (event_id, comedian_id)
Tickets (ticket_id, event_id , ticket_type, price, num_tickets)
Customers (customer_id, first_name, last_name, email, phone_number)
Reservations (res_id, customer_id , ticket_id , res_date, tickets_count)
Reviews (review_id, event_id , customer_id , rating, review_text)

The Comedy Show Management System schema encompasses tables for Comedians, Venues, Events, Tickets, Customers, Reservations and Reviews. Each comedian is uniquely identified with a comedian_id and information about individual comedians, including their bio and contact phone number. Venues are identified by a venue_id, and have a name, address, contact_info, and capacity. Each event is identified by an event_id, is scheduled at a venue, and involves one or more comedians. A ticket has a unique ticket_id, an event_id to associate it with a specific event, a ticket_type to distinguish between different ticket categories (e.g., VIP, General Admission), a price, and the number of tickets available for each ticket type. (Thus, there is one entry for each event and each type of ticket.) Customers store customer data. Each reservation is uniquely identified by a res_id, and also stores which customer made the reservation, which ticket the reservation is for, the date when the reservation was made, and the number of tickets that were reserved (since a person may need more than one ticket, for family or guests). The Reviews table captures customer feedback, including ratings on a scale of 1 to 10.

(a) Draw an ER diagram that is consistent with this relational schema. Identify any weak entities within the schema and specify the cardinalities for all relationships.

(b) Create the schema in a database system. Choose appropriate attribute types, and define primary keys, foreign keys, and other constraints. You may use any relational database system, as long as it supports basic SQL, views, and some sort of triggers. Write the create table statements.

(c) Write the following SQL queries and execute them on your database. Show the queries and the screenshot of the results:
    1. List all events along with the total number of tickets sold (reserved) for each event.
    2. Rank comedians based on the number of events they participate in. You should list the comedians' IDs and names from highest to lowest number of appearances.

3. List all pairs of customers and comedians where the customer made reservations for more than 5 different events in which the comedian appeared.
4. Find the top 5 comedians with the highest average rating based on their reviews, among those comedians who have appeared in at least 5 different events.

(d) Write the following SQL delete queries and execute them on your database. Show the delete queries.
1. Delete customers who have not attended any events after 2020 from the Customers table.
2. Delete events that happened before 2022 and had less than 50 attendees from the Events table.

(e) Write the following SQL update and insert queries and execute them on your database. Show the update queries.
1. Update the end_time of all future events held in venues with a capacity greater than 500 to be one hour later, unless there is another event starting within an hour after the original end_time of the event.
2. Suppose the company would like to thank its most loyal customers, by giving them free tickets for an event. Write a query to insert into the Reservation table a reservation for one VIP ticket, for all customers who have reserved more than 20 tickets during 2023, for the first future event that has enough tickets of type VIP available for all of these customers.

(f) Write expressions in Relational Algebra for queries 1 - 4 in part (c).

(g) Consider adding an attribute called availability to the Tickets table, which stores the number of tickets that are still available (i.e., have not been reserved yet). This attribute is set to num_tickets when a record in Tickets is first created. Create a trigger that automatically checks if the number of tickets requested in a new reservation is still available. If no, the new reservation should be rejected; otherwise, the availability column in the Tickets table should be updated appropriately.

(h) Create a view that shows for each customer and each year the total number of tickets they have reserved during the year. The view should contain as attributes the customer_id, the year, and the total number of tickets.

(i) Utilizing the previously created view, output all users who have reserved more than 100 tickets total (meaning, adding across all years).

Problem 2: Imagine you are assisting Marina Inc., a boat rental company, in developing their new database to streamline boat rentals. Marina Inc. specializes in offering canal boat rental services in the waterways of France. They also offer overnight docking facilities at various locations. The service is designed to provide a unique vacation experience, allowing customers to navigate through scenic canals and to explore French towns and villages along the way. The database should capture details about customers, boats, and overnight docking information.

Every customer who wishes to rent has to provide some basic details: their first name, last name, phone number, and email. Each customer acts as the primary renter but can bring along multiple guests, and the total number of guests on the rented boat should be recorded for each boat trip.

Each boat available for rental is identified by a unique boat ID. Boats belong to classes such as luxury, standard, or economy. Each boat can have one or more descriptive tags, such as "family friendly," "fishing equipment," or "luxury experience." Tags are drawn from a short catalog of available tags defined by the boating company. For safety reasons, each boat also has a specified maximum capacity.

The total rental price is determined by several factors, in particular the boat's class, the season, and the total number of people onboard. For simplicity, we assume that the season only depends on the starting date of the rental, and in particular on the month in which it started. Thus, if a boat is rented from May 20 to July 10 for a total of 41 days, and May is considered off peak season, then the entire rental is charged as off peak. The total rental price then consists of a daily base price that depends on the boat's class and the season, and a daily per-person charge that is added, which also depends on the boat's class and the season. In the above case, for say 4 passengers, this would be 41 * (daily base price + 4 * daily per-person charge). You should store in your schema which months are peak or off-peak, and also the daily base price and per-person charge depending on the boat and season.

For simplicity, we assume that reservations cannot be canceled and that all customers show up for their reservation. We also do not model payment and credit card issues.

During the rental, each boat is required to dock at one of a number of docking stations overnight. Each docking station is identified by a unique ID, and has a name and a location. Docking stations are located in various places along the system of canals and lakes, so that customers can explore different towns and locations during their cruise. Each docking station also has a maximum number of boats that can dock overnight. The database must record at which docking station a boat is located for each night.

(a) Design a database for the above scenario using the ER model. Draw the ER diagram, show the cardinalities of all relationships, and identify primary keys and any weak entities. List any assumptions you have as well.

(b) Convert your ER diagram into a relational schema. Identify all tables, attributes, primary keys, and foreign keys.

(c) Write SQL statements for the following questions or updates. If you cannot answer a query using your schema, then you have to modify your solutions in (a) and (b) appropriately.

> (i) Find customers who have rented both class "luxury" and "economy" boats, but have never rented a boat with a "fishing equipment" tag.
>
> (ii) For each day in August 2023, list any docking stations that were completely full overnight.
>
> (iii) Find all boats that can accommodate at least 6 people and that are available from October 24 to November 7, 2023. Output the boat ID, the class, and the total cost of renting it for 6 people during that time.

(iv) Display the names and IDs of all customers who arrived at the docking place after 10pm in at least two evenings during a single rental.

(v) Given a maximum budget of 2,000 dollars, list all boats that are available for a 10-day rental for 4 guests during August of 2022. You should output the boat ID, the class, the starting date of the rental, and the total cost of the rental.

(d) Create tables in the database system, and insert some sample data (5-10 tuples per table, but choose an interesting and meaningful data set, so that queries do not output empty results). Submit screenshots of what your tables look like after you inserted the data. Then execute the queries in (c) and submit the screenshots of the queries and outputs.