

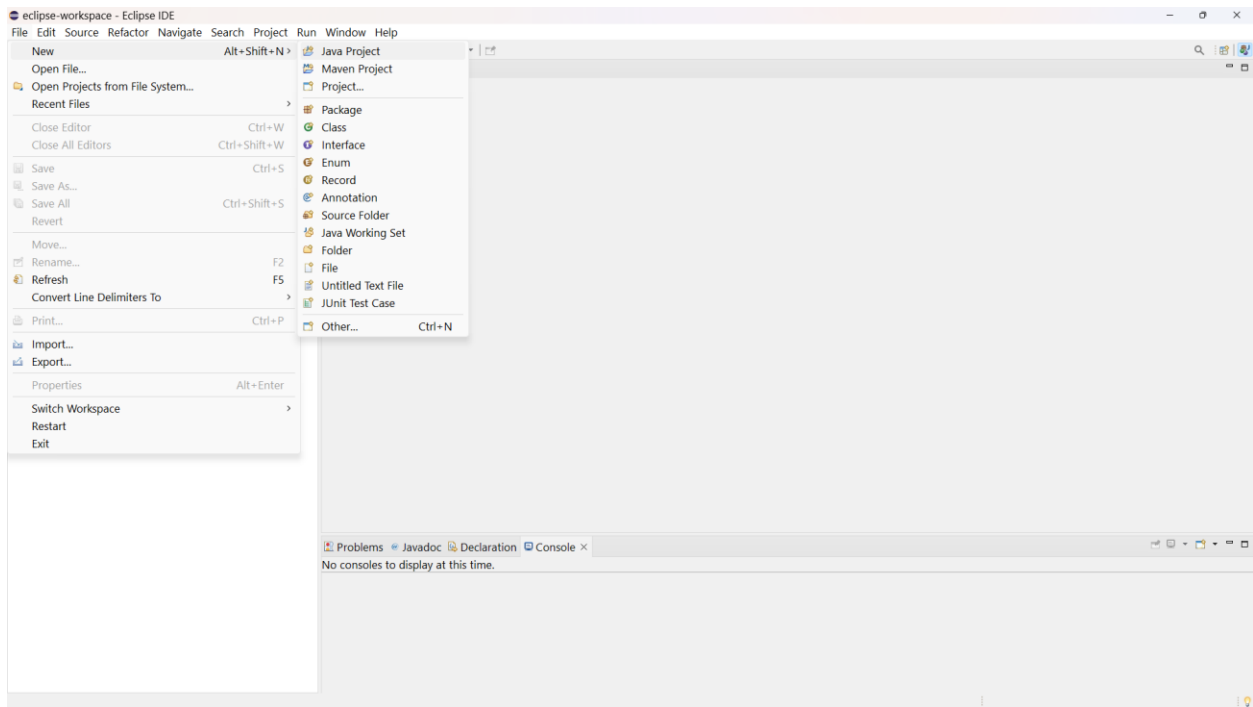
# Hotel Room Reservation using JAVA.

## INTRODUCTION AND PROJECT DESCRIPTION

Hotel Booking application is a desktop based application used to book a reservation, update a reservation to view the reservation and delete the reservation. This application is built in java for the backend and java awt swing for the user interface. With the help of this application, we can book an reservation by providing details like name, number, email and type of the service needed.

## RUNNING INSTRUCTIONS AND SCREENSHOTS

Using Eclipse to create a new java project as shown below.



Click finish to proceed.

New Java Project

Create a Java Project

Create a Java project in the workspace or in an external location.

