Visvesvaraya Technological University
Belagavi-590 018, Karnataka

A Mini Project Report on

"HOSTEL DATABASE MANAGEMENT "

Mini Project Report submitted in partial fulfilment of the requirement for the DBMS Laboratory with Mini Project [18CSL58]

Bachelor of Engineering
in
Computer Science and Engineering

Submitted by

Sarvajith [1JT19CS081]
Vishwas J S [1JT19CS105]

Department of Computer Science and Engineering
Jyothy Institute of Technology
Tataguni, Bengaluru-560082

Jyothy Institute of Technology
Tataguni, Bengaluru-560082
Department of Computer Science and Engineering

CERTIFICATE

Certified that the mini project work entitled **"HOSTEL DATABASE MANAGEMENT "** carried out by **Sarvajith [1JT19CS081] and Vishwas J S [1JT19CS105]** bonafide students of Jyothy Institute of Technology, in partial fulfilment for the award of **Bachelor of Engineering** in **Computer Science and Engineering** department of the **Visvesvaraya Technological University, Belagavi** during the year **2021-2022**. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the Report deposited in the departmental library. The project report has been approved as it satisfies the academic requirements in respect of Project work prescribed for the said Degree.

Mrs. Nikitha S
Guide, Asst. Professor
Dept. of CSE

Dr. Prabhanjan S
Professor & HOD
Dept. of CSE

External Viva Examiner

1.
2.

Signature with Date :

ACKNOWLEDGEMENT

Firstly, we are very grateful to this esteemed institution **“Jyothy Institute of Technology**”for providing us an opportunity to complete our project.

We express our sincere thanks to our **Principal Dr. Gopalakrishna K** for providing us withadequate facilities to undertake this project.

We would like to thank **Dr. Prabhanjan S, Professor and Head of Computer Science** andEngineering Department for providing for his valuable support.

We would like to thank our guides **Mrs.Nikitha S, Assistant Professor** for their keeninterest and guidance in preparing this work.

Finally, we would thank all our friends who have helped us directly or indirectly in thisproject.

Sarvajith [1JT19CS081]

Vishwas J S [1JT19CS105]

ABSTRACT

The application is designed to make the existing system more reliable, fast and easy for all, provides a methodical way of managing large databases. For this application we used the backend as SQL to store the data which is used in the application and for the user interface we have used JAVA.

We have designed a database system named, Hostel management database to maintain the details of Students of Hostel and manage the data effectively.

First activity while using the database is to add Student details along with details of Staffs details. This authority is given only to Admin.

Any modification to be done in the Hostel database can be done by Admin

TABLE OF CONTENTS

CHAPTER 1
INTRODUCTION

1. INTRODUCTION

1.1 Introduction to DBMS

A database is simply an organized collection of related data, typically stored on disk, and accessible by many concurrent users, it is a logically coherent collection of data with some inherent meaning, representing some aspect of real world and which is designed, built and populated with data for a specific purpose.

Databases are managed by a Database Management System(DBMS) which is a collection of programs that enables user to create and maintain a database.

Advantages of DBMS:

Redundancy is controlled.

Unauthorized access is restricted.

Providing multiple user interfaces.

Enforcing integrity constraints.

Providing backup and recovery.

1.2 Introduction to SQL

Structured Query Language (SQL), is a language used to request data from a database which includes database creation, deletion, retrieval of required tables and even manipulation of data held in a relational database management system.

SQL is considered as a Non-Procedural or a High level language in which the expected result or operation is given without the specific details about how to accomplish the task. So, SQL is a declarative language.

Therefore, SQL is designed at a higher conceptual level of operation than procedural languages as procedural languages includes only the information about opening and closing tables, loading and searching indexes, or flushing buffers and writing data to file systems, but the lower level logical and physical operations are not specified in SQL.

1.3 Introduction to Hostel Database Management

A Hostel is a place where many students who are from different places come to reside, During the study of courses , each hostel has respective staffs who are responsible for taking care of the hostel operations.

The “Hostel Database” is a database that is based on the details of the students and details of the rooms which they stay in , who manages the hostel ,etc. Hence Hostel Database Management software deals with all the above details related to hostel and its operations.

1.4 Scope and importance of work

The scope of the project is to give a simple application to overcome the drawbacks of the normal file processing system. That is to deal with software’s rather than having a hardcopy.

In our application, the database stores details about students , staffs, rooms , hostels and furniture’s which are available in the rooms.

Here we mainly deal in storing the details of all the students who have taken admission to the hostel and who can avail the services of the hostel

It also deals with details of staffs who are managing respective hostel and their operations

The main aim is to have proper record or details of the students who are present in the hostel Without any errors

We also get the details of the rooms available and hostels available with addition to all the furniture’s and facilities that are given to each room by the institution.

Hence Hostel Database Management software helps us in dealing with all the operations related to the Hostel.

CHAPTER 2
DESIGN

Theory of ER Diagram

The Entity–Relationship model (ER model) describes the structure of a database with the help of a diagram, which is known as **Entity Relationship Diagram (ER Diagram)**

An **Entity Relationship Diagram (ERD)** shows the relationships of entity sets stored in a database. An entity in this context is an object, a component of data.
An entity set is a collection of similar entities. These entities can have attributes that define its properties. By defining the entities, their attributes, and showing the relationships between them, an ER diagram illustrates the logical structure of database.

ER diagrams are used to sketch out the design of a database.

ENTITIES

An entity is an 'object' in the real world with an independent existence and an entity type defines a collection (or set) of entities that have the same attributes. Each entity type in the database is described by its name and attributes.

An entity type is represented in ER diagrams as a rectangular box enclosing the entity type name.

RELATIONSHIPS

A relationship among two or more entities represents an association among the entities and whenever an attribute of one entity refers to another entity, there exists a relationship between the two entities.

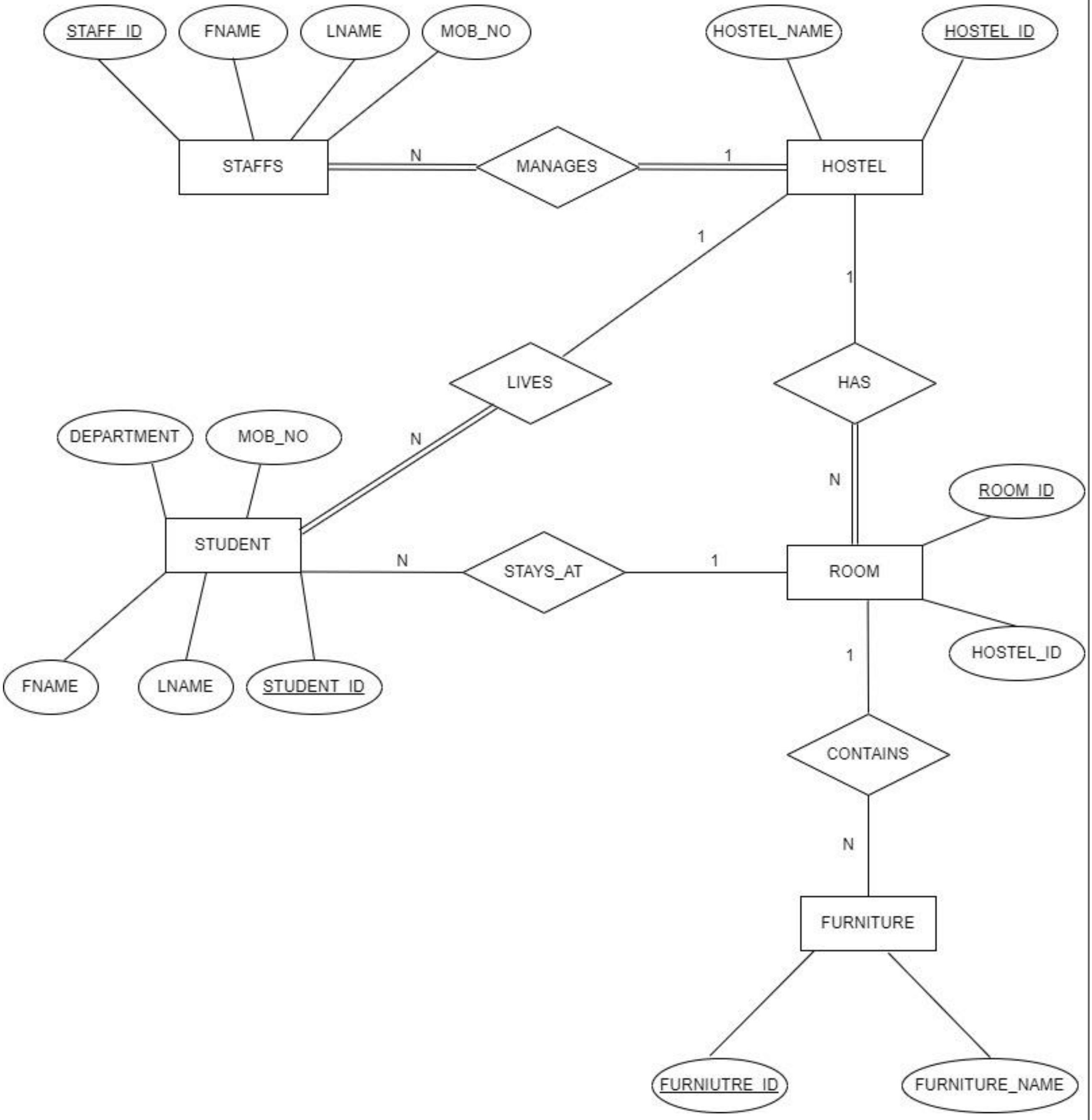
In a relationship, a foreign key of one table refers the primary key of the other table and it is represented by diamond shape in ER diagram.

