```
package hotelreservationsystem;



import java.util.Scanner;
import java.time.LocalDate;
import java.time.format.DateTimeFormatter;
import java.time.format.DateTimeParseException;
import java.util.HashSet;
import java.util.InputMismatchException;
import java.util.Set;

public class HotelReservationSystem {
    private Room room;
    private static int nextSingleRoomNumber = 100;
    private static int nextDoubleRoomNumber = 200;
    private static int nextSuiteRoomNumber = 300;
    private static String adminUsername = "admin";
    private static String adminPassword = "java";



   public static int ToReadInt(Scanner scanner, String prompt) {
        System.out.print(prompt);
        return ToReadInt(scanner);
    }

   private static Set<Room> rooms = new HashSet<>();

    public static int getNextSingleRoomNumber() {
        return nextSingleRoomNumber;
    }
```

```
public static int getNextDoubleRoomNumber() {
    return nextDoubleRoomNumber;
}

public static int getNextSuiteRoomNumber() {
    return nextSuiteRoomNumber;
}



public static void addRoom(Room room) {
    if (rooms.add(room)) {
    System.out.println("Room added successfully: Room #" + room.getRoomNumber());
    }else {
    System.out.println("Room already exists: Room #" + room.getRoomNumber());
    }
}

public static Set<Room> getRooms() {
    return rooms;
}

private static boolean AdminLogin(Scanner scanner) {
    System.out.println("---- Admin Login ----");
    System.out.print("Enter Username: ");
    String username = scanner.nextLine();

    System.out.print("Enter Password: ");
    String password = scanner.nextLine();

    if (username.equals(adminUsername) && password.equals(adminPassword)) {
        System.out.println("Login successful! Welcome, Admin.");
        return true;
```

```
  } else {
     System.out.println("Please try again!!!!!!");
     return false;
  }
}

public static void AdminMenu(Scanner scanner){
  boolean exit= false;

  while(!exit){
     System.out.println("------------------------------------------");
     System.out.println("                Admin Menu                 ");
     System.out.println("------------------------------------------");
     System.out.println("1- View All Booking");
     System.out.println("2- Update Booking");
     System.out.println("3- Add New Rooms");
     System.out.println("4- Delete Booking");
     System.out.println("5- Back to Main Menu");

     System.out.println("------------------------------------------");
     System.out.print("Enter Your Choice: ");

    int choice = ToReadInt(scanner);

    switch(choice){
       case 1:
           ViewAllBookings();
          break;
       case 2:
           UpdateBookingAdmin(scanner);
          break;
       case 3:
```

```
                AddNewRooms(scanner);
                break;
            case 4:
                DeleteBookingAdmin(scanner);
                break;
            case 5:
                exit= true;
                break;
            default:
                System.out.println("Invalid Choice!!!");
        }
    }
}

public static void CustomerMenu(Scanner scanner){
    boolean exit = false;

    while(!exit){

        System.out.println("--------------------------------------------");
        System.out.println("           Customer Menu                 ");
        System.out.println("--------------------------------------------");
        System.out.println("1- Book a Room");
        System.out.println("2- View Booking");
        System.out.println("3- Cancel Booking");
        System.out.println("4- Back to Main Menu");



        System.out.println("--------------------------------------------");
        System.out.print("Enter Your Choice: ");
```

```
            int choice = ToReadInt(scanner);

            switch(choice){
                case 1:
                    BookRoom(scanner);
                    break;
                case 2:
                    ViewBooking(scanner);
                    break;
                case 3:
                    CancelBooking(scanner);
                    break;
                case 4:
                    exit= true;
                    break;
                default:
                    System.out.println("Invalid Choice!!!");
            }
        }

    }

// Customer Menu

    public static void BookRoom(Scanner scanner){
        System.out.print("Enter Your Name: ");
        scanner.nextLine();
        String name= scanner.nextLine();

        long fayda= ToReadLong(scanner,"Enter Your Fayda No: " );

        int phone = ToReadInt(scanner,"Enter Your Phone No: "  );
```

```
scanner.nextLine();
System.out.print("Enter Your Address: ");
String address= scanner.nextLine();

Guest guest = new Guest(fayda,name, phone, address);

System.out.println("~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~");
System.out.println("           Select Room Type:           ");
System.out.println("~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~");
System.out.println("             1. Single Room            ");
System.out.println("             2. Double Room            ");
System.out.println("             3. Suite Room             ");
