**Hotel Reservation System**

**Abstract:-**

The **Hotel Reservation System** is a console-based application designed to automate and streamline the process of hotel booking. The system provides an interactive interface for users to register, log in, browse available hotels, select rooms, and make payments. By inputting details such as check-in/check-out dates, users can receive a summary of their bookings and confirm their reservations. Additionally, the system allows payment through credit cards or UPI. The program makes use of Java’s date and time handling features, random number generation for OTP validation, and formatted outputs to enhance the user experience.

**Introduction:-**

The Hotel Reservation System is designed to simplify the booking process for customers seeking to book hotel rooms. It provides a user-friendly interface for registering, logging in, and selecting from a list of available hotels and rooms. Users can input check-in and check-out dates, calculate the number of nights, and determine the total cost of their stay based on room type. To secure the user's identity, the system uses an OTP validation process. Upon successful login, users can browse hotels, select rooms, and make payments either through credit cards or UPI. This system simulates a real-world booking scenario with detailed booking summaries and time-stamped reservation logs.

**Objective of the Hotel Reservation System:-**

The primary objective of the Hotel Reservation System project is to automate and streamline the process of booking hotel rooms, providing a convenient and efficient way for users to search, select, and reserve accommodations. The key objectives of the project are as follows:

1. User Registration and Authentication:
   * To allow users to register and log in securely by validating their identity through a one-time password (OTP) system.
   * Provide a secure login mechanism to ensure only authorized users can make reservations.
2. Browse and Select Hotels:
   * To offer users a list of available hotels to choose from based on their preferences and locations.
   * Display detailed information about each hotel, including room types and pricing.
3. Room Reservation:
   * Enable users to select from different room types (e.g., single room, double room, suite) based on availability and preferences.
   * Allow users to input check-in and check-out dates, and calculate the number of nights for the stay.
4. Calculate Total Cost:
   * Automatically calculate the total cost of the reservation based on the room type selected and the number of nights the user will stay.
   * Ensure the pricing is accurate, with room costs reflecting the number of nights.
5. Payment Integration:
   * Provide users with payment options such as credit card or UPI for the reservation.
   * Ensure secure and accurate payment processing by validating the payment amount before confirming the reservation.
6. Reservation Confirmation and Summary:
   * After successful payment, generate a summary of the reservation, including hotel details, room type, check-in/check-out dates, and the total cost.
   * Confirm the booking to the user and provide the necessary information for their stay.

By achieving these objectives, the Hotel Reservation System will serve as an automated tool for users to efficiently plan and book their hotel stays while ensuring that the process is secure, seamless, and user-friendly.

**Tech Stack of the Hotel Reservation System:-**

The Hotel Reservation System is developed using a set of foundational tools and technologies that ensure the project is efficient, scalable, and easy to maintain. Below is the tech stack used for the project:

Frontend:-

• Java Console (CLI): The primary interface for user interaction in this project, providing a simple text-based input/output mechanism.

Backend:-

• Java: The core programming language used for the entire application. Java provides object-oriented principles, robust libraries, and scalability, making it ideal for building management systems. Development Tools:

• JDK (Java Development Kit): For compiling and running the Java application.

• IDE (Integrated Development Environment): Tools such as Eclipse, IntelliJ IDEA, or NetBeans can be used for coding and debugging.

Operating Environment:-

• Platform Independent: As Java is platform-independent, the system can run on Windows, macOS, or Linux with Java Runtime Environment (JRE) installed.

This tech stack ensures the project is lightweight, adaptable, and easy to deploy while maintaining the simplicity needed for a beginner-level system.

**System Requirements for Hotel Reservation System:-**

The Hotel Reservation System requires specific hardware and software configurations to ensure smooth development, deployment, and execution. Below are the recommended system requirements :

**Hardware Requirements:-**

1.Processor:

* Minimum: Intel Core i3 or equivalent
* Recommended: Intel Core i5/i7 or equivalent for faster performance

2.RAM:

* Minimum: 4 GB
* Recommended: 8 GB or higher for multitasking during development

3. Storage:

* Minimum: 500 MB of free disk space (for Java installation, IDE, and project files)
* Recommended: SSD with at least 2 GB free space for faster read/write operations

4. Display:

* Minimum resolution: 1024x768
* Recommended resolution: 1366x768 or higher

5. Keyboard and Mouse:

* Required for navigation and input in CLI or GUI.