ATTRIBUTES

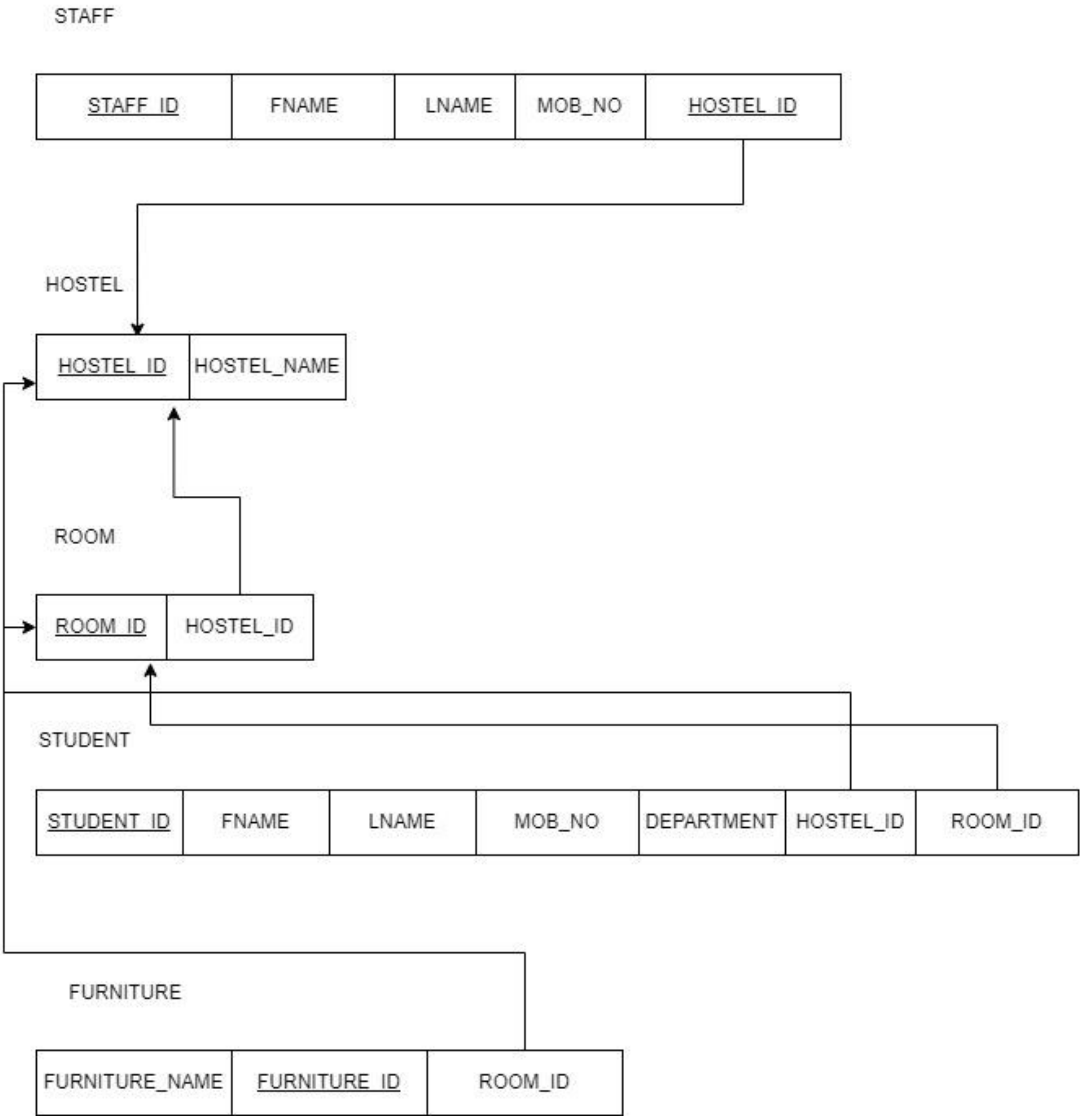
An attribute represents some property of interest that further describes an entity and the column header of the table shows the attributes. Each attribute in a table has a certain domain which allows it to accept a certain ‘set of values’ only.

The attribute values, of each entity, will define its characteristics in the table and is represented by oval in the ER diagram.

ER DIAGRAM


STAFF_ID
FNAME
LNAME
MOB_NO
HOSTEL_NAME
HOSTEL_ID
STAFFS
N
MANAGES
1
HOSTEL
1
1
LIVES
HAS
DEPARTMENT
MOB_NO
N
N
ROOM_ID
STUDENT
N
STAYS_AT
1
ROOM
1
HOSTEL_ID
FNAME
LNAME
STUDENT_ID
CONTAINS
N
FURNITURE
FURNIUTRE_ID
FURNITURE_NAME

SCHEMA DIAGRAM



List of Tables

1. **STAFFS**
2. **STUDENTS**
3. **HOSTEL**
4. **ROOM**
5. **FURNITURE**

CHAPTER 3
IMPLEMENTATION

Create table commands:-

1. Create table HOSTEL(Hostel_id integer , Hostel_name varchar(20),primary key(Hostel_id));

2. Create table ROOM (Room_id integer , Hostel_id integer , primary key(room_id) ,foreign key(hostel_id) references HOSTEL(Hostel_id) on delete cascade on update cascade);

3. Create table FURNITURE(Furniute_id integer, Furniture_name varchar(20),Room_id integer,primary key(Furniture_id),foreign key(Room_id) references HOSTEL(Hostel_id) on delete cascade on update cascade) ;

4. Create table STUDENT (Student_id integer, fname varchar(20),lname varchar(20) ,mob_no varchar(10),department varchar(20),Hostel_id integer,Room_id integer,primary key (Student_id),foreign key(Hostel_id) references HOSTEL(Hostel_id),foreign key (Room_id) references ROOM(Room_id) on delete cascade on update cascade);

5. Create table STAFF(Staff_id integer, fname varchar(20),lname varchar(20),mob_no varchar(10),Hostel_id integer,primary key(Staff_id),foreign key(Hostel_id) references HOSTEL(Hostel_id) on delete cascade on update cascade);

Insertion tables values

Insertion of STAFF table

Insert into STAFF values(1,’GANESH’,’KUMAR’, 7259708456, 1);

Insert into STAFF values(2,’MUKESH’,’SHINDE’, 9945998391, 2);

Insert into STAFF values(3,’LATHA’,’KUMARI’, 7878789999, 3);

Insertion of STUDENT table

Insert into STUDENT values(1,’ROHITH’,’SHARMA’, 9845551036, ‘CS’,1 , 101);

Insert into STUDENT values(2,’RAMESH’,’RAO’, 9844441035 , ‘IS’, 1 , 102);

Insert into STUDENT values(3,’VISHWAS’,’JS’, 9988776655, ‘CS’, 2 ,
201);

Insertion of HOSTEL table

Insert into HOSTEL values(1,’bh1’);

Insert into HOSTEL values(2,’bh2’);

Insert into HOSTEL values(3,’bh3’);

Insert into HOSTEL values(4,’bh4’);

Insertion of ROOM table

Insert into ROOM values(101,1);

Insert into ROOM values(102,1);

Insert into ROOM values(103,1);

Insert into ROOM values(201,2);

Insert into ROOM values(202, 2);

Insertion of FURNITURE table

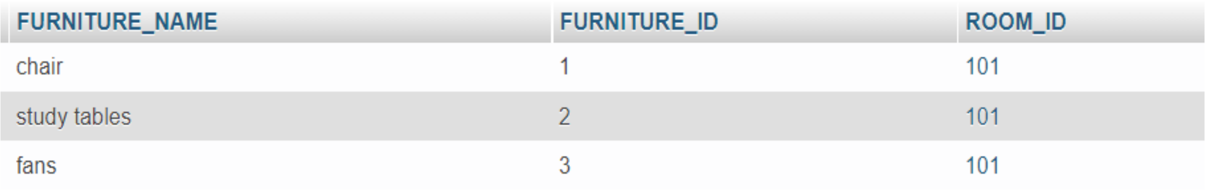
Insert into FURNITURE values(‘chair’, 1 , 101);

Insert into FURNITURE values(‘study tables’, 2 , 101);

Insert into FURNITURE values(‘fans’, 3 , 101);

