

HMS

#main method

#admin login.java

```
package com.test2;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.Scanner;
import java.util.regex.Matcher;
import java.util.regex.Pattern;
import java.util.Scanner;

public class AdminLogin {

    public static final String ANSI_RED="\u001B[31m";
    public static final String ANSI_BLACK="\u001B[30m";

    static Scanner s=new Scanner(System.in);
    static String id;
    private static Connection conn;

    private static final String USERNAME = "aparna";
    private static final String PASSWORD = "admin@123";
    private static final int MAX_ATTEMPTS = 3;

    public static void main(String[] args) throws SQLException {
        Scanner scanner = new Scanner(System.in);
        int attempts = 0;
```

```

while (attempts < MAX_ATTEMPTS) {

    System.out.print("Enter admin username: ");

    String username = scanner.nextLine();

    System.out.print("Enter admin password: ");

    String password = scanner.nextLine();

    if (username.equals(USERNAME) && password.equals(PASSWORD)) {

        System.out.println("Login successful!");

        adminMenu();

        break;

    } else {

        attempts++;

        System.out.println("Invalid credentials. Attempts remaining: " + (MAX_ATTEMPTS -
attempts));

    }

}

if (attempts == MAX_ATTEMPTS) {

    System.out.println("Maximum login attempts exceeded!");

}

}

static Patients patient= new Patients();

static Doctors doctor=new Doctors();

static Appointments appointment=new Appointments();

static MedicalHistories medicalrecord=new MedicalHistories();

static Rooms room=new Rooms();

private static void adminMenu() throws SQLException {

```

```

Scanner scanner = new Scanner(System.in);

while(true)

{
    System.out.println("****RVS HOSPITAL MANAGEMNT SYSTEM*****");
    System.out.println("Admin Menu:");
    System.out.println("1. Patient");
    System.out.println("2. Doctor");
    System.out.println("3. Appointment");
    System.out.println("4.Medical Records");
    System.out.println("5.Room");
    System.out.println("6. Exit");
    System.out.println("*****");

    System.out.println("Enter Your Choice within 1 to 6: ");
    int outerchoice = scanner.nextInt();
    conn=DriverManager.getConnection("jdbc:mysql://localhost:3306/hms","root","aparna");
    boolean flag=true;
    while(flag)
    {
        switch (outerchoice) {
        case 1:{
            System.out.println("*****");
            System.out.println("Patient Details");
            System.out.println("1.Add Patient");
            System.out.println("2.View Patient");
            System.out.println("3.Get Patient Details by Id");
            System.out.println("4.update patient Details");
            System.out.println("5. Exit");
            System.out.println("*****");

```

```
        System.out.println("Enter Your Choice with in 1 to 5 : ");
int innerchoice=scanner.nextInt();
boolean f=true;
while(f) {
    switch(innerchoice)
    {
        case 1:
            patient.addPatient();
            break;
        case 2:
            patient.viewPatient();
            break;
        case 3:
            System.out.println("enter the id to search");
            id=s.nextLine();
            patient.getPatientById(id);
            break;
        case 4:
            patient.updatePatient();
            break;
        case 5:
            System.out.println("exit from Patient....Thank you");
            f=false;
            flag=false;

    }
    break;
}
```

```
break;  
}
```

case 2:

```
{  
    boolean flag1=true;  
    while(flag1)  
    {  
        System.out.println("*****");  
        System.out.println("Doctor Details");  
        System.out.println("1.Add Doctor");  
        System.out.println("2.View Doctor");  
        System.out.println("3.Get Doctor Details by Id");  
        System.out.println("4.update Doctor Details");  
        System.out.println("5. Exit");  
        System.out.println("*****");  
        System.out.println("Enter Your Choice with in 1 to 5 : ");  
        int innerchoice=scanner.nextInt();
```

```
        switch(innerchoice)
```

```
{
```