System.out.print("Enter your choice : ");
int choice = scanner.nextInt();

int nof = ToReadInt(scanner,"Enter Number of Nights: "  );

scanner.nextLine();

String date;
 while (true) {
    System.out.print("Enter your check-in date (year-month-day): ");
    date = scanner.nextLine();

    try {
        LocalDate Ldate = LocalDate.parse(date, DateTimeFormatter.ISO_LOCAL_DATE);
                //past date
        if (Ldate.isBefore(LocalDate.now())) {
        System.out.println("Check-in date cannot be in the past.");
        continue;
        }
```

```
        break;
    }
    catch (DateTimeParseException e) {
        System.out.println("Invalid date format. Please enter in year-month-day format.");
    }
}

Room room = null;
 int roomNumber;
 switch (choice) {
    case 1:
         if(nextSingleRoomNumber > 150){
            System.out.println("No more Single Rooms available.");
            return;
        }
        roomNumber = nextSingleRoomNumber++;
        room = new SingleRoom(roomNumber);
        break;
    case 2:
        if(nextDoubleRoomNumber > 250){
            System.out.println("No more Double Rooms available.");
            return;
        }
        roomNumber = nextDoubleRoomNumber++;
        room = new DoubleRoom(roomNumber);
        break;
    case 3:
        if(nextSuiteRoomNumber > 350){
            System.out.println("No more Suite Rooms available.");
            return;
        }
        roomNumber = nextSuiteRoomNumber++;
        room = new SuiteRoom(roomNumber);
        break;
      default:
      System.out.println("Invalid Choice.");
      return;



  }
 //null pointer exception
     if (room == null) {
  System.out.println("Room creation failed. Please check input.");
  return;
 }
  try {
    room.setNumber_of_Nights(nof);// dreferencing null pointer
  }
  catch(IllegalArgumentException iae){
    System.out.println("Error: " + iae.getMessage());
    return;
  }
  room.setNumber_of_Nights(nof);
  addRoom(room);
  Bookings booking = new Bookings(guest, room, date);
  room.Calculate_Total_Price();
  booking.showBookingDetails();


}


  public static void ViewBooking(Scanner scanner) {
  long fayda =ToReadLong(scanner, "Enter your Fayda Number to view booking: ");
  scanner.nextLine();
```

```
    Bookings.viewBookingByFayda(fayda);
}


    public static void CancelBooking(Scanner scanner){
    long fayda =ToReadLong(scanner, "Enter your Fayda Number to Cancel booking: ");
    scanner.nextLine();


    Bookings.cancelBookingByFayda(fayda);
    }


// Admins side
    public static void AddNewRooms(Scanner scanner) {
        int nextSingleRoomNumber = 150;
        int nextdoubleRoomNumber = 250;
        int nextSuiteRoomNumber = 350;
        System.out.println("~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~");
        System.out.println("              Select Room Type to Add:          ");
        System.out.println("~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~");
        System.out.println("                  1. Single Room               ");
        System.out.println("                  2. Double Room               ");
        System.out.println("                  3. Suite Room                ");


    int roomTypeChoice = ToReadInt(scanner);
    int numberOfRooms = ToReadInt(scanner, "Enter the number of rooms to add: ");


        for (int i = 0; i < numberOfRooms; i++) {
            int roomNumber;
            Room newRoom;


        switch (roomTypeChoice) {
            case 1:
```

```
          roomNumber = nextSingleRoomNumber++;
          newRoom = new SingleRoom(roomNumber);
          break;
      case 2:

        roomNumber = nextDoubleRoomNumber++;
          newRoom = new DoubleRoom(roomNumber);
          break;
      case 3:

        roomNumber = nextSuiteRoomNumber++;
          newRoom = new SuiteRoom(roomNumber);
          break;
      default:
        System.out.println("Invalid room type.");
        return;
    }

    addRoom(newRoom); //room manager
  }

  System.out.println(numberOfRooms + " new rooms added successfully.");
}

  public static void ViewAllBookings() {
  if (Bookings.getAllBookings().isEmpty()) {
    System.out.println("No bookings found.");
  } else {
    for (Bookings booking : Bookings.getAllBookings()) {
      booking.showBookingDetails();
    }
    }
}

    public static void UpdateBookingAdmin(Scanner scanner) {
     long fayda =ToReadLong(scanner, "Enter your Fayda Number to view booking: ");
        scanner.nextLine();

    Bookings booking = Bookings.findBookingByFayda(fayda);

    if (booking != null) {
        System.out.println("Select what you want to update:");
        System.out.println("1. Change Room Type");
        System.out.println("2. Change Check-In Date");

        int updateChoice = ToReadInt(scanner);
        Room newRoom = null;
        scanner.nextLine();

        switch (updateChoice) {
            case 1:
                System.out.println("Select New Room Type:");
                System.out.println("1. Single Room");
                System.out.println("2. Double Room");
                System.out.println("3. Suite Room");



            int type = ToReadInt(scanner);