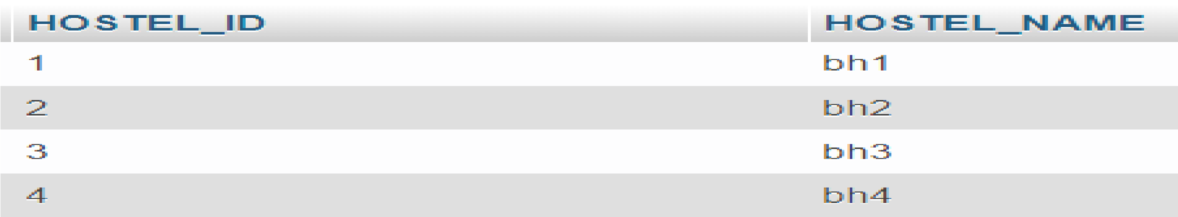
SnapShots

FURNITURE



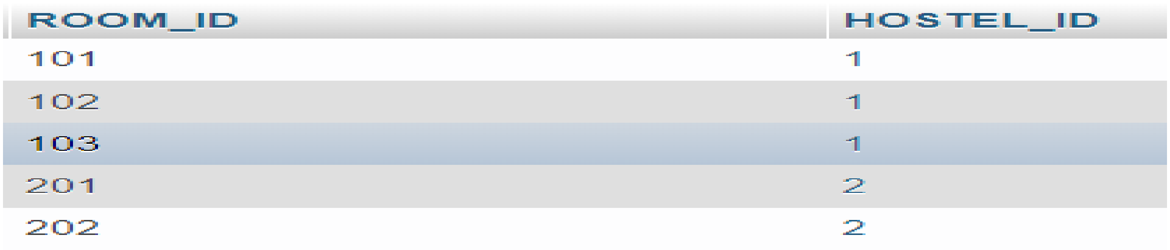
FURNITURE_NAME	FURNITURE_ID	ROOM_ID
chair	1	101
study tables	2	101
fans	3	101

HOSTEL



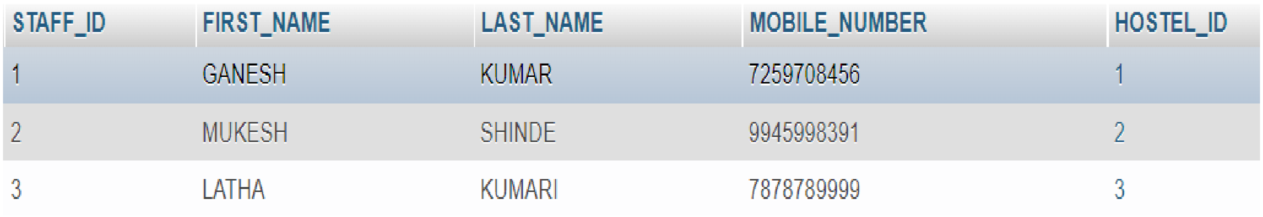
HOSTEL_ID	HOSTEL_NAME
1	bh1
2	bh2
3	bh3
4	bh4

ROOM



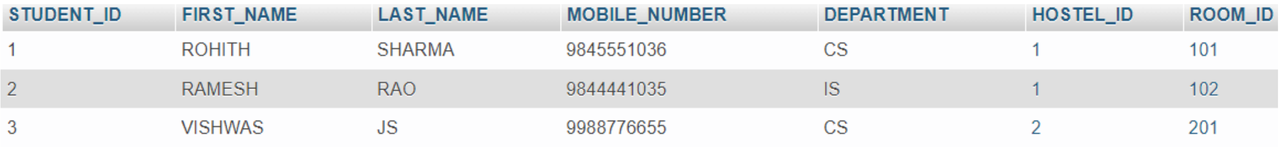
ROOM_ID	HOSTEL_ID
101	1
102	1
103	1
201	2
202	2

STAFF



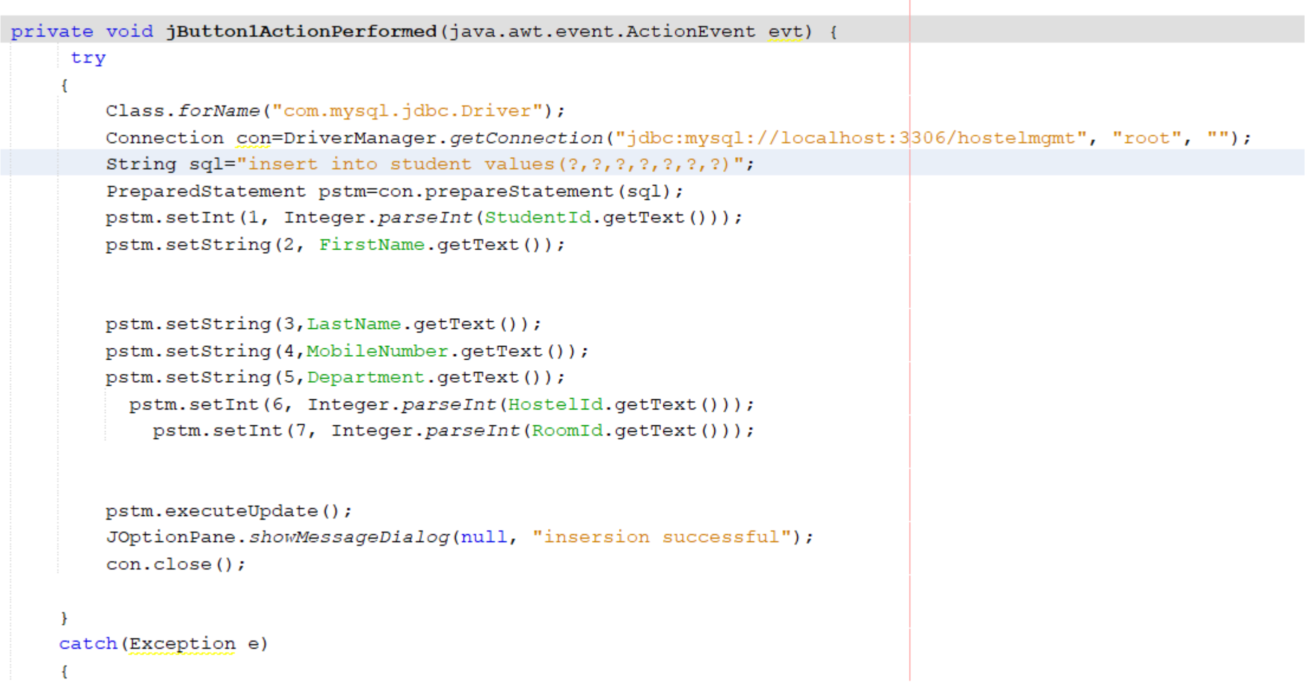
STAFF_ID	FIRST_NAME	LAST_NAME	MOBILE_NUMBER	HOSTEL_ID
1	GANESH	KUMAR	7259708456	1
2	MUKESH	SHINDE	9945998391	2
3	LATHA	KUMARI	7878789999	3

STUDENT

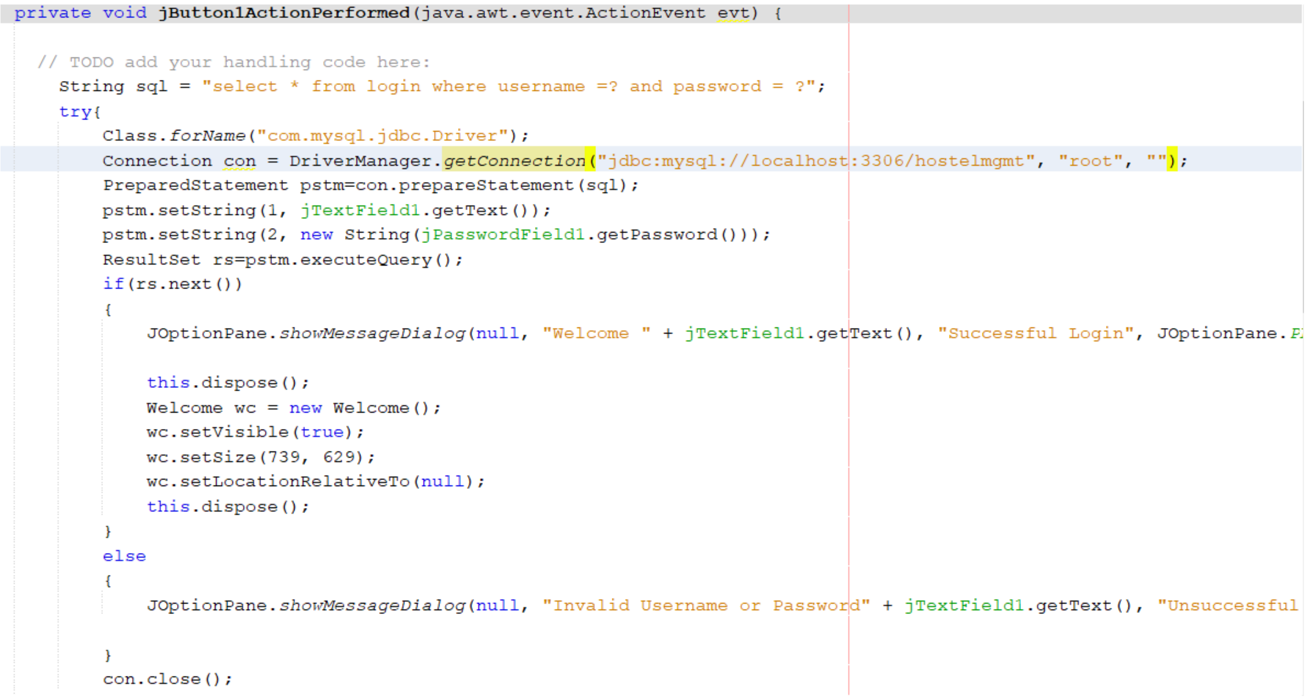


STUDENT_ID	FIRST_NAME	LAST_NAME	MOBILE_NUMBER	DEPARTMENT	HOSTEL_ID	ROOM_ID
1	ROHITH	SHARMA	9845551036	CS	1	101
2	RAMESH	RAO	9844441035	IS	1	102
3	VISHWAS	JS	9988776655	CS	2	201