Project name:

☒ Use default location

Location:  [Browse...](#)

JRE

☒ Use an execution environment JRE:

☐ Use a project specific JRE:

☐ Use default JRE 'jdk-17.0.6.10-hotspot' and workspace compiler preferences [Configure JREs...](#)

Project layout

☐ Use project folder as root for sources and class files

☒ Create separate folders for sources and class files [Configure default...](#)

Working sets

☐ Add project to working sets [New...](#)

Working sets:  [Select...](#)

Module

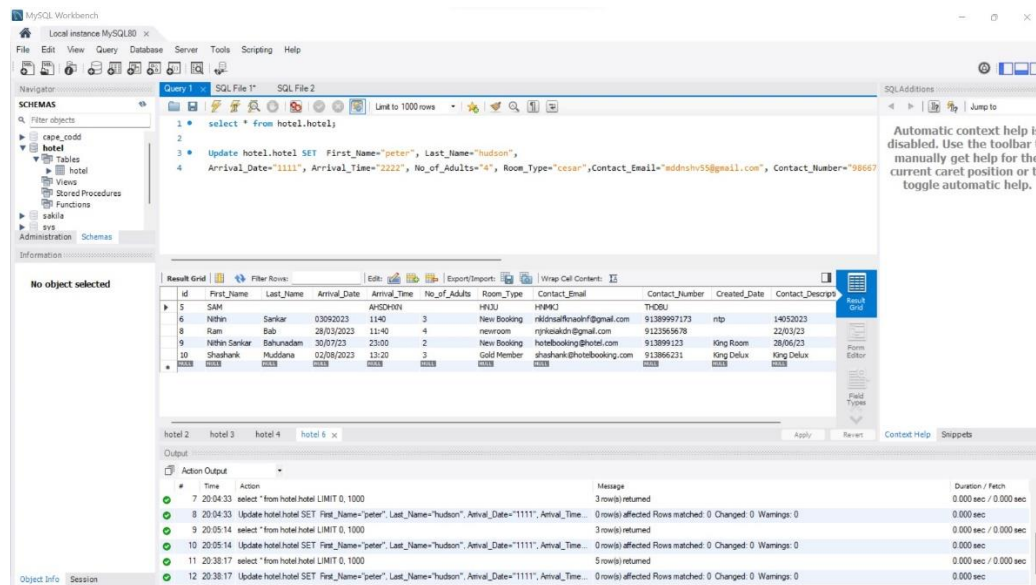
☐ Create module-info.java file

Module name:

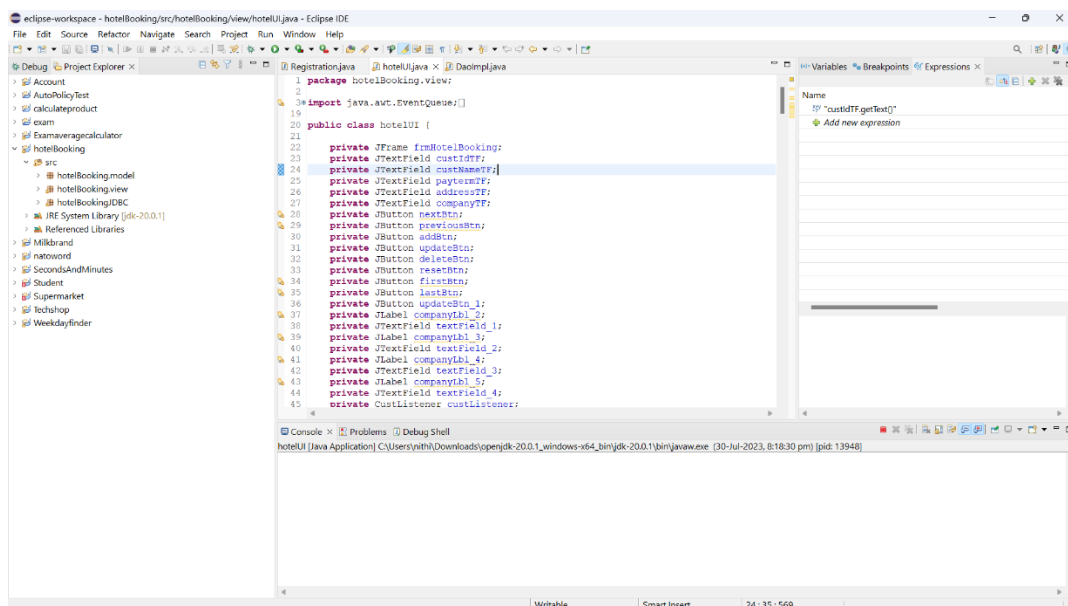
☐ Generate comments

[?](#) [< Back](#) [Next >](#) [Finish](#) [Cancel](#)

## Create a Database using mySQL



Click the Run as Java program button in hotelUI.java, which is where our main class is located, to launch the program.