case 1:

```
        doctor.addDoctor();  
        break;
```

case 2:

```
        doctor.viewDoctor();  
        break;
```

case 3:

```
        System.out.println("enter the id to search");  
        id=s.nextLine();  
        doctor.getDoctorById(id);
```

```
        break;
    case 4:
        doctor.updateDoctor();
        break;
    case 5:
        System.out.println("Exiting from Doctor Module...Thank you!!");
        flag1=false;
        flag=false;

    }
    break;

}
break;

}
```

```
case 3:
{
    boolean flag2=true;
    while(flag2)
    {
        System.out.println("*****");
        System.out.println("Appointment Details");
        System.out.println("1.Book Appointment");
        System.out.println("2.Check Appointment Availability");
        System.out.println("3. Exit");
        System.out.println("*****");
    }
}
```

```
        System.out.println("Enter Your Choice within 1 to 3 : ");
int innerchoice=scanner.nextInt();

switch(innerchoice)
{
case 1:
    appointment.BookAppointment();
    break;
case 2:
    System.out.println("Enter Doctor Id to check for appointment");
    id=s.nextLine();
    System.out.println("Enter appointment date ");
    String app_date=s.nextLine();
    appointment.checkavailability(id,app_date);
    break;
case 3:
    System.out.println("Exiting from Apointment...Thank you!!!");
    flag2=false;
    flag=false;

}
break;
}
break;
}
case 4:
{
    boolean flag4=true;
    while(flag4)
    {
        System.out.println("*****");
```

```

        System.out.println("Medical Record Details");
        System.out.println("1.Add Medical History");
        System.out.println("2.View medical History");
        System.out.println("3.Get Medical Record Details by Id");
        System.out.println("4. Exit");
        System.out.println("*****");

        System.out.println("Enter Your Choice within 1 to 4: ");
        int innerchoice=scanner.nextInt();
        switch(innerchoice)
        {
        case 1:
            medicalrecord.addMedicalHistory();
            break;
        case 2:
            medicalrecord.viewMedicalHistory();
            break;
        case 3:
            System.out.println("enter the id to search");
            id=s.nextLine();
            medicalrecord.getMedicalHistoryById(id);
            break;
        case 4:
            System.out.println("Exiting from Medical Record.... Thnak you!!");
            flag4=false;
            flag=false;
        }
        break;
    }
    break;
}

```



```
}

        case 5:

    {
        boolean flag5=true;
        while(flag5)
        {
            System.out.println("*****");
            System.out.println("Room Details");
            System.out.println("1.Add Room Details");
            System.out.println("2.View Room Details");
            System.out.println("3.Get Room Details by Id");
            System.out.println("4.update patient Details");
            System.out.println("5. Exit");
            System.out.println("*****");

            System.out.println("Enter Your Choice with in 1 to 5 : ");
            int innerchoice=scanner.nextInt();

            switch(innerchoice)
            {
                case 1:
                    room.addRoom();
                    break;

                case 2:
                    room.viewRoom();
                    break;

                case 3:
                    System.out.println("Enter id to get room details:");
                    id=s.nextLine();
                    room.getRoomById(id);
```

```

        break;

    case 4:

        System.out.println("Exit from Room Details... Thank you!!");

        flag5=false;

        flag=false;

    }

    break;

}

break;

}

case 6:

    System.out.println("Exit From Menu");

    System.exit(0);

    break;

}

}

}

}

}

```

#Appointments.java

```

package com.test2;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.Scanner;

```

```
public class Appointments {

    static Connection conn;

    Patients patient;

    Doctors doctor;

    Scanner s=new Scanner(System.in);

    public Appointments() {
        this.conn = conn;
    }

    public void BookAppointment() throws SQLException {

        System.out.println("Enter appointment Id: ");
        String app_id=s.nextLine();

        System.out.println("Enter patient Id: ");
        String P_id=s.nextLine();

        System.out.println("Enter Doctor Id: ");
        String d_id=s.nextLine();

        System.out.println("Enter appointment date(yy--mm--dd): ");
        String app_date=s.nextLine();

        System.out.println("Enter appointment Time: ");
        String time=s.nextLine();

        Connection
        conn=DriverManager.getConnection("jdbc:mysql://localhost:3306/hms","root","aparna");

        String query="insert into appointment values(?,?,?,?)";
```

```

try(PreparedStatement ps=conn.prepareStatement(query))
{
    ps.setString(1, app_id);
    ps.setString(2, P_id);
    ps.setString(3, d_id);
    ps.setString(4,app_date);
    ps.setString(5, time);

    if(ps.executeUpdate(>0)
    {
        System.out.println("Appointment Booked Successfully");

    }
    else
    {
        System.out.println("Appointment is not Booked ");
    }
}
}