GUI implementation



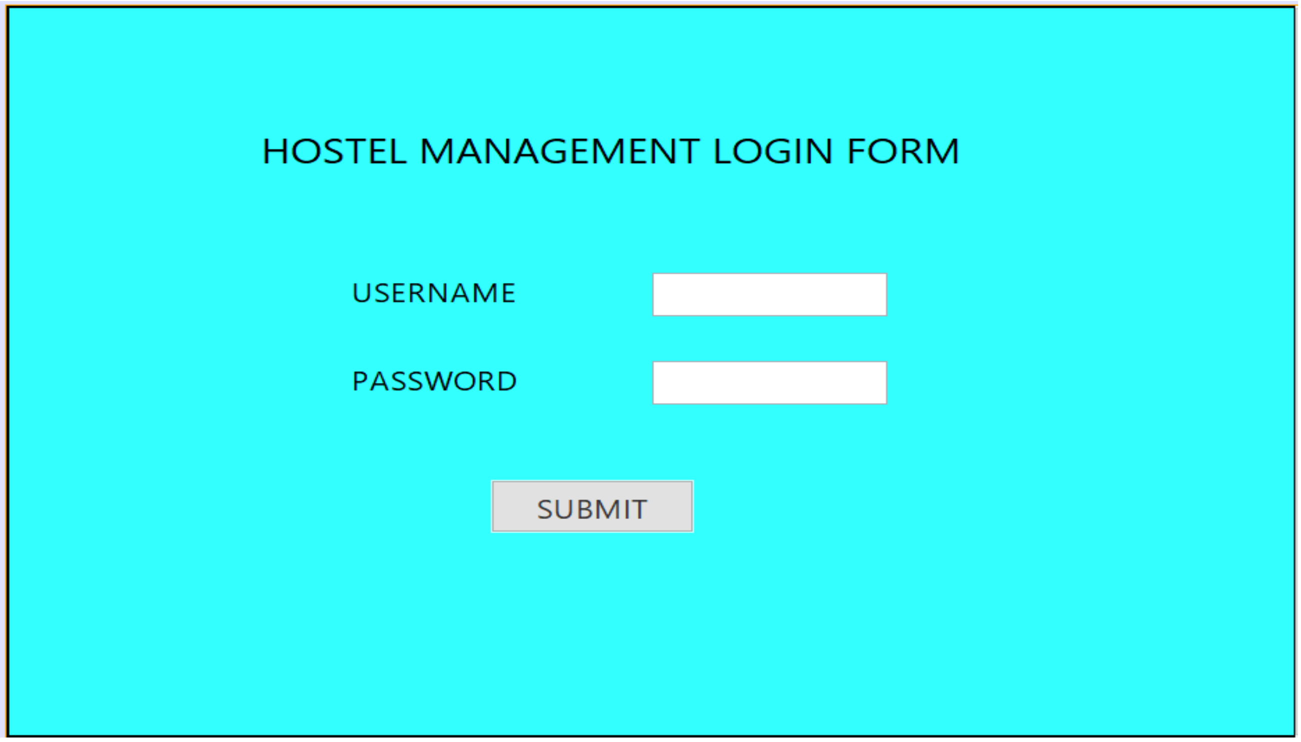
```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
     try
    {
        Class.forName("com.mysql.jdbc.Driver");
        Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/hostelmgmt", "root", "");
        String sql="insert into student values(?,?,?,?,?,?,?)";
        PreparedStatement pstm=con.prepareStatement(sql);
        pstm.setInt(1, Integer.parseInt(StudentId.getText()));
        pstm.setString(2, FirstName.getText());


        pstm.setString(3,LastName.getText());
        pstm.setString(4,MobileNumber.getText());
        pstm.setString(5,Department.getText());
          pstm.setInt(6, Integer.parseInt(HostelId.getText()));
            pstm.setInt(7, Integer.parseInt(RoomId.getText()));


        pstm.executeUpdate();
        JOptionPane.showMessageDialog(null, "insersion successful");
        con.close();

    }
    catch(Exception e)
    {
```



```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {

    // TODO add your handling code here:
    String sql = "select * from login where username =? and password = ?";
    try{
        Class.forName("com.mysql.jdbc.Driver");
        Connection con = DriverManager.getConnection("jdbc:mysql://localhost:3306/hostelmgmt", "root", "");
        PreparedStatement pstm=con.prepareStatement(sql);
        pstm.setString(1, jTextField1.getText());
        pstm.setString(2, new String(jPasswordField1.getPassword()));
        ResultSet rs=pstm.executeQuery();
        if(rs.next())
        {
            JOptionPane.showMessageDialog(null, "Welcome " + jTextField1.getText(), "Successful Login", JOptionPane.P

            this.dispose();
            Welcome wc = new Welcome();
            wc.setVisible(true);
            wc.setSize(739, 629);
            wc.setLocationRelativeTo(null);
            this.dispose();
        }
        else
        {
            JOptionPane.showMessageDialog(null, "Invalid Username or Password" + jTextField1.getText(), "Unsuccessful

        }
        con.close();
```