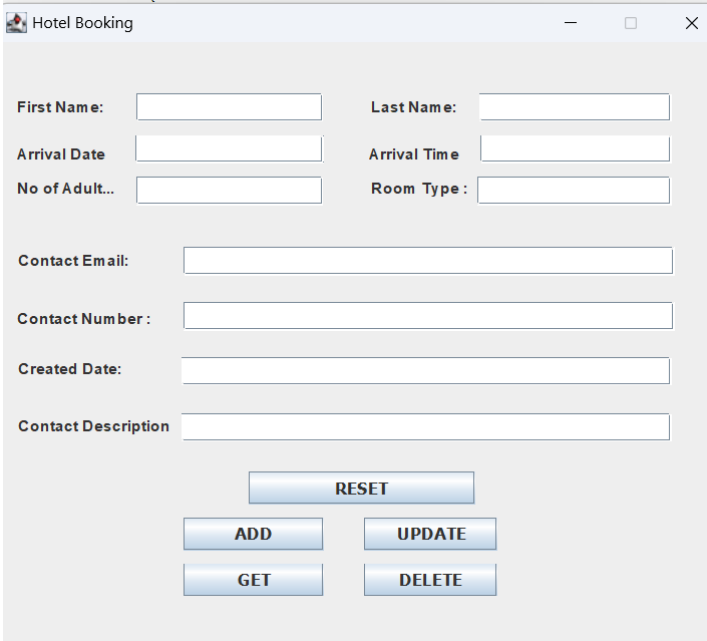


Open the application as shown in figure.

We have added four services and Reset in the application:

1. Add
2. Update
3. Get
4. Delete
5. Reset

1. In order to reserve a room at a hotel, we must enter information and click on ADD.



The screenshot shows a window titled "Hotel Booking" with a standard Windows-style title bar (minimize, maximize, close buttons). The form contains the following fields and labels:

- First Name:
- Last Name:
- Arrival Date:
- Arrival Time:
- No of Adult...:
- Room Type:
- Contact Email:
- Contact Number:
- Created Date:
- Contact Description:

Below the form, there are five buttons arranged in three rows:

- Row 1: A single button labeled "RESET".
- Row 2: Two buttons labeled "ADD" and "UPDATE".
- Row 3: Two buttons labeled "GET" and "DELETE".

Insert the data required

Hotel Booking

First Name:  Last Name:

Arrival Date:  Arrival Time:

No of Adult...:  Room Type:

Contact Email:

Contact Number:

Created Date:

Contact Description:

RESET

ADD UPDATE

GET DELETE

Hotel Booking

First Name:  Last Name:

Arrival Date:  Arrival Time:

No of Adult...:  Room Type:

Contact Email:

Contact Number:

Created Date:

Contact Description:

RESET

ADD UPDATE

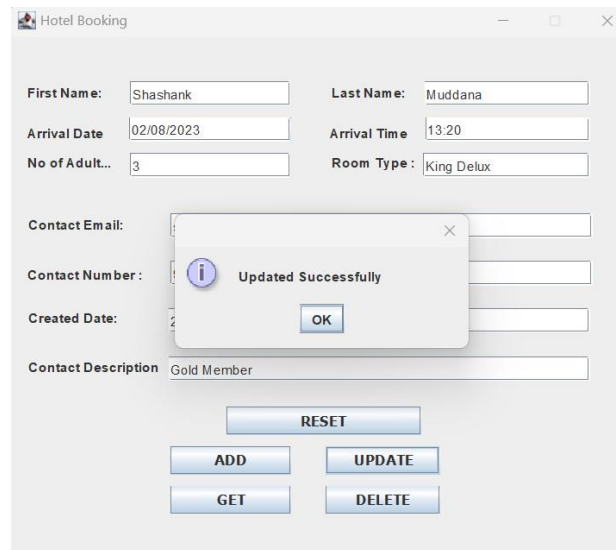
GET DELETE

Reservation Successful

OK

After clicking on the Add button, the Reservation Successful will be shown as Reservation Booked.