```

String d_id,app_date ;

```

public void checkavailability(String d_id,String app_date) throws SQLException
{
    Connection
conn=DriverManager.getConnection("jdbc:mysql://localhost:3306/hms","root","aparna");
    String query="select * from appointment where d_id=? and app_date=?";
    try(PreparedStatement ps=conn.prepareStatement(query))
    {

        ps.setString(1, d_id);
        ps.setString(2, app_date);
    }
}

```

```

        try(ResultSet rs=ps.executeQuery())
        {
            if(rs.next())
            {
                System.out.println(rs.getString(1) + "\t" + rs.getString(2)+
"\t" + rs.getString(3) + "\t" + rs.getString(4) + "\t" + rs.getString(5));
            }
            else
            {
                System.out.println("Doctor not Available");
            }
        }
    }
}

```

#Patients.java

```

package com.test2;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.SQLException;
import java.sql.ResultSet;
import java.util.Scanner;
import java.util.regex.Matcher;
import java.util.regex.Pattern;

public class Patients {

    public static final String ANSI_RED="\u001B[31m";
    public static final String ANSI_BLACK="\u001B[30m";

```

```

        public static final String ANSI_GREEN="\u001B[32m";
static Connection conn;

Scanner s=new Scanner(System.in);


        public void addPatient() throws SQLException
        {

            System.out.println("Enter patient id: ");
            String p_id=s.nextLine();

            System.out.println("Enter patient FirstName: ");
            String fname=s.nextLine();

            System.out.println("Enter patient LastName: ");
            String lname=s.nextLine();

            System.out.println("Enter patient Age: ");
            int age=s.nextInt();
            s.nextLine();

            System.out.println("Enter patient Gender(M/F/O): ");
            String gender=s.nextLine();

            if(!validateGender(gender))
            {
                System.out.println(ANSI_RED);
                System.out.println("Invalid input. Please enter 'M' for Male, 'F' for
Female, or 'O' for Other.");
            }

            System.out.println("Enter patient BloodGroup(A+,A-,B+,B-,AB+,AB-,O+,O-):
");