CHAPTER 4
RESULTS
AND
SNAPSHOTS

LOGIN PAGE

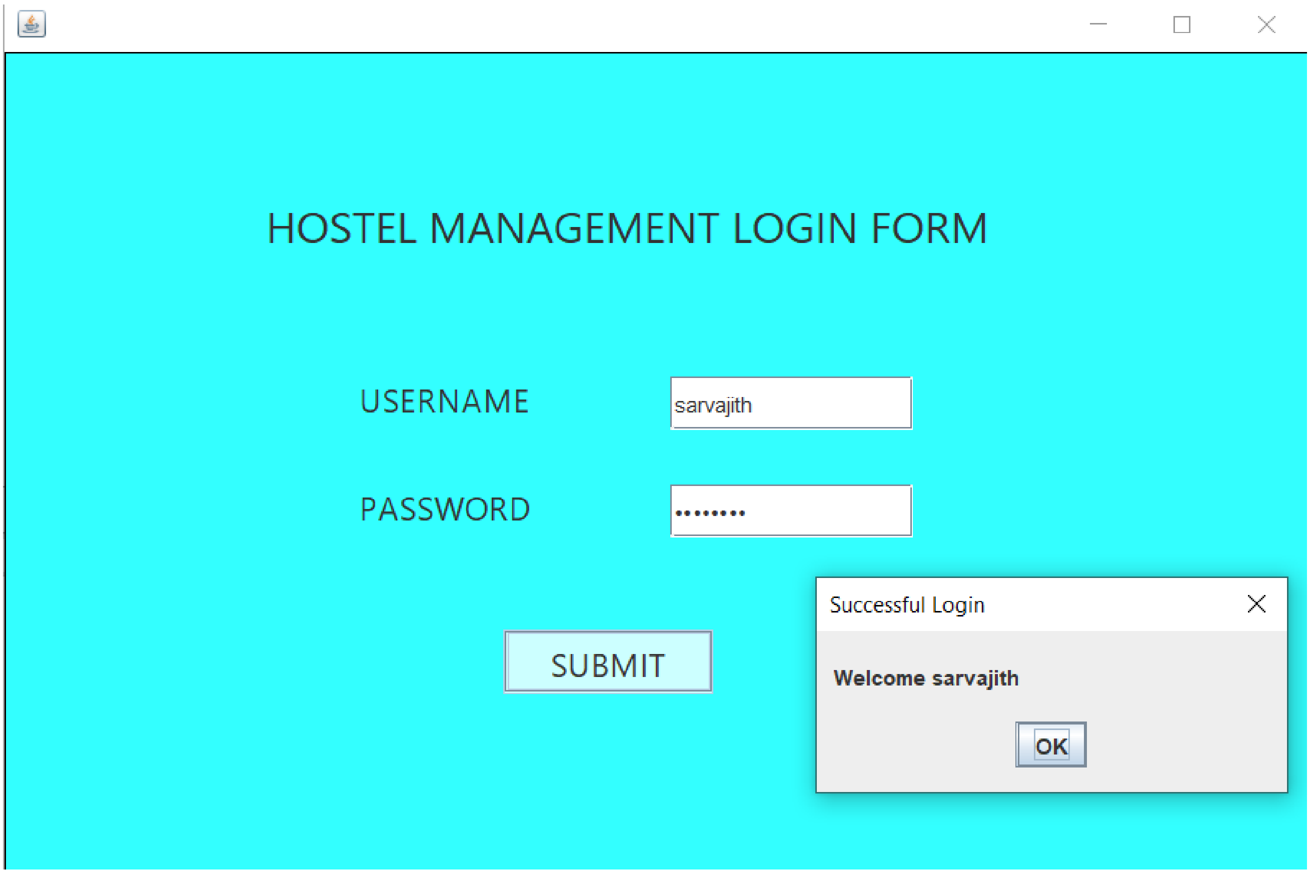


HOSTEL MANAGEMENT LOGIN FORM

USERNAME

PASSWORD

SUCCESSFUL LOGIN MESSAGE



HOSTEL MANAGEMENT LOGIN FORM

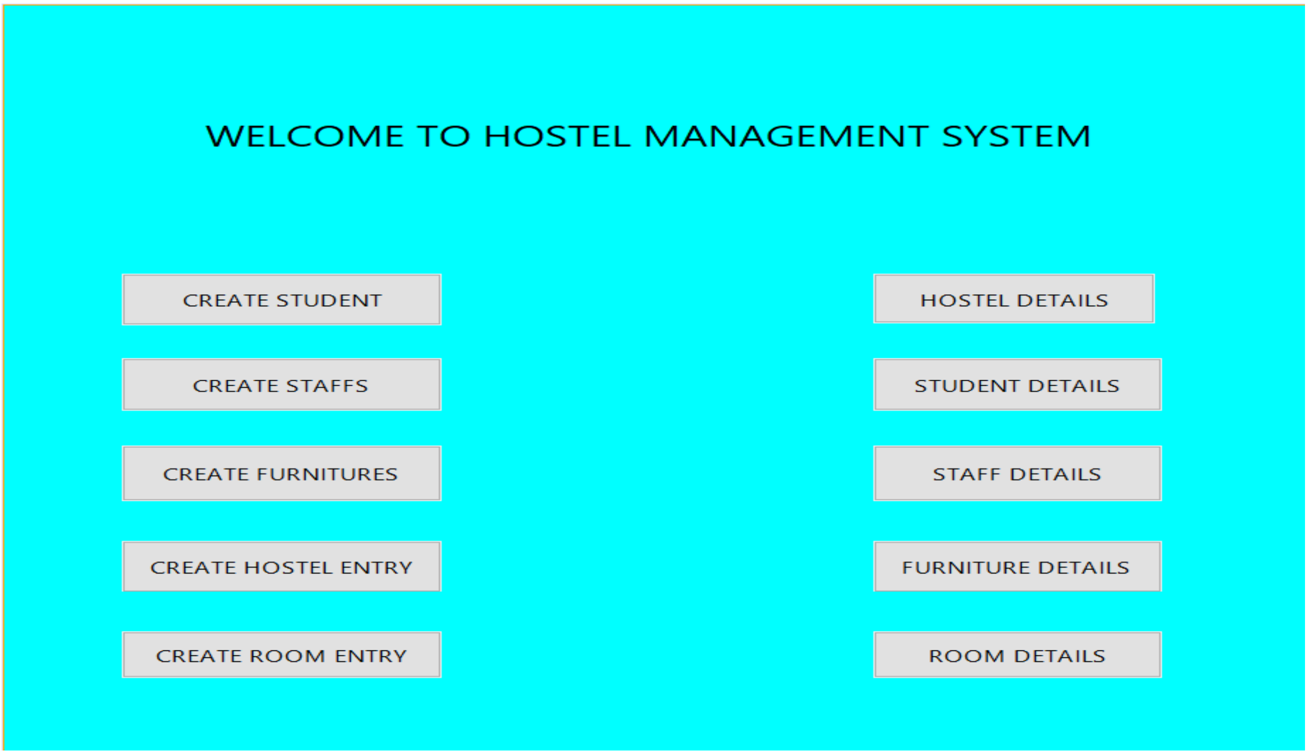
USERNAME sarvajith

PASSWORD ••••••••

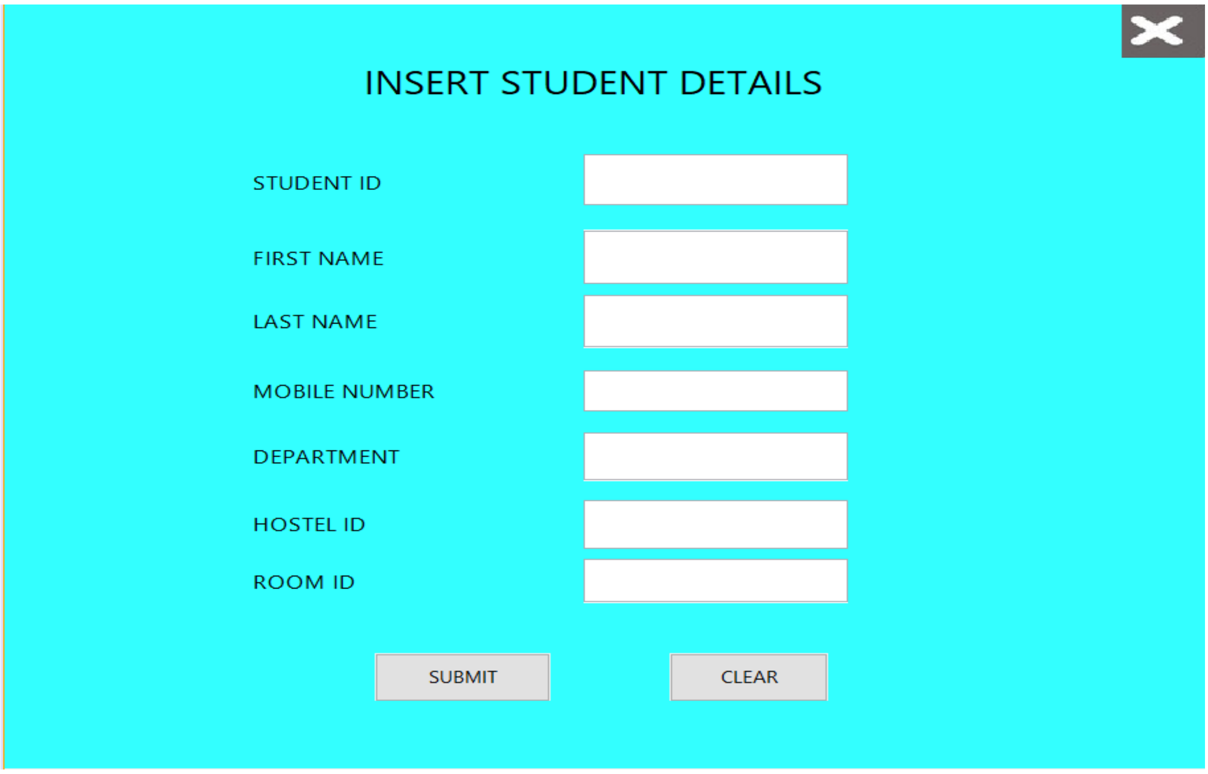
Successful Login

Welcome sarvajith

MAIN PAGE


WELCOME TO HOSTEL MANAGEMENT SYSTEM

INSERT STUDENT PAGE


INSERT STUDENT DETAILS