                    int newRoomNumber;
                switch (type) {
                    case 1:
                        if(nextSingleRoomNumber > 150){
                                System.out.println("No more Single Rooms available.");
                    return;
                }
                newRoomNumber = nextSingleRoomNumber++;
                newRoom = new SingleRoom(newRoomNumber);
                break;
        case 2:
            if(nextDoubleRoomNumber > 250){
                    System.out.println("No more Double Rooms available.");
                    return;
                }
                newRoomNumber = nextDoubleRoomNumber++;
                newRoom = new DoubleRoom(newRoomNumber);
                break;

        case 3:
            if(nextSuiteRoomNumber > 350){
                    System.out.println("No more Suite Rooms available.");
                    return;
                }
                newRoomNumber = nextSuiteRoomNumber++;
                newRoom = new SuiteRoom(newRoomNumber);
                break;
    }

    booking.setRoom(newRoom);
    addRoom(newRoom);
    System.out.println("Room type updated.");
    break;

case 2:
    System.out.print("Enter new check-in date (yyyy-mm-dd): ");
    String newDate = scanner.nextLine();
```

```
                booking.setCheckInDate(newDate);
                System.out.println("Check-in date updated.");
                break;

            default:
                System.out.println("Invalid choice.");
        }
    } else {
        System.out.println("No booking found for " + fayda);
    }
}

    public static void DeleteBookingAdmin(Scanner scanner) {

        long fayda =ToReadLong(scanner, "Enter your Fayda Number to view booking: ");
        scanner.nextLine();
        Bookings.deleteBookingByFayda(fayda);
}
//Main Menu
    public static void main(String[] args) {
        // TODO code  logic here

        Scanner scanner = new Scanner(System.in);
        boolean exit = false;

        while(!exit){
        System.out.println("----------------------------------------------------------");
        System.out.println("        Welcome to Our Hotel Reservation System!!!!!        ");
        System.out.println("----------------------------------------------------------");
        System.out.println("1- Customer Menu");
        System.out.println("2- Admin Menu ");
        System.out.println("3- Exit");
    System.out.print("Enter your choice: ");
   int choice = ToReadInt(scanner);
    scanner.nextLine();

    switch (choice) {
        case 1:
            CustomerMenu(scanner);
            break;
        case 2:
            if(AdminLogin(scanner)){
                AdminMenu(scanner);
            }
            break;
        case 3:
            exit=true;
            System.out.println("Thank you for using Our Hotel Reservation System!");
            break;
        }
    }

  scanner.close();
}

  public static int ToReadInt(Scanner scanner) {
    while (true) {
        try {
            return scanner.nextInt();
        } catch (InputMismatchException e) {
            System.out.print("Please enter a valid number: ");
            scanner.nextLine();
        }
        }
    }

    public static long ToReadLong(Scanner scanner, String prompt) {
    System.out.print(prompt);
    while (true) {
        try {
            return scanner.nextLong();
        } catch (InputMismatchException e) {
            System.out.print("Please enter a valid long number: ");
            scanner.nextLine();
        }
    }
}
}

package hotelreservationsystem;



public abstract class Room {
    private int RoomNumber;
    private int Number_of_Nights;

    public Room(int RoomNumber) {
        this.RoomNumber = RoomNumber;
    }
    public abstract double CalculatePricePerNight();
    public abstract String getRoomtype();



    public int getRoomNumber() {
        return RoomNumber;
    }

    public void setRoomNumber(int RoomNumber) {
        this.RoomNumber = RoomNumber;
    }



    public int getNumber_of_Nights() {
        return Number_of_Nights;
    }

    public void setNumber_of_Nights(int Number_of_Nights) {
        if(Number_of_Nights<=0)
            throw new IllegalArgumentException ("Number of Nights must greater than 0.");
        this.Number_of_Nights = Number_of_Nights;
    }

   public double Calculate_Total_Price(){
       return Number_of_Nights * CalculatePricePerNight();
   }



}

package hotelreservationsystem;


public class SingleRoom extends Room {
    private static final double PricePerNight= 1000.00;

    public SingleRoom(int RoomNumber) {
        super(RoomNumber);
    }

    @Override
     public double CalculatePricePerNight(){
         return PricePerNight;
     }
    @Override
    public String getRoomtype(){
        return "Single Room!!";
    }
}

package hotelreservationsystem;


public class SuiteRoom extends Room {
    private static final double PricePerNight= 3000.00;

    public SuiteRoom(int RoomNumber) {
        super(RoomNumber);
    }

    @Override
     public double CalculatePricePerNight(){
         return PricePerNight;
     }
    @Override
    public String getRoomtype(){
        return "Suite Room!!";
