

# **Hotel DBMS**

## **Assignment 9**

November 2025

### **Group 7**

Jack Nguyen  
Michelle Nguyen  
**Kemble Xie**

<b>Overview.....</b>	<b>3</b>
<b>Requirements &amp; Setup.....</b>	<b>3</b>
Downloading the Project from GitHub.....	3
How to Compile.....	3
How to Run.....	4
<b>Launching the Application.....</b>	<b>4</b>
<b>Feature Guide.....</b>	<b>5</b>
Drop Tables.....	5
Create Tables.....	5
Populate Tables.....	5
List All Guests.....	5
Search Guest by Name.....	6
List Reservations.....	6
Update Guest Email.....	6
Delete Guest.....	6
Exit Application.....	6

## Overview

This project is a Java Swing GUI application backed by a SQLite database (hotel.db).

The system allows users to:

- Create all database tables
- Populate the database with sample hotel data
- View and search guest information
- Display reservation information
- Update guest email addresses
- Delete guest records
- Execute multiple database operations through a simple GUI

The system is fully self-contained. No Oracle, VPN, or server setup is required. All data is stored locally.

---

## Requirements & Setup

### Downloading the Project from GitHub

You can download the required files from GitHub:

<https://github.com/kembletmu/CPS510>

Place all the following files in the same folder:

- Main.java
- DB.java
- SQL.java
- Queries.java
- sqlite-jdbc-3.51.0.0.jar

When the program runs, it automatically generates:

- hotel.db (SQLite database file)
- 

## How to Compile

Open a terminal inside the project folder and run:

`javac -cp ".;sqlite-jdbc-3.51.0.0.jar" *.java` (for Windows)  
`javac -cp ".:sqlite-jdbc-3.51.0.0.jar" *.java` (for Mac/Linux)

---

## How to Run

To launch the graphical interface, run:

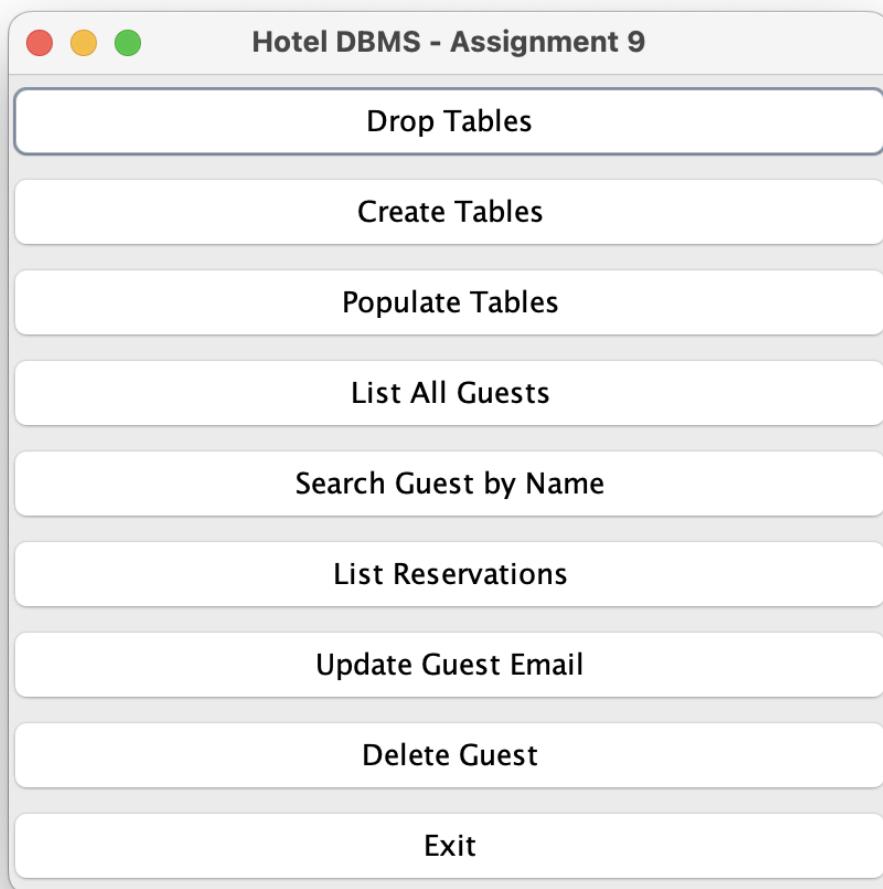
`java -cp ".:sqlite-jdbc-3.51.0.0.jar" Main` (for Windows)

`java -cp ".:sqlite-jdbc-3.51.0.0.jar" Main` (for Mac/Linux)

---

## Launching the Application

Once you run the program, the main menu window appears:



Each button corresponds to one of the system's features.

---

# **Feature Guide**

## **Drop Tables**

Deletes all tables from hotel.db.

Use this before creating a fresh database or reloading data.

A message will confirm the operation.

---

## **Create Tables**

Creates all tables defined in the BCNF schema:

- ProvinceCountry
- PostalAddress
- Guests
- Staff
- RoomType
- Rooms
- Reservations
- Invoices
- ServiceTypes
- ServiceRequests
- RoomAssignment

A pop-up message confirms that the tables were created.

---

## **Populate Tables**

Inserts the hotel data into the new tables.

A message will confirm that the data was inserted.

---

## **List All Guests**

Prints all guest records to the console.

Includes guest ID, name, email, and phone number.

---

## **Search Guest by Name**

Prompts the user to enter a name (or part of a name).

Search is case-insensitive.

Results are shown in the console.

---

## **List Reservations**

Displays reservations joined with guest names and room numbers.

---

## **Update Guest Email**

Prompts for:

1. Guest ID
2. New email address

Updates the corresponding record.

Confirmation appears via pop-up.

---

## **Delete Guest**

Prompts for a guest ID and deletes the record if it exists.

---

## **Exit Application**

Closes the program immediately.