**Software Requirements:-**

1.Operating System:

* Windows 10/11
* macOS (Big Sur or later)
* Linux (Ubuntu 18.04 or later)

2.Java Development Kit (JDK):

* Minimum: JDK 8
* Recommended: JDK 17 or later for enhanced features and performance

3. Java Runtime Environment (JRE):

* Ensure the JRE is installed to run Java applications.

4. Integrated Development Environment (IDE):

* Recommended IDEs:
* Eclipse
* IntelliJ IDEA
* NetBeans o Alternatively, a lightweight text editor like VS Code with Java extensions can be used.

5.Compiler:

* Java compiler (included in the JDK).

**Java code :-**

import java.time.LocalDate;

import java.time.LocalDateTime;

import java.time.format.DateTimeFormatter;

import java.time.temporal.ChronoUnit;

import java.util.\*;

public class HotelReservation{

    public static void main(String[] args) {

        Scanner sc=new Scanner(System.in);

        System.out.println("Welcome to Hotel Reservation System!");

        System.out.println("1. Register/Login");

        System.out.println("2. Browse Hotels");

        System.out.println("3. Exit");

        System.out.println("Choose any one of them");

        int t=sc.nextInt();

        User  user=new User();

        switch(t){

            case 1:

                boolean a=user.login();

                if(a){

                    System.out.println("\nLogin Successful!\n");

                }

                else{

                    System.out.println("Login Unsucccessfull! Invalid Code Entered Please Verify your Code");

                    return;

                }

                break;

            case 2:

                user.hotels();

                return;

            case 3:

                return;

            default:

                System.out.println("Invalid Choose");

                break;

        }

        System.out.println("\nAvailable Hotels:");

        String[] hotel=user.hotels();

        System.out.println("\nSelect the Hotel you want to Book:");

        int m=sc.nextInt();

        String[] rooms={"Single Room","Double Room","suite"};

        int[] p={100,150,300};

        boolean b=false;

        int r=0;

        long price=0;

        String checkin="";

        String checkout="";

        long nights=0;

        long total=0;

        while(b==false){

            if(m>0 && m<5){

                b=true;

                System.out.println("\nRooms at "+hotel[m-1]+" : ");

                for(int i=0;i<rooms.length;i++){

                    System.out.println((i+1)+". "+rooms[i]+" ($" +p[i]+" per night)");

                }

                System.out.println("\nselect the room you want to book:");

                r=sc.nextInt();

                boolean c=false;

                while(c==false){

                    if(r>0 && r<4){

                        c=true;

                        System.out.println("\nSelected Room : "+rooms[r-1]);

                    }

                    else{

                        System.out.println("invalid choice try again:");

                        r=sc.nextInt();

                    }

                }

                System.out.println("\nEnter Check-in Date (YYYY-MM-DD) : ");

                sc.nextLine();

                checkin=sc.nextLine();

                System.out.println("Enter Check-out Date (YYYY-MM-DD) : ");

                checkout=sc.nextLine();

                nights=user.date(checkin,checkout);

                System.out.println("\nReservation Summary: \n");

                System.out.println("Hotel: "+hotel[m-1]);

                System.out.println("Room: "+rooms[r-1]);

                System.out.println("Check-in: "+checkin);

                System.out.println("Check-out: "+checkout);

                boolean n=false;

                while(n==false){

                    if(nights>0){

                        System.out.println("Total Nights: "+nights);

                        n=true;

                    }

                    else{

                        System.out.println("Please make sure You enter the dates correctly!");

                        System.out.println("\nEnter Check-in Date (YYYY-MM-DD) : ");

                        checkin=sc.nextLine();

                        System.out.println("Enter Check-out Date (YYYY-MM-DD) : ");

                        checkout=sc.nextLine();

                        nights=user.date(checkin,checkout);

                    }

                }

                total=p[r-1]\*nights;

                System.out.println("Total Price: "+total);

            }

            else{

                System.out.println("invalid chocie Try Again :");

                m=sc.nextInt();