```

```
  }

}
package hotelreservationsystem;


import java.time.LocalDate;
import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;

public class Bookings {

    private Guest guest;
    private Room room;
    private int numberOfNights;
    private String checkInDate;
    private String checkOutDate;

    private static final List<Bookings> bookingsList = new ArrayList<>();

    public Bookings(Guest guest, Room room, String checkInDate) {
        this.guest = guest;
        this.room = room;
        this.checkInDate = checkInDate;
        this.room.Calculate_Total_Price();
        bookingsList.add(this);
```

```
}

public int getRoomNumber() {
    return room.getRoomNumber();
}

public Guest getGuest() {
    return guest;
}

public void setRoom(Room room) {
    this.room = room;
}

public void setCheckInDate(String date) {
    this.checkInDate = date;
}

public String getCheckInDate() {
    return checkInDate;
}

public String getCheckOutDate() {
    return checkOutDate;
}

public void setCheckOutDate(String checkOutDate) {
    this.checkOutDate = checkOutDate;
}
```

```
public static void addBooking(Bookings booking) {
    bookingsList.add(booking);
}

public static void cancelBookingByFayda(long fayda) {
    Bookings booking = findBookingByFayda(fayda);
    if (booking != null) {
        bookingsList.remove(booking);
        System.out.println("Booking canceled successfully for " + fayda);
    } else {
        System.out.println("No booking found for " + fayda);
    }
}

public static void viewBookingByFayda(long fayda) {
    Bookings booking = findBookingByFayda(fayda);
    if (booking != null) {
        booking.showBookingDetails();
    } else {
        System.out.println("No booking found for " + fayda);
    }
}

public static void updateBookingByFayda(long fayda, Scanner scanner) {
    Bookings booking = findBookingByFayda(fayda);
    if (booking != null) {
        System.out.println("Enter new check-in date (year-month-day): ");
        String newDate = scanner.nextLine();
        booking.setCheckInDate(newDate);
        System.out.println("Booking updated successfully.");
    } else {
        System.out.println("No booking found for " + fayda);
    }
}

public static void deleteBookingByFayda(long fayda) {
    cancelBookingByFayda(fayda);
}

public static Bookings findBookingByFayda(long fayda) {


    for (Bookings b : bookingsList) {
        if (b.getGuest().getFayda()==fayda) {
            return b;
        }
    }
    return null;
}


public void showBookingDetails() {
    double totalPrice = room.Calculate_Total_Price();
    guest.displayGuest();
    System.out.println("Room Type: " + room.getRoomtype());
    System.out.println("Room Number: " + room.getRoomNumber());
    System.out.println("Number of Nights: " + room.getNumber_of_Nights());
```

```
        System.out.println("Total Price: " + totalPrice + " ETB.");
        System.out.println("Check-In Date: " + (checkInDate != null ? checkInDate : "Not Set"));

        if (checkInDate != null) {
            LocalDate checkIn = LocalDate.parse(checkInDate);
            LocalDate checkOut = checkIn.plusDays(room.getNumber_of_Nights());
            System.out.println("Check-Out Date: " + checkOut);
        } else {
            System.out.println("Check-Out Date: Not Set");
        }
    }

    public static List<Bookings> getAllBookings() {
        return bookingsList;
    }
}

package hotelreservationsystem;


public class DoubleRoom extends Room {

     private static final double PricePerNight= 2000.00;

    public DoubleRoom(int RoomNumber) {
        super(RoomNumber);
    }

    @Override
```

```
     public double CalculatePricePerNight(){
         return PricePerNight;
     }
    @Override
    public String getRoomtype(){
        return "Double Room!!";
    }

}

package hotelreservationsystem;

public class Guest {
    private long fayda;
    private String guestName;
    private int PhoneNo;
    private String Address;



   public Guest(long fayda ,String guestName, int Phone, String Address) {
        this.fayda = fayda;
        this.guestName = guestName;
        this.PhoneNo = Phone;
        this.Address = Address;

    }

    public String getGuestName() {
        return guestName;
```

```
}

public void setGuestName(String guestName) {
    this.guestName = guestName;
}

public int getPhone() {
    return PhoneNo;
}

public void setPhone(int Phone) {
    this.PhoneNo = Phone;
}

public String getAddress() {
    return Address;
}

public void setAddress(String Address) {
    this.Address = Address;
}


public long getFayda() {
    return fayda;
}

public void setFayda(long fayda) {
    this.fayda = fayda;
    }



    public void displayGuest(){
        System.out.println("*********************************************");
        System.out.println("            Guest Details:              ");
        System.out.println("*********************************************");
        System.out.println("FAYDA Number: " + fayda);
        System.out.println("Name: "+ guestName);
        System.out.println("Phone No: "+ PhoneNo);
        System.out.println("Address: " + Address);


    }


}
```