STUDENT ID

FIRST NAME

LAST NAME

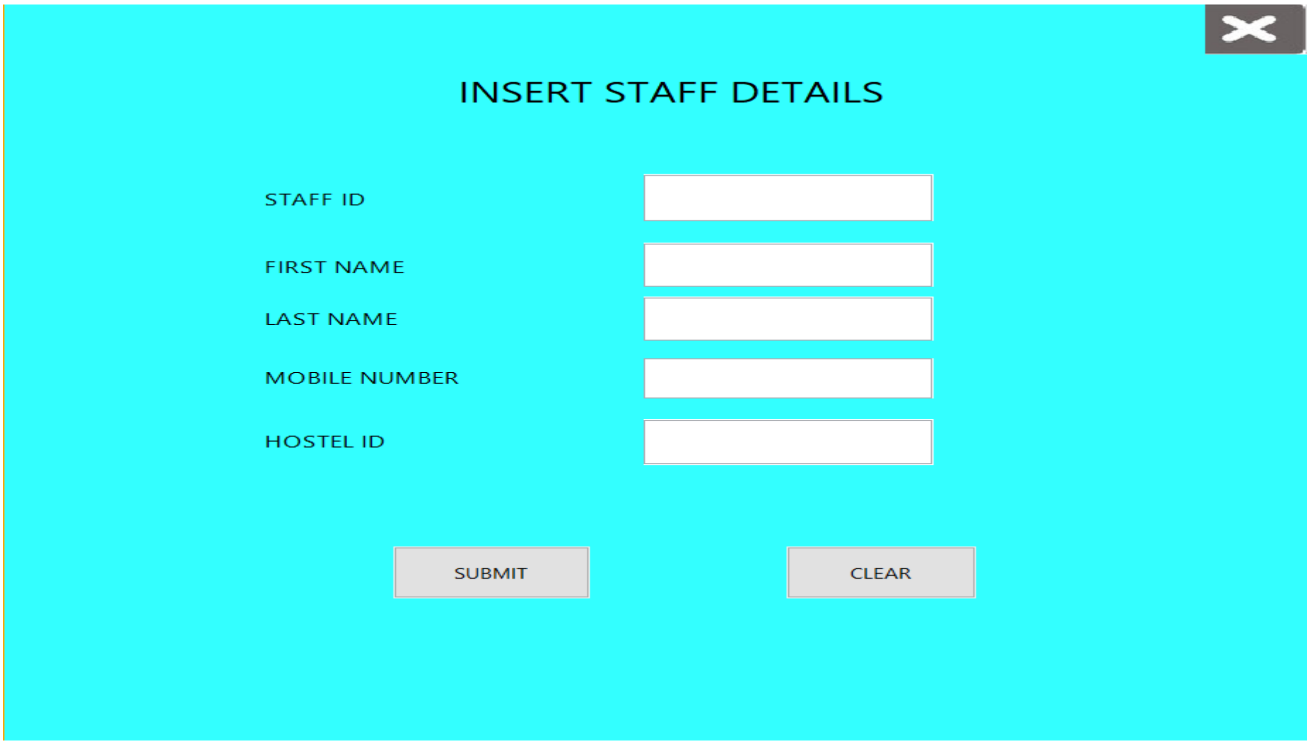
MOBILE NUMBER

DEPARTMENT

HOSTEL ID

ROOM ID

INSERT STAFFS PAGE



INSERT STAFF DETAILS

STAFF ID

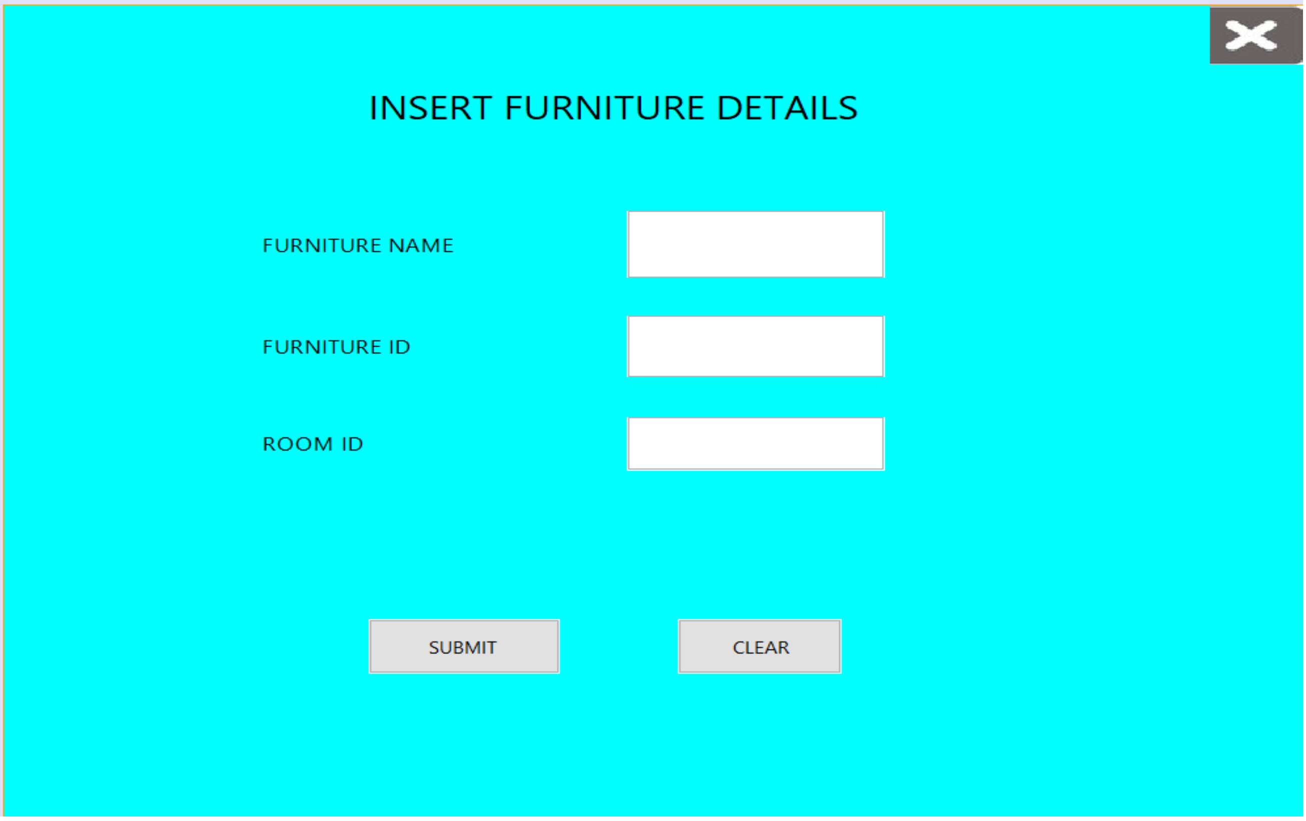
FIRST NAME

LAST NAME

MOBILE NUMBER

HOSTEL ID

INSERT FURNITURE DETAILS



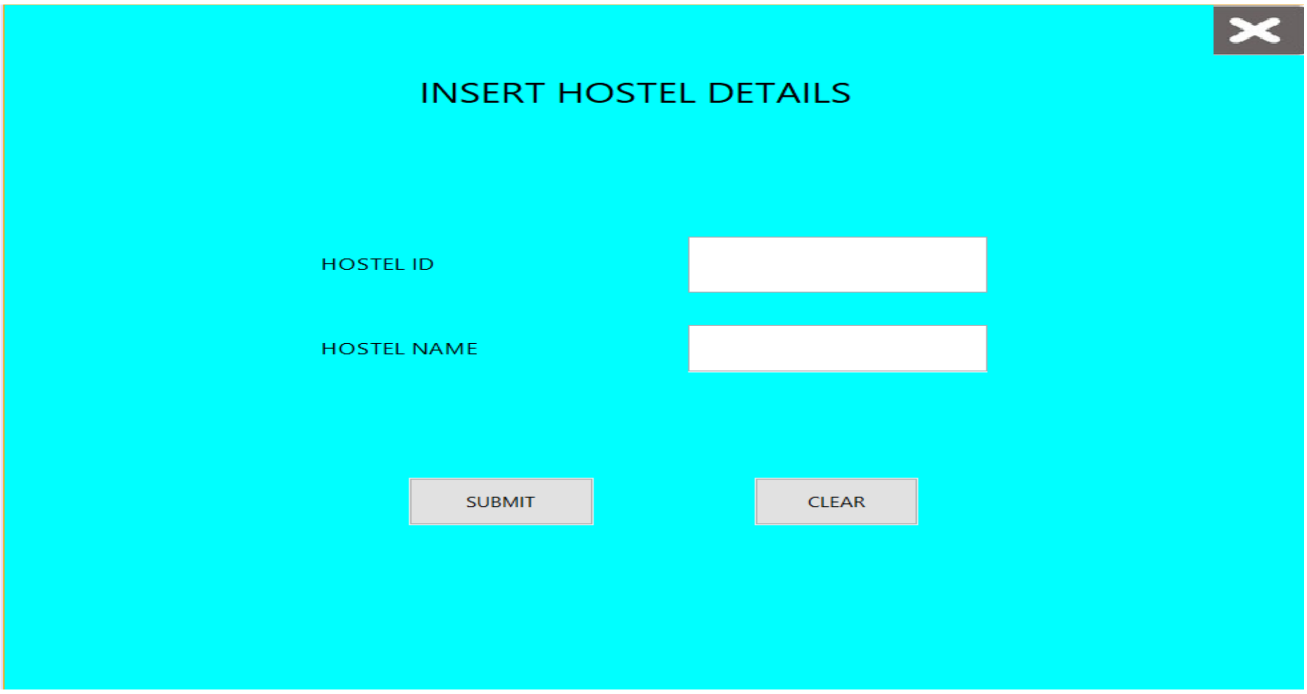
INSERT FURNITURE DETAILS

FURNITURE NAME

FURNITURE ID

ROOM ID

INSERT HOSTEL DETAILS

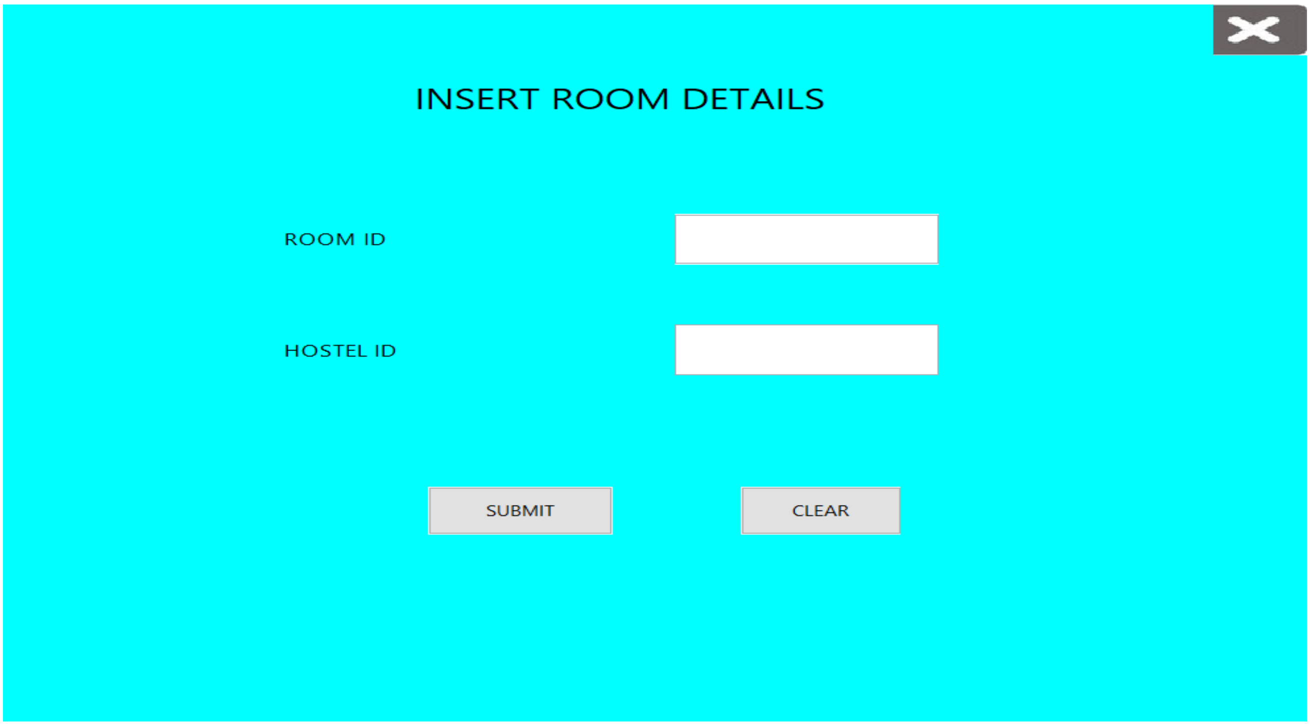