            }

        }

        System.out.println("\nDo you want to Confirm Reservation(yes/no): ");

        String str=sc.nextLine();

        boolean tr=false;

        while(tr==false){

        if(str.equals("yes") || str.equals("Yes") || str.equals("YES")){

             user.payment(total);

             tr=true;

        }

        else if(str.equals("no")||str.equals("No")||str.equals("NO")){

            return;

        }

        else{

            System.out.println("Please enter yes/no: ");

            str=sc.nextLine();

        }

         }

        System.out.println("\nYour Reservation details are :\n");

        System.out.println("Hotel: "+hotel[m-1]);

        System.out.println("Room: "+rooms[r-1]);

        System.out.println("Check-in: "+checkin);

        System.out.println("Check-out: "+checkout);

        System.out.println("Total Nights: "+nights);

        user.time();

        System.out.println("1. Logout");

        int c=sc.nextInt();

        boolean l=false;

        while(l==false){

        if(c==1){

            l=true;

            System.out.println("\nThanks for the Booking!");

            System.out.println("Hope you will enjoy the stay!\n");

        }

        else{

            System.out.println("Please Enter 1 to Logout");

            c=sc.nextInt();

        }

        }

    }

}

class User{

    static String name;

    static String email;

    static String password;

    static int uc;

   public boolean login(){

        Scanner sc=new Scanner(System.in);

        System.out.println("\nEnter Your Name : ");

        name=sc.nextLine();

        formatter();

        System.out.println("Enter Your Email :");

        User.email=sc.nextLine();

        System.out.println("Enter Your Mail ID Password :");

        User.password=sc.nextLine();

        System.out.println("Enter the below provided OTP :");

        int rc=random();

        System.out.println(rc);

        User.uc=sc.nextInt();

        if(rc==uc){

            return true;

        }

        return false;

   }

   public int random(){

    Random ran=new Random();

    int rc=ran.nextInt(10000);

    return rc;

   }

   public void formatter(){

        Formatter f=new Formatter();

        f.format("\nWelcome, %s!\n",User.name);

        System.out.println(f);

        f.close();

   }

   public void time(){

    LocalDateTime currentDateTime=LocalDateTime.now();

    DateTimeFormatter formatter=DateTimeFormatter.ofPattern("dd-mm-yyyy HH:mm:ss\n");

    String formattedDate=currentDateTime.format(formatter);

    System.out.println(formattedDate);

    }

   public String[] hotels(){

    String hotel[]={"Grand Plaza (Location: New York)","Sea View Resort (Location: Miami)","Taj Lake Palace (Location: Udaipur)","Marina Bay Sands (Location: Singapore)","Taj Hotel (Location: Mumbai)"};

    for(int i=0;i<hotel.length;i++){

        System.out.println((i+1)+". "+hotel[i]);

    }

    return hotel;

}

public void payment(long price){

    Scanner sc=new Scanner(System.in);

    System.out.println("\nEnter Payment Mode");

    System.out.println("1. Credit Card");

    System.out.println("2. UPI");

    int pay=sc.nextInt();

    if(pay==1){

        System.out.println("\nEnter Credit Card details:");

        System.out.println("Enter credit card number: ");

        int acc=sc.nextInt();

        System.out.println("Enter your credit pin :");

        int pin=sc.nextInt();

        System.out.println("Enter "+price+" rupees: ");

        int amo=sc.nextInt();

        boolean a=false;

        while(a==false){

            if(price==amo){

                a=true;

                System.out.println("\nPayment Successful!\n");

            }

            else{

                System.out.println("\nInvalid amount entered please Enter correct amount :");

                amo=sc.nextInt();

            }

        }

    }

    else if(pay==2){

        System.out.println("Enter your UPI pin:");

        int pin=sc.nextInt();

        System.out.println("\nEnter "+price+" rupees: ");

        int amo=sc.nextInt();

        boolean a=false;

        while(a==false){

            if(price==amo){

                a=true;

                System.out.println("\nPayment Successful!\n");

            }

            else{

                System.out.println("\nInvalid amount entered please Enter correct amount :");

                amo=sc.nextInt();