2. To update the Reservation, we need to change the fields and click the UPDATE button. In the below image, we can see the update success as we changed the date of reservation from 30/07/2023 to 02/08/2023.

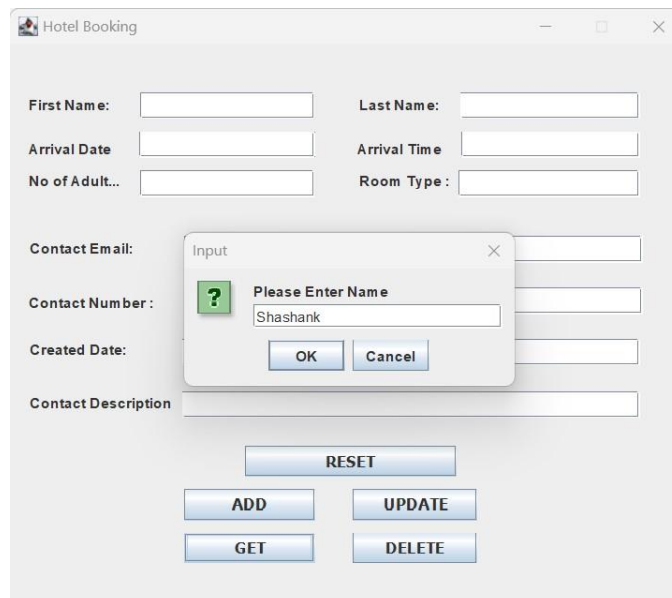


The screenshot shows the 'Hotel Booking' application window. The form contains the following fields and values:

Field	Value
First Name	Shashank
Last Name	Muddana
Arrival Date	02/08/2023
Arrival Time	13:20
No of Adult...	3
Room Type	King Deluxe
Contact Email	
Contact Number	
Created Date	
Contact Description	Gold Member

A modal dialog box is displayed in the center with the title 'Updated Successfully' and an 'OK' button. Below the form are four buttons: 'RESET', 'ADD', 'UPDATE', and 'DELETE'.

3. To view the Reservation, click on the GET button. This will ask us to enter the First Name. After providing the First Name we can see the appointment details.



The screenshot shows the 'Hotel Booking' application window. The form contains the following fields and values:

Field	Value
First Name	
Last Name	
Arrival Date	
Arrival Time	
No of Adult...	
Room Type	
Contact Email	
Contact Number	
Created Date	
Contact Description	

An 'Input' modal dialog box is displayed in the center with the title 'Please Enter Name'. It contains a text input field with the value 'Shashank' and 'OK' and 'Cancel' buttons. Below the form are four buttons: 'RESET', 'ADD', 'UPDATE', and 'DELETE'.

Hotel Booking

First Name:  Last Name:

Arrival Date:  Arrival Time:

No of Adult:  Room Type:

Contact Email:

Contact Number:

Created Date:

Contact Description:

RESET

ADD UPDATE

GET DELETE

- Click the DELETE button to remove the reservation.

Hotel Booking

First Name:  Last Name:

Arrival Date:  Arrival Time:

No of Adult:  Room Type:

Contact Email:

Contact Number:

Created Date:

Contact Description:

Records Deleted Succssfully

OK

RESET

ADD UPDATE

GET DELETE

- To clear the form, click on Reset Button. This will remove all the data in the form.