INSERT HOSTEL DETAILS

HOSTEL ID

HOSTEL NAME

INSERT ROOM DETAILS

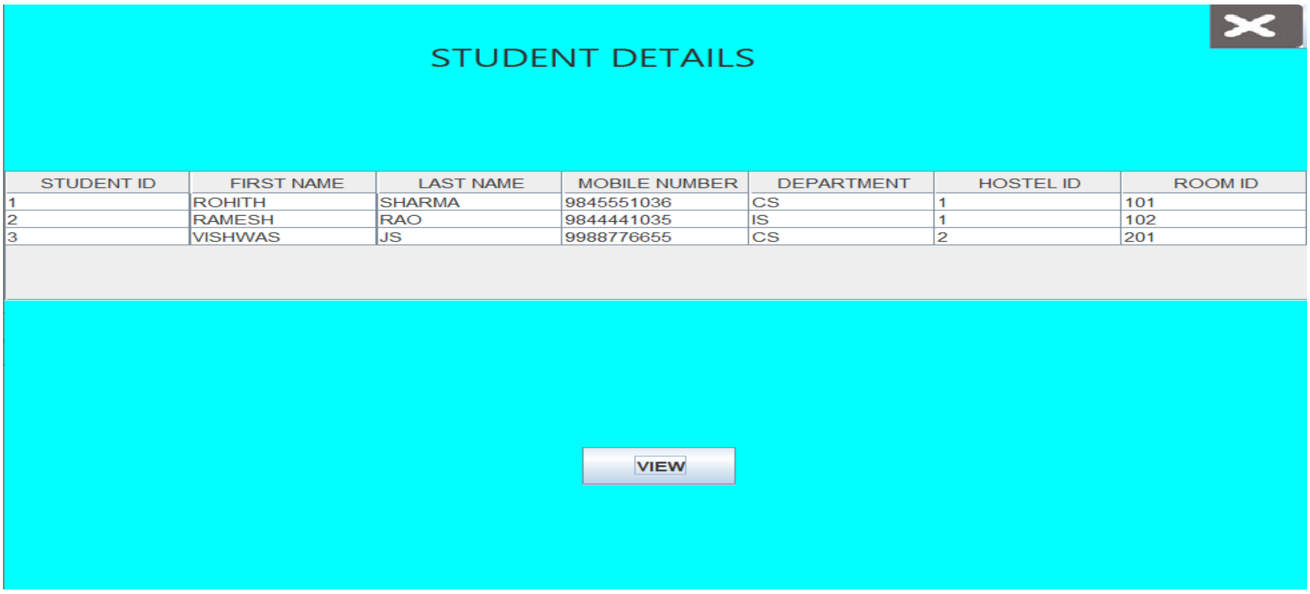


INSERT ROOM DETAILS

ROOM ID

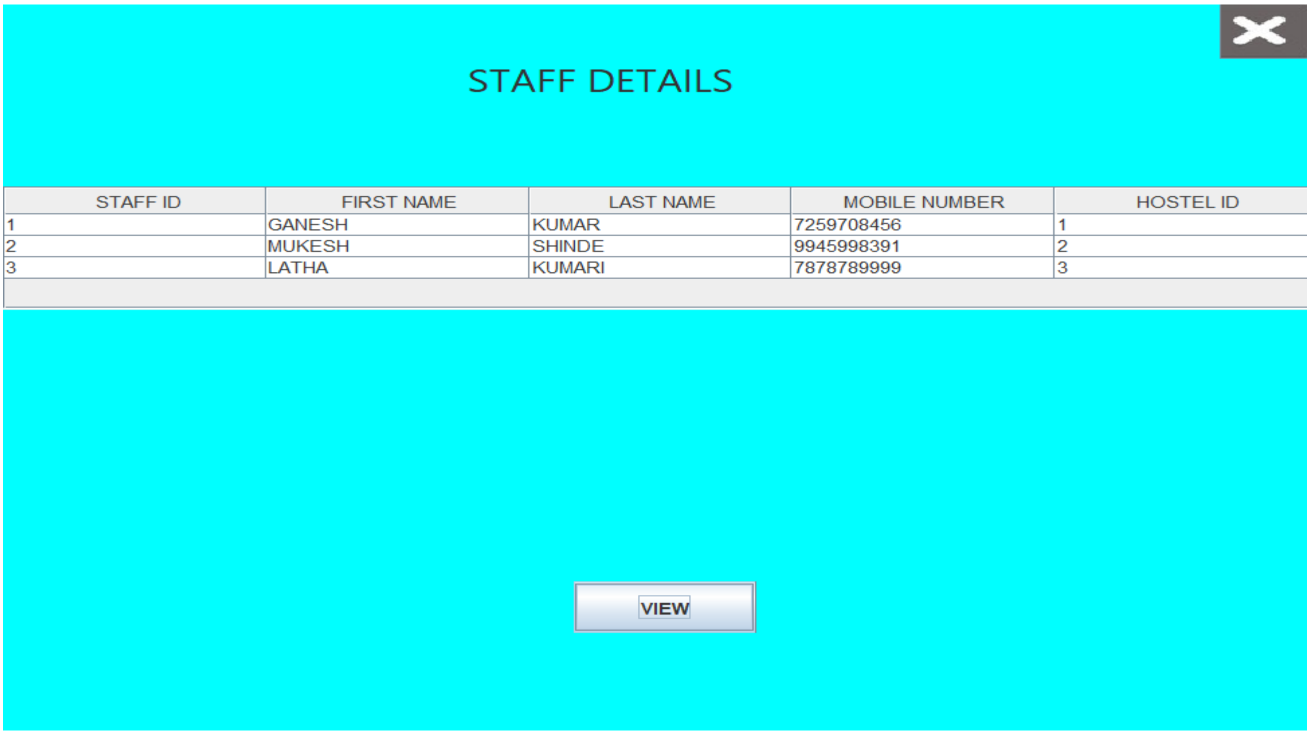
HOSTEL ID

STUDENT DETAILS



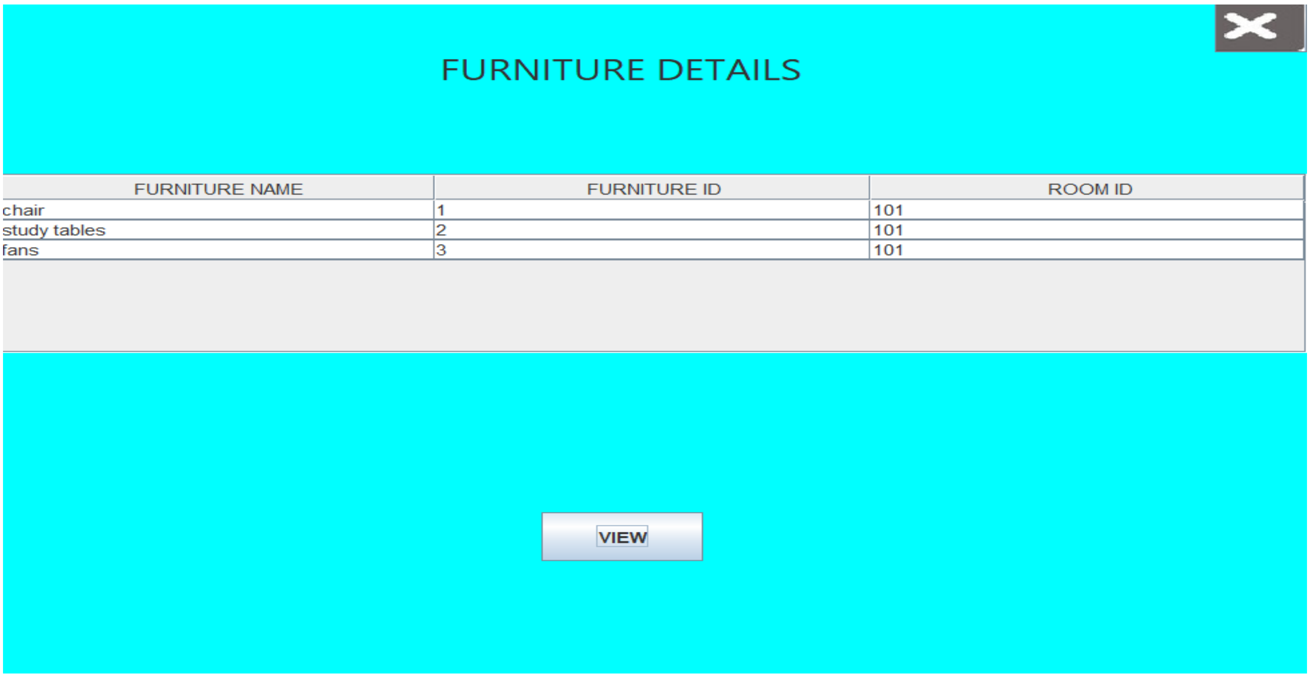
STUDENT ID	FIRST NAME	LAST NAME	MOBILE NUMBER	DEPARTMENT	HOSTEL ID	ROOM ID
1	ROHITH	SHARMA	9845551036	CS	1	101
2	RAMESH	RAO	9844441035	IS	1	102
3	VISHWAS	JS	9988776655	CS	2	201