            }

        }

    }

}

   public long date(String str1,String str2) {

    String dateBeforeString = str1;

    String dateAfterString = str2;

    //Parsing the date

    LocalDate dateBefore = LocalDate.parse(dateBeforeString);

    LocalDate dateAfter = LocalDate.parse(dateAfterString);

    //calculating number of days in between

    long noOfDaysBetween = ChronoUnit.DAYS.between(dateBefore, dateAfter);

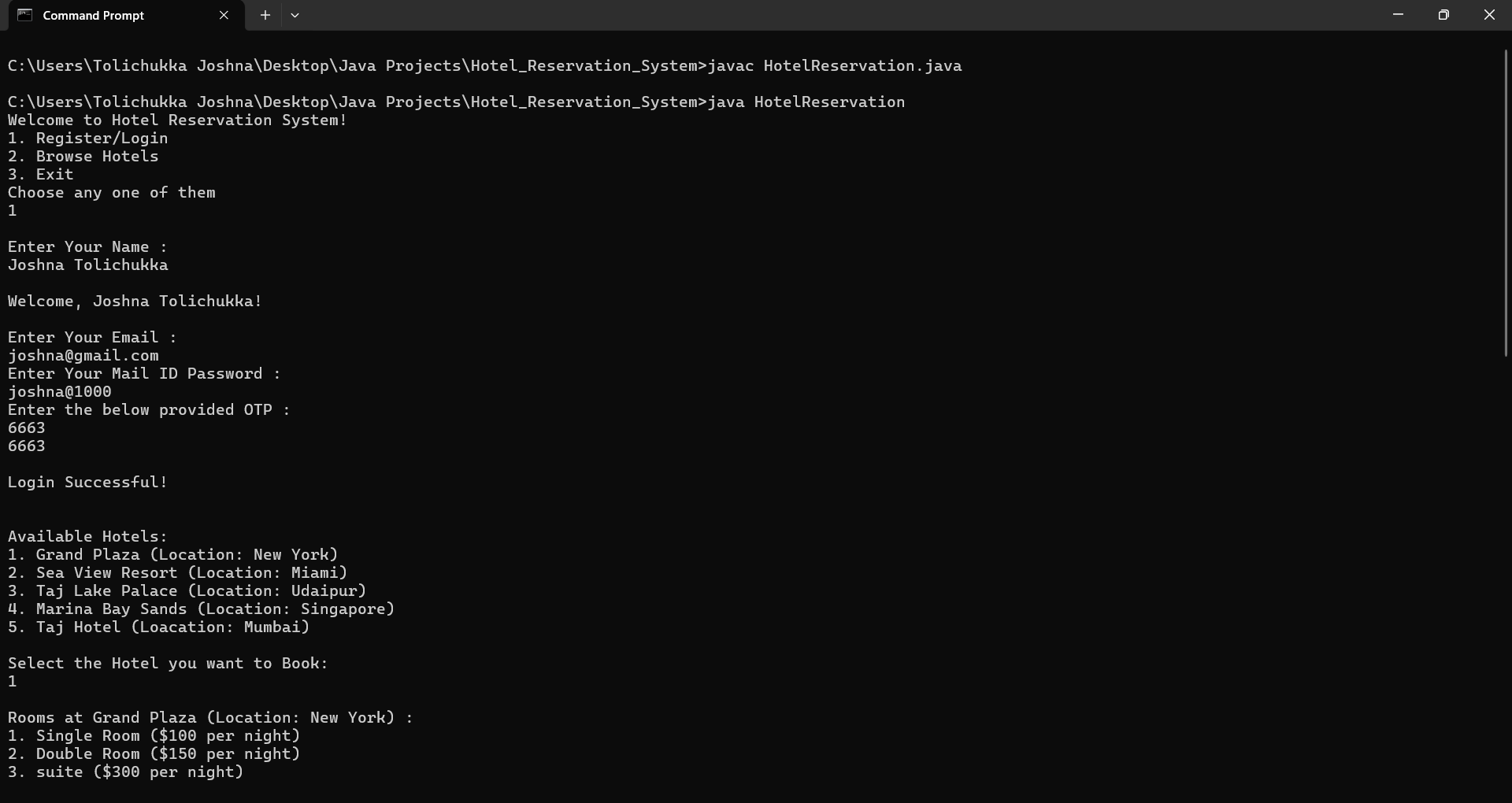
    //displaying the number of days

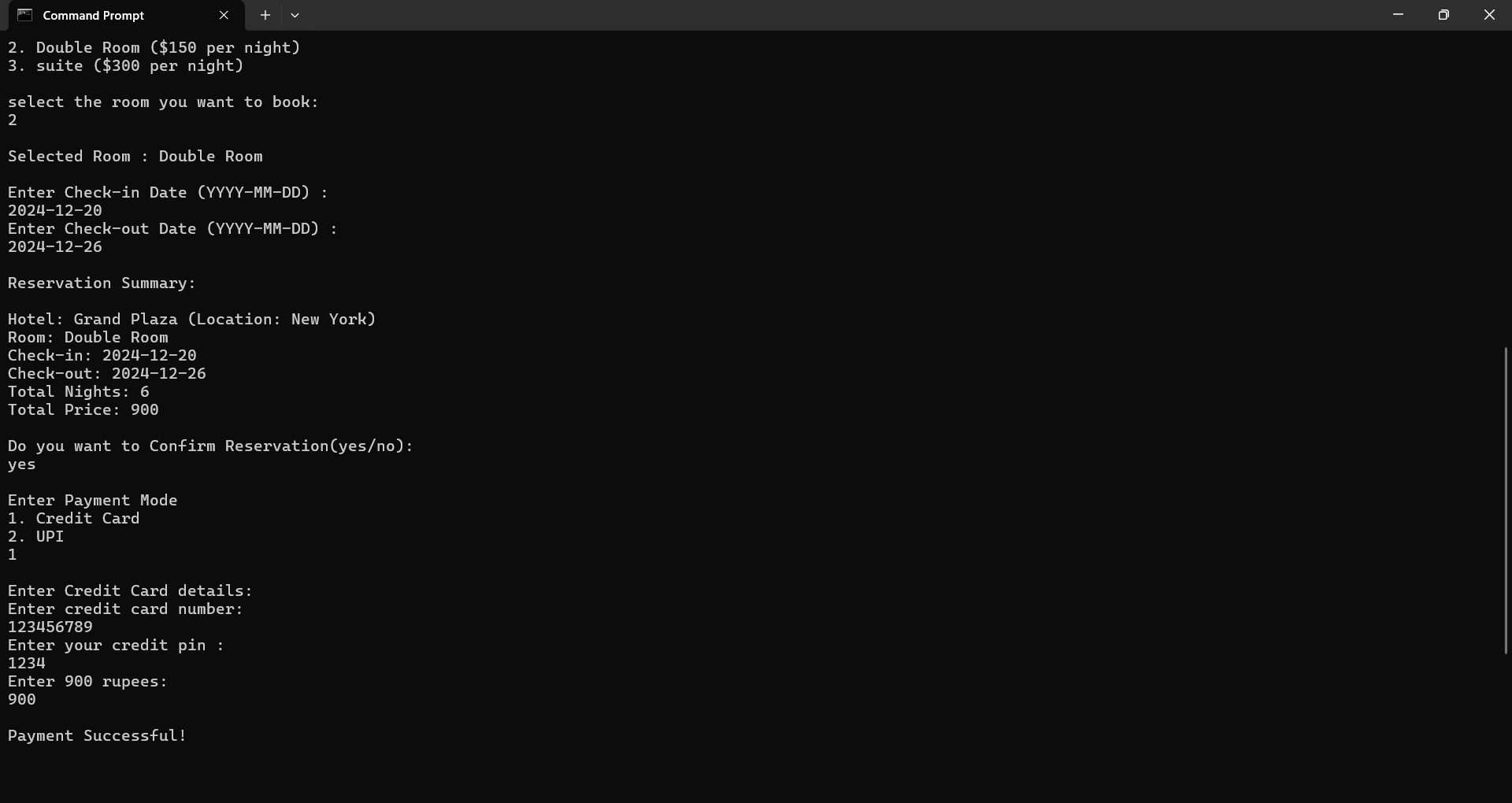
    return noOfDaysBetween;

