Hotel Reservation System

In this project you will be creating a hotel reservation system that supports creating reservations, querying for customer information, and running report. You will be utilizing a MySQL database to store customer and reservation info as well as transactions to customer credit cards.

This is an individual or group(at most 2) project. You will need to version your project as a Github repo named “PA1”. Please include a README file with your name and that of your group member. If you are working in a group you only need to maintain a single repo. The project due date is March 2nd.

The structure of your code and your database is up to you. The only requirements for the structure of the project is that you must implement all the requirements outlined below and you must use Java servlets and Mysql as discussed in class. You will have 2 java processes at work. A client process will send requests to a server which will process requests via servlets. The Mysql database will store all persistent data. The table structure is up to you but be sure to store each field in the nonfunctional requirements and take into account foreign key constraints (ie deleting a customer should delete all that customer’s transactions). All input will be via console input.

Nonfunctional Requirements

- Customer Information
    - First name, last name, phone number, billing address, billing city, billing state, billing zip, customer id(auto generated), checkin date, checkout date
- Room Information
    - Room type (Single, Double, Presidential), Room price, current occupant, room number.
    - Singles cost 100$, Doubles cost 150$ and Presidential costs 300$.
    - There are 40 Single rooms, 50 Double rooms, and 10 Presidential Rooms.
- Transactions
    - Transaction id, customer id, room id, amount, credit card number, expiration date

Functional Requirements

- A main menu will allow the customer service rep to enter the following commands. Commands are case-insensitive.

- CREATE CUSTOMER - creates a new customer in the customer table. Gather all information for the customer. After the customer is created display the customer id generated (using auto-incremented id). For any errors display them.
- RESERVE ROOM - reserves a room for a given customer id. This will take the customer id as well as a room number. A success message is displayed when the room is successfully reserved else display an error.
- CREATE PAYMENT - creates a payment transaction for a given customer id and room id. The customer service rep will enter in all payment information. Upon success a transaction id is displayed else display an error. The amount of the transaction is entered by the customer service rep as well meaning that multiple transactions can be processed for the same customer and room.
- GET CUSTOMER [CUSTOMER_ID] - Gets the customer with the specified id and prints that customer's information.
- GET CUSTOMERS BYNAME [CUSTOMER_NAME] - Returns a list of customers who have the same first or last name specified by the input parameter. This list only contains the customer last name, first name, id, and phone number.
- GET CUSTOMERS CURRENT - Returns a list of customers who are currently checked into the hotel with same output as above.
- GET TRANSACTIONS [CUSTOMER_ID] - Returns a list of transactions for a specified customer. The list contains the transaction id, amount, and customer first name, last name.
- GET VACANCIES - Returns a list of all vacant rooms including their room types.
- GET RESERVATIONS - Returns a list of all rooms currently occupied by a customer. Include the customer first name/last name in the output.

As with most requirements some things have been left obscure. Please ask questions in or outside of class if something is unclear.