STAFF DETAILS



STAFF ID	FIRST NAME	LAST NAME	MOBILE NUMBER	HOSTEL ID
1	GANESH	KUMAR	7259708456	1
2	MUKESH	SHINDE	9945998391	2
3	LATHA	KUMARI	7878789999	3

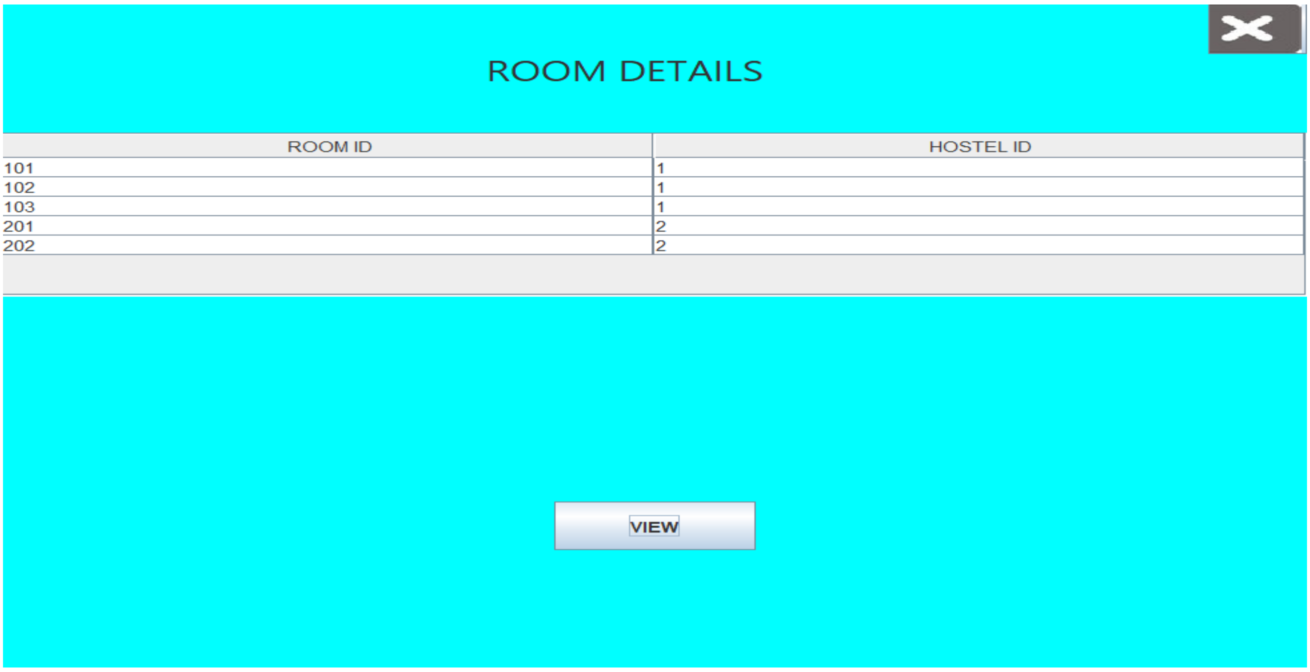
FURNITURE DETAILS



FURNITURE DETAILS

FURNITURE NAME	FURNITURE ID	ROOM ID
chair	1	101
study tables	2	101
fans	3	101

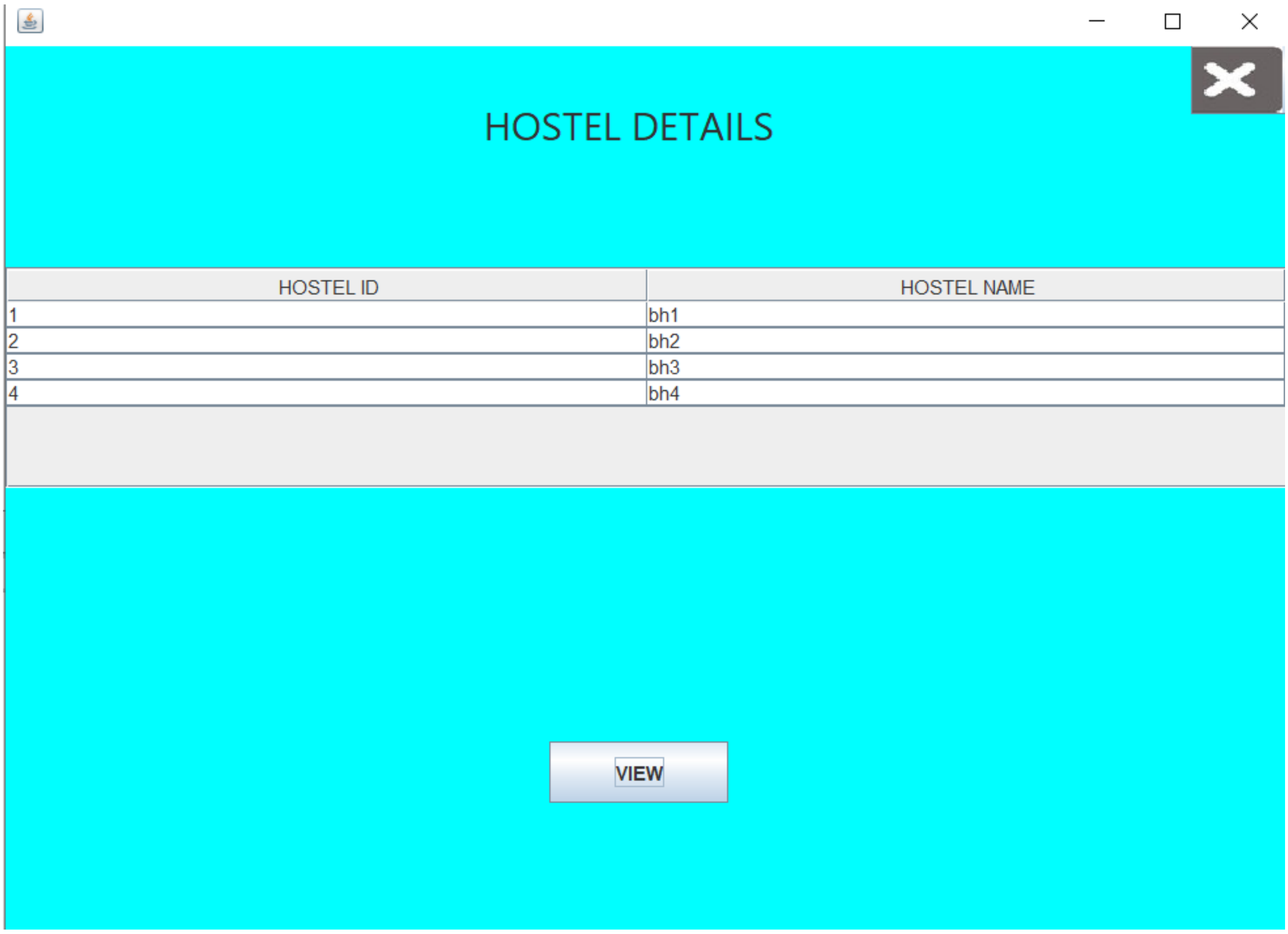
ROOM DETAILS



ROOM DETAILS

ROOM ID	HOSTEL ID
101	1
102	1
103	1
201	2
202	2

HOSTEL DETAILS PAGE



HOSTEL DETAILS

HOSTEL ID	HOSTEL NAME
1	bh1
2	bh2
3	bh3
4	bh4

QUERIES

1. SELECT * FROM STUDENT
2. SELECT * FROM STAFF
3. SELECT *FROM ROOM
4. SELECT * FROM HOSTEL
5. SELECT * FRON FURNITURE

Conclusion

We have successfully implemented the HOSTEL DATABASE MANAGEMENT which helps in managing the data used to perform the various tasks in the HOSTEL.

View tables are used to display all the components of different entities that user needs. One can just select the buttons and modify the data as per requirements.

We have successfully used various functionalities of JAVA and SQL and created the fully functional database system

HOSTEL Database has to do with making appropriate effort to stop the rising problem of data ambiguity and proper management of hostel students details and the hostel operations.

In this project, the software or system that can be used to aid all hostels and make it simpler to manage all the student and staff involved in the proper working of a HOSTEL.
The software can be implemented in any Hostels.

Features

1. A password system that will be embedded into login page to increase the Security of the system.
2. A good Printing module should be included.
3. A data required for different operations are accessible to the admin.
4. Quick and easy saving and loading of database file.

References

Net Beans 8.2

https://docs.oracle.com/netbeans/nb82/netbeans/docs.htm

JDBC Driver for MySQL (Connector/J)

https://dev.mysql.com/downloads/connector/j/5.1.html

MySQL Database

https://www.mysql.com/downloads/