```

```
String bdg=s.nextLine();

if(!validateBloodGroup(bdg)) {
    System.out.println(ANSI_RED);
    System.out.println("Invalid Blood Group");
}

System.out.println("Enter patient address: ");
String address=s.nextLine();

System.out.println("Enter patient phoneNumber: ");
String phnumber=s.nextLine();

System.out.println("Enter Doctor id: ");
String d_id=s.nextLine();

conn=DriverManager.getConnection("jdbc:mysql://localhost:3306/hms","root","aparna");
String query="insert into patient values(?,?,?,?,?,?,?,?)";
try(PreparedStatement ps=conn.prepareStatement(query))
{
    ps.setString(1, p_id);
    ps.setString(2,fname);
    ps.setString(3, lname);
    ps.setInt(4, age);
    ps.setString(5,gender);
    ps.setString(6,bdg);
    ps.setString(7,phnumber);
    ps.setString(8,address);
    ps.setString(9, d_id);

    if(ps.executeUpdate(>0)
```

```
        {

            System.out.println("patient details added successfully");

        }
        else
        {

            System.out.println("Failed to add patient details");

        }
    }
}
```

```
private boolean validateGender(String gender) {

    System.out.println(ANSI_BLACK);
    String[] validGenders= {"M","F","O"};
    for(String validGender:validGenders) {
        if(validGender.equalsIgnoreCase(gender)) {
            return true;
        }
    }

    return false;

}
```

```
private boolean validateBloodGroup(String bloodGroup) {
    String regexp="^(A|B|AB|O)[+-]?$";
    return bloodGroup.matches(regexp);
}
```



```
public void viewPatient() throws SQLException {

conn=DriverManager.getConnection("jdbc:mysql://localhost:3306/hms","root","aparna");

String query="select * from patient";

try(PreparedStatement ps=conn.prepareStatement(query))
{
    try(ResultSet rs=ps.executeQuery())
    {

        while(rs.next())
        {

            String p_id=rs.getString("p_id");
            String fname=rs.getString("p_firstname");
            String lname=rs.getString("p_lastname");
            int age=rs.getInt("p_age");
            String gender=rs.getString("p_gender");
            String bdg=rs.getString("p_bloodgroup");
            String phnumber=rs.getString("phnumber");
            String address=rs.getString("p_address");
            String d_id=rs.getString("d_id");


            System.out.println("*****");
            System.out.println("Patient Id: " + p_id);
            System.out.println("First Name: " + fname);
            System.out.println("Last Name: "+ lname);
            System.out.println("Age: " + age);
            System.out.println("Gender: " + gender);
            System.out.println("BloodGroup: " + bdg);
            System.out.println("PhoneNumber: " + phnumber);
```

```

        System.out.println("Address: " + address);

        System.out.println("Doctor Id: "+ d_id);

        System.out.println("*****");

    }

}

}

}

public void getPatientById(String p_id) throws SQLException
{

    conn=DriverManager.getConnection("jdbc:mysql://localhost:3306/hms","root","aparna");

    String query="select * from patient where p_id=?";
    try(PreparedStatement ps=conn.prepareStatement(query))
    {

        ps.setString(1,p_id);

        try(ResultSet rs=ps.executeQuery())
        {

            if(rs.next())

            {

                System.out.println(rs.getString(1) + "\t" + rs.getString(2)+ "\t" +
rs.getString(3) + "\t" + rs.getString(4) + "\t" + rs.getString(5) + "\t" + rs.getString(6) + "\t" +
rs.getString(7) + "\t" + rs.getString(8) + "\t" + rs.getString(9));

            }

        }

    }

}

public void updatePatient() throws SQLException
{

    System.out.println("Enter patient id");

```

```
String p_id=s.nextLine();

System.out.println("Enter patient address to update");
String p_address=s.nextLine();
System.out.println("Enter phone number to update ");
String p_phnumber=s.nextLine();

conn=DriverManager.getConnection("jdbc:mysql://localhost:3306/hms","root","aparna");

String query="update patient set p_address=?, phnumber=? where
p_id=?";

try(PreparedStatement ps=conn.prepareStatement(query)){

    ps.setString(1, p_address);
    ps.setString(2, p_phnumber);
    ps.setString(3, p_id);

    int res = ps.executeUpdate();

    if(res>0)
    {

        System.out.println(p_id + " Patient details updated
successfully");

    }
    else
    {

        System.out.println("Failed to update patient details");

    }

}
```

```
}  
}
```

