Illinois Institute of Technology
CS425: Database Organization Spring 2013

Some Important Dates:

Final Exam

(according to university final exam schedule, subject to change):
THURS – May 9, 2 pm to 4 pm

http://www.iit.edu/registrar/important_dates/final_exam_schedule.shtml

Final Grades Due at Noon of Wed, May 15, 2013

Project Due Date: before the week of May 10, will be decided later

Project:

You have been hired to design and build a database application that supports the business needs of Sunsational Vacations, a travel agent specializing in vacation getaways. The organizational requirements are described below.

The application needs to store, update, delete, and query the data. The user who uses your application does not know SQL. Your application needs to have an interface for data entry, as well as updating, deleting, and querying the data stored in the database. This should be a complete application so that all transactions can be done through the user interface and the results are presented to the user via the application interface. A web-based interface is preferable, but not necessary (see below in the discussion of deliverable).

Note that some requirements may be implicit in the sorts of queries that will need to be implemented, so consider carefully the required functionality when designing your database and user interface.

Your task in this assignment is to design an application to support the given application requirements, using an SQL database as a back-end. Specific deliverable are described at the end of this document. Read through the requirements carefully, and email any questions to me as soon as possible, if anything is not clear.

- Sunsational Vacations needs to store and query the demographic information of its clients. These are: a client identifier, phone number, email address, first name, last name, date of birth, zip code, and gender.

- Sunsational Vacations also needs to store information on its agents. For each agent the following are stored: an agent identifier, phone number, email address, first name, last name, date of birth, zip code, and gender.
- Additionally, Sunsational Vacations needs to store each agent's position within the company. Each agent is either a "travel agent" or an "agent manager". An agent manager can do anything that a travel agent can, and has additional abilities as well (as explained later).
- Sunsational Vacations has a predefined list of destination cities (in various countries) where its clients may travel to. Additional cities may be added at any time by an agent manager. Within each destination city, Sunsational Vacations lists at least one resort (possibly many), each of whose information must be stored. This includes: the resort's name, its city and country, address, and telephone number. Also, the price of one night's stay per person in the resort hotel must be stored; each hotel may have several room types with different prices. Of course, prices will vary among the different resorts as well.
- Sunsational Vacations also has amenities at its resorts which are offered free to guests. However, not all amenities are offered at all resorts. Thus, a list of which amenities are available at each resort must be maintained. The list of possible amenities is: gym, wireless internet, spa, kayaks, same-day laundry/dry cleaning, day care facilities, and entertainment. This list may be added to or deleted from at any time by an agent manager.
- For each resort, we maintain a score of how luxurious the resort is. Thus, we keep a "Sun Rating", where one sun indicates a value resort, two suns indicate a mid-range resort, and three suns indicate a high-end resort.
- The database must also store client bookings, keeping all the relevant information about the trip (which resort, the booking date, the arrival and departure dates, the room type reserved, and the agent who booked the trip).

Besides the above specified requirements, your database must support all the transactions and queries listed below.

1. Should be able to insert, update, and delete all the pre-defined lists (only by an agent manager).
2. Should be able to query any data in database.
3. Should be able to query the data based on certain criteria.
4. Given a client name as the input, find what trips were booked by

that client (give all relevant information about the trip including which agent booked each trip).

5. Given the name of a resort (selected by the user), what is the Sun Rating of that resort?
6. Find the total revenue from hotel room bookings (sum of all bookings' nightly rates for all nights plus all inclusive fees), for a given agent, across all agents, or for a given country or city.
7. Which resorts offer all possible amenities?
8. List the number of resorts in a given city or country.
9. For a client, list all resorts which that individual has vacationed at, as well the number of days of their trip.
10. List all amenities which contain a certain string (provide by a user) exactly in their description.
11. List the users who have stayed all-inclusive at any hotel within the last 3 months.
12. What is the average number of days booked per trip? For a given client?
13. Which agent booked the largest number of bookings?
14. Which clients have booked the largest numbers of trips in a given period? Which have spent the most money in some period?
15. List all agents and the number of clients who have booked with them in descending order.

Project Demo & Deliverable:

Each group will give a 25 minutes demo of a working project as described above. All group members must be present in the demo. The demo should introduce each member of the group and the roles that they play in the project development. It should include a conclusion slide summarizing the results.

In the course of your demo it is your responsibility to have necessary data available in your database to be able in the allocated time to give a demo to show the functionality of your application. Make sure that you have this data in your system and an organized and systematic test plan that you can demonstrate that your database application is functional and supports all the requirements within the allocated time frame in order to receive the full points! You will also be asked to enter new data and perform new transactions. The first thing you have to provide before your actual demo is your documentation. You need to hand in your documentation (hard copy) at the beginning of the demo to be able to start your demo. Your application must be functional, satisfying all the requirements and the specified queries.

Documentation:

- Database Design Document (submit a printout at your demo): Document your database design (just the schema and needed explanations) and all assumptions made in your design. Your database tables MUST be loss-less and at least in 3rd normal Form (3NF). List all the schema and underline the primary keys on each schema. Underneath each schema specify all candidate and foreign keys. All schema must be given on a single page for readability.
- Test Plan Document (submit a printout at your demo): to show how you have tested all the requirements of the system (hard copy). Via your test document you need to show that you have tested successfully each requirement and query. This is a list of all requirements (all requirements listed in the project description and also queries) your application has to satisfy (first column), your test path/steps (second column), the expected result (third column), the generated result (fourth column), and the last column specifies if the test was successful or not. Make sure that the result is reported correctly, in order not to lose points. You must provide all the necessary test data for the above tests.
- A User Manual that describes how to build and execute the code and test data that you submitted

The soft copy of your application and all documents has to be uploaded onto blackboard by group leader via digital dropbox. Please list out your group number, names of members when submitting your project deliverable.

The design document and the test document should both be printed and handed in at the start of your demo time.

Project Points Distribution:

- Database Design Document: 10 pts.
- Demo Presentation: 10 pts.
- Answering Questions at the Demo: 10 pts.
- Test Plan Document: 10 pts.
- Project functionality: 60 pts.