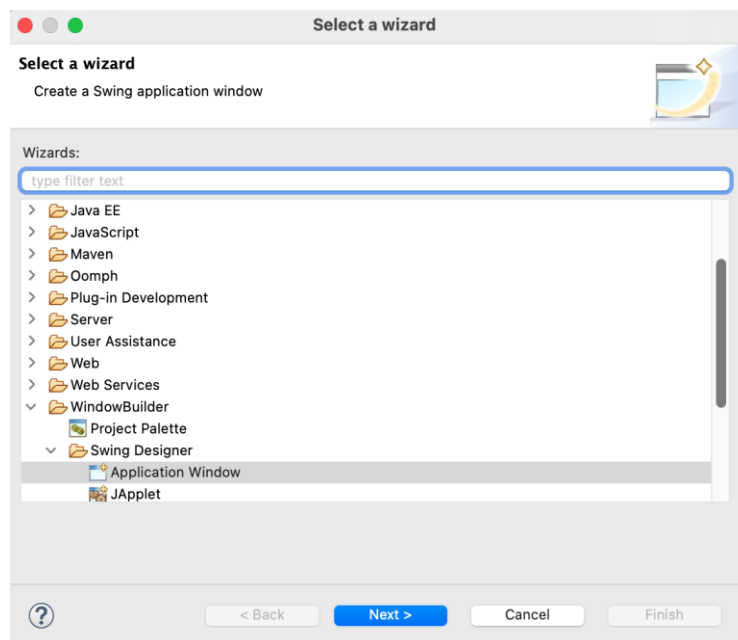
## CODE STRUCTURE AND FEATURES

The project contains three packages.

1. **hotelbooking.model** : In this package, we instantiate a model class Registration.java where we give the properties like for our project it will be

FirstName  
LastName  
ArrivalDate  
ArrivalTime  
NoOfAdults  
RoomType  
ContactEmail  
ContactNumber  
CreatedDate  
ContactDescription

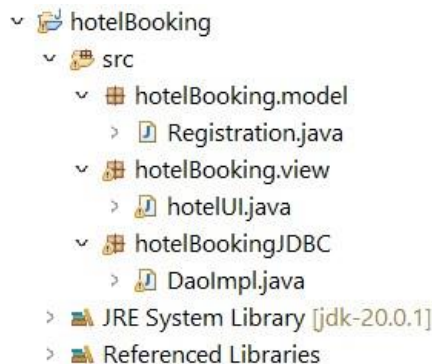
2. **hotelBooking.view:** In this package, we create a user interface hotelUI.java by right clicking on the package to create a new wizard. Select the window Builder. Click on the Application Window. Click on next and give the name and click Ok.





3. **hotelBookingJDBC:** In this package, we implement the database layer DaoImpl.java by adding database properties and do the CRUD operations.

We will use the MySQL driver in the project to connect to the database.



## Steps to develop the code

### Prerequisites :

To create Java Swing apps, install the WebBuilder program from the eclipse Marketplace.

Download mysql connector jar file from the internet.

1. Create a basic java project by right clicking on file -> new project -> Java project -> Create.
2. After creating the project, we must make packages by right-clicking on the src file and selecting "Create a New Package." Create classes in each of the packages.
3. To load the mysql jar file, right click on the project -> click on build path -> configure build path -> In the libraries tab, on the classpath right click -> add external jar file. Here we need to provide our downloaded jar file.
4. Once the project is set up we need to create our properties class and generate getters and setters.
5. For the User Interface, we can design our application directly by using drag and drop features provided by awt swing.

6. Once the design is completed we need to add functionalities to the design.

To read the values from the input fields we can use the `.getText()` method. The values from the UI need to be binded to the properties class which we already created ( getters and setters ).

To create an action event for the buttons, we have to use event listeners and create methods for each functionalities.

7. These static methods will be used in the database implementation. To connect to the database server, we need to add properties of the database server like hostname, port, username and password.
8. In each of the services, we need to create sql queries like insert, update, get and delete and assign the values to the properties class by using getters and setters methods.
9. The result coming from the services will be sent back to the UI and response will be shown in the application.

## **References**

<http://cswire.blogspot.com/2017/11/make-simple-crud-java-ui-application.html>