#Doctors.java

```
package com.test2;  
  
import java.sql.Connection;  
import java.sql.DriverManager;  
import java.sql.PreparedStatement;  
import java.sql.ResultSet;  
import java.sql.SQLException;  
import java.util.Scanner;  
import java.util.regex.Matcher;  
import java.util.regex.Pattern;  
  
public class Doctors {  
  
    public static final String ANSI_RED="\u001B[31m";  
    public static final String ANSI_BLACK="\u001B[30m";  
  
    static Connection conn;  
  
    Scanner s=new Scanner(System.in);  
  
    public void addDoctor() throws SQLException  
    {  
  
        System.out.println("Enter Doctor id: ");  
        String d_id=s.nextLine();  
  
        System.out.println("Enter Doctor FirstName: ");  
        String fname=s.nextLine();  
  
        System.out.println("Enter Doctor LastName: ");  
        String lname=s.nextLine();
```

```
System.out.println("Enter Doctor Gender(M/F/O): ");
String gender=s.nextLine();
if(!validateGender(gender))
{
    System.out.println(ANSI_RED);
    System.out.println("Invalid input. Please enter 'M' for Male, 'F' for Female, or
'O' for Other.");
}
```

```
System.out.println("Enter Doctor specialization: ");
String sp=s.nextLine();
```

```
System.out.println("Enter Doctor Email: ");
String email=s.nextLine();
if(!validateEmail(email))
{
    System.out.println(ANSI_RED);
    System.out.println("Invalid Email Format, Try Again!!!");
}
```

```
conn=DriverManager.getConnection("jdbc:mysql://localhost:3306/hms","root","aparna");
String query="insert into doctor values(?,?,?, ?, ?)";
try(PreparedStatement ps=conn.prepareStatement(query))
{

    ps.setString(1,d_id);
    ps.setString(2,fname);
    ps.setString(3,lname);
    ps.setString(4,gender);
    ps.setString(5,sp);
```

```

        ps.setString(6,email);

        if(ps.executeUpdate(>0)
        {
            System.out.println("Doctor details added successfully");

        }
        else
        {
            System.out.println("Failed to add Doctor details");
        }
    }
}

private boolean validateGender(String gender) {

    System.out.println(ANSI_BLACK);
    String[] validGenders= {"M","F","O"};
    for(String validGender:validGenders) {
        if(validGender.equalsIgnoreCase(gender)) {
            return true;
        }
    }

    return false;

}

public static boolean validateEmail(String email)

{

    String emailRegex="^[a-zA-Z0-9_+&*-]+(?:\\.[a-zA-z0-9_+&*-]+)*@(?:[a-zA-Z0-9-]+\\.)+[a-zA-Z]{2,7}$";

    Pattern pattern=Pattern.compile(emailRegex);

```

```

        Matcher matcher=pattern.matcher(email);

        return matcher.matches();
    }

    public void viewDoctor() throws SQLException {
        String query="select * from doctor";

        conn=DriverManager.getConnection("jdbc:mysql://localhost:3306/hms","root","aparna");

        try(PreparedStatement ps=conn.prepareStatement(query))
        {
            try(ResultSet rs=ps.executeQuery())
            {

                while(rs.next())
                {

                    String d_id=rs.getString("d_id");
                    String fname=rs.getString("d_firstname");
                    String lname=rs.getString("d_lastname");
                    String gender=rs.getString("d_gender");
                    String sp=rs.getString("specialization");
                    String email=rs.getString("email");

                    System.out.println("*****");
                    System.out.println("Doctor Id: " + d_id);
                    System.out.println("FirstName: " + fname);
                    System.out.println("LastName: " + lname);
                    System.out.println("Gender: " + gender);
                    System.out.println("Specialization: " + sp);
                    System.out.println("Email: " + email);
                    System.out.println("*****");

                }

            }

        }
    }

```

```

        }
    }

    public void getDoctorById(String id) throws SQLException
    {
        String query="select * from doctor where id=?";
        try(PreparedStatement ps=conn.prepareStatement(query))
        {
            ps.setString(1,id);
            try(ResultSet rs=ps.executeQuery())
            {
                if(rs.next())
                {
                    System.out.println(rs.getString(1) + "\t" + rs.getString(2)+
"\t" + rs.getString(3) + "\t" + rs.getString(4) + "\t" + rs.getString(5) + "\t" + rs.getString(6));
                }
            }
        }
    }

    String id,email;
    public void updateDoctor() throws SQLException
    {
        System.out.println("Enter Doctor id");
        String d_id=s.nextLine();

        System.out.println("Enter Doctor email to update");
        String email=s.nextLine();
    }
}

```



```

conn=DriverManager.getConnection("jdbc:mysql://localhost:3306/hms","root","aparna");

String query="update patient set d_email=? where d_id=?";

try(PreparedStatement ps=conn.prepareStatement(query))
{

    ps.setString(1, d_id);
    ps.setString(6, email);

    int res = ps.executeUpdate();

    if(res>0)
    {

        System.out.println(d_id + " Doctor details updated successfully");

    }
    else
    {

        System.out.println("Failed to update Doctor details");

    }

}

}

}

```

#Medicalhistories.java

```
package com.test2;
```

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.Scanner;

public class MedicalHistories {
    Connection conn;

    public void addMedicalHistory()throws SQLException{

        Scanner s=new Scanner(System.in);

        System.out.println("Enter MedicalRecord id:");
        String m_id=s.nextLine();

        System.out.println("enter Patient id:");
        String P_id=s.nextLine();

        System.out.println("enter Doctor id:");
        String d_id=s.nextLine();

        System.out.println("enter Appointment id:");
        String app_id=s.nextLine();

        System.out.println("enter date of examination(yy--mm--dd):");
        String date_of_examination=s.nextLine();

        System.out.println("enter diagnosis:");
```

```
String diagnosis=s.nextLine();
```

```
System.out.println("enter prescription:");
```

```
String prescription=s.nextLine();
```

```
conn=DriverManager.getConnection("jdbc:mysql://localhost:3306/hms","root","aparna");
```

```
String query="insert into medical_history values(?,?,?, ?, ?, ?)";
```

```
try(PreparedStatement ps=conn.prepareStatement(query))
```

```
{
```

```
ps.setString(1, m_id);
```

```
ps.setString(2,P_id);
```

```
ps.setString(3, d_id);
```

```
ps.setString(4,app_id);
```

```
ps.setString(5,date_of_examination);
```

```
ps.setString(6,diagnosis);
```

```
ps.setString(7,prescription);
```

```
//rs=ps.executeQuery();
```

```
if(ps.executeUpdate(>0)
```

```
{
```

```
System.out.println("Medical Record Details added successfully");
```

```
}
```

```
else
```

```
{
```

```
System.out.println("Failed to add Medical Record details");
```

```
}
```

```
}
```

```
}
```

```
public void viewMedicalHistory() throws SQLException {
```

```
conn=DriverManager.getConnection("jdbc:mysql://localhost:3306/hms","root","aparna");
String query="select * from medical_history";

try(PreparedStatement ps=conn.prepareStatement(query))
{
    try(ResultSet rs=ps.executeQuery())
    {

        while(rs.next())
        {
            String m_id=rs.getString("m_id");
            String P_id=rs.getString("p_id");
            String d_id=rs.getString("d_id");
            String app_id=rs.getString("app_id");
            String date_of_examination=rs.getString("dateofexamination");
            String diagnosis=rs.getString("diagnosis");
            String prescription=rs.getString("prescription");

            System.out.println("*****");
            System.out.println("MedicalRecord Id: " + m_id);
            System.out.println("Patient Id: " + P_id);
            System.out.println("Doctor Id: " + d_id);
            System.out.println("Appointment Id: " + app_id);
            System.out.println("Date Of Examination: " + date_of_examination);
            System.out.println("Diagnosis: " + diagnosis);
            System.out.println("Prescription: " + prescription);
            System.out.println("*****");

        }
    }
}
```

```

    }
    }
}

public void getMedicalHistoryById(String id) throws SQLException
{
    conn=DriverManager.getConnection("jdbc:mysql://localhost:3306/hms","root","aparna");
    String query="select * from medical_history where id=?";
    try(PreparedStatement ps=conn.prepareStatement(query))
    {
        ps.setString(1,id);
        try(ResultSet rs=ps.executeQuery())
        {
            if(rs.next())
            {
                System.out.println(rs.getString(1) + "\t" + rs.getString(2)+ "\t" +
rs.getString(3) + "\t" + rs.getString(4) + "\t" + rs.getString(5) + "\t" + rs.getString(6) + "\t" +
rs.getString(7));
            }
        }
    }
}
}

```

#Rooms.java

```

package com.test2;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

```

```
import java.sql.SQLException;
import java.util.Scanner;

public class Rooms {
    Connection conn;

    public void addRoom( )throws SQLException{

        Scanner s=new Scanner(System.in);

        System.out.println("enter Room no:");
        String rno=s.nextLine();

        System.out.println("enter Patient id:");
        String P_id=s.nextLine();

        System.out.println("enter room type:");
        String r_type=s.nextLine();

        System.out.println("enter duration:");
        String period=s.nextLine();

        conn=DriverManager.getConnection("jdbc:mysql://localhost:3306/hms","root","aparna");

        String query="insert into room values(?,?,?,?)";

        try(PreparedStatement ps=conn.prepareStatement(query))
        {
            ps.setString(1, rno);
            ps.setString(2, P_id);
            ps.setString(3, r_type);
            ps.setString(4, period);
```

```
        if(ps.executeUpdate()>0)
        {
            System.out.println("Room details added successfully");
        }
        else
        {
            System.out.println("Failed to add Room details");
        }
    }
}
```

```
public void viewRoom() throws SQLException {
```

```
    conn=DriverManager.getConnection("jdbc:mysql://localhost:3306/hms","root","aparna");
    String query="select * from room";
```

```
    try(PreparedStatement ps=conn.prepareStatement(query))
```

```
    {
        try(ResultSet rs=ps.executeQuery())
        {
```

```
            System.out.println("Room Details");
```

```
            while(rs.next())
```

```
            {
                String rmno=rs.getString("rno");
                String P_id=rs.getString("p_id");
                String r_type=rs.getString("r_type");
                String period=rs.getString("period");
```

```

        System.out.println("*****");

        System.out.println("Room No: " + rmno);

        System.out.println("Patient Id: " + P_id);

        System.out.println("Room Type: " + r_type);

        System.out.println("Period: " + period);

        System.out.println("*****");

    }

}

}

public void getRoomById(String rno) throws SQLException
{
    conn=DriverManager.getConnection("jdbc:mysql://localhost:3306/hms","root","aparna");
    String query="select * from room where rno=?";
    try(PreparedStatement ps=conn.prepareStatement(query))
    {
        ps.setString(1,rno);
        try(ResultSet rs=ps.executeQuery())
        {
            if(rs.next())
            {
                System.out.println(rs.getString(1) + "\t" + rs.getString(2)+ "\t" +
rs.getString(3) + "\t" + rs.getString(4));;

            }
        }
    }
}
}

```

#Validations.java

```
package com.test2;
```



```

import java.util.Scanner;
import java.util.regex.Matcher;
import java.util.regex.Pattern;

public class Validation {
    public static final String ANSI_RED="\u001B[31m";
    public static final String ANSI_BLACK="\u001B[30m";

    static Scanner s=new Scanner(System.in);
    public static String validateUserPassword()
    {
        boolean validUSePassword=false;
        String UserPassword="";
        while(validUSePassword)
        {
            System.out.println(ANSI_BLACK);

            //System.out.println("Enter New Password ");
            UserPassword=s.nextLine();

            String UserPasswordRegex="^(?=.*[a-z])(?=.*[A-Z])(?=.*\\d)(?=.*[@#$!%^&*()_+\\-
            =\\[\\]\\{\\};':\"\\\\\\\\\\/,.<>\\V?]).{8,}$";

            Pattern pattern=Pattern.compile(UserPasswordRegex);
            Matcher matcher=pattern.matcher(UserPassword);

            if(!matcher.matches())
            {
                System.out.println(ANSI_RED);

                System.out.println("Invalid Password Format.Password Should Contain
                Atleast 8 Characters,one Special Character,one Number,and one Uppercase letter.");

                continue;
            }

            validUSePassword=true;
        }
    }
}

```

```

    }

    return UserPassword;
}

public static String validatePhoneNumber() {
    boolean validPhoneNumber = false;
    String phoneNumber = "";

    while (validPhoneNumber) {
        System.out.print(ANSI_BLACK);
        //System.out.print("Enter Phone number : ");
        phoneNumber = s.nextLine();
        String phoneNumberRegex = "^[0-9]{10}$";
        Pattern pattern = Pattern.compile(phoneNumberRegex);
        Matcher matcher = pattern.matcher(phoneNumber);

        if (!matcher.matches()) {
            System.out.print(ANSI_RED);
            System.out.println("Invalid phone number format. Please enter a 10-
digit number.");

            continue;
        }

        validPhoneNumber = true;
    }

    return phoneNumber;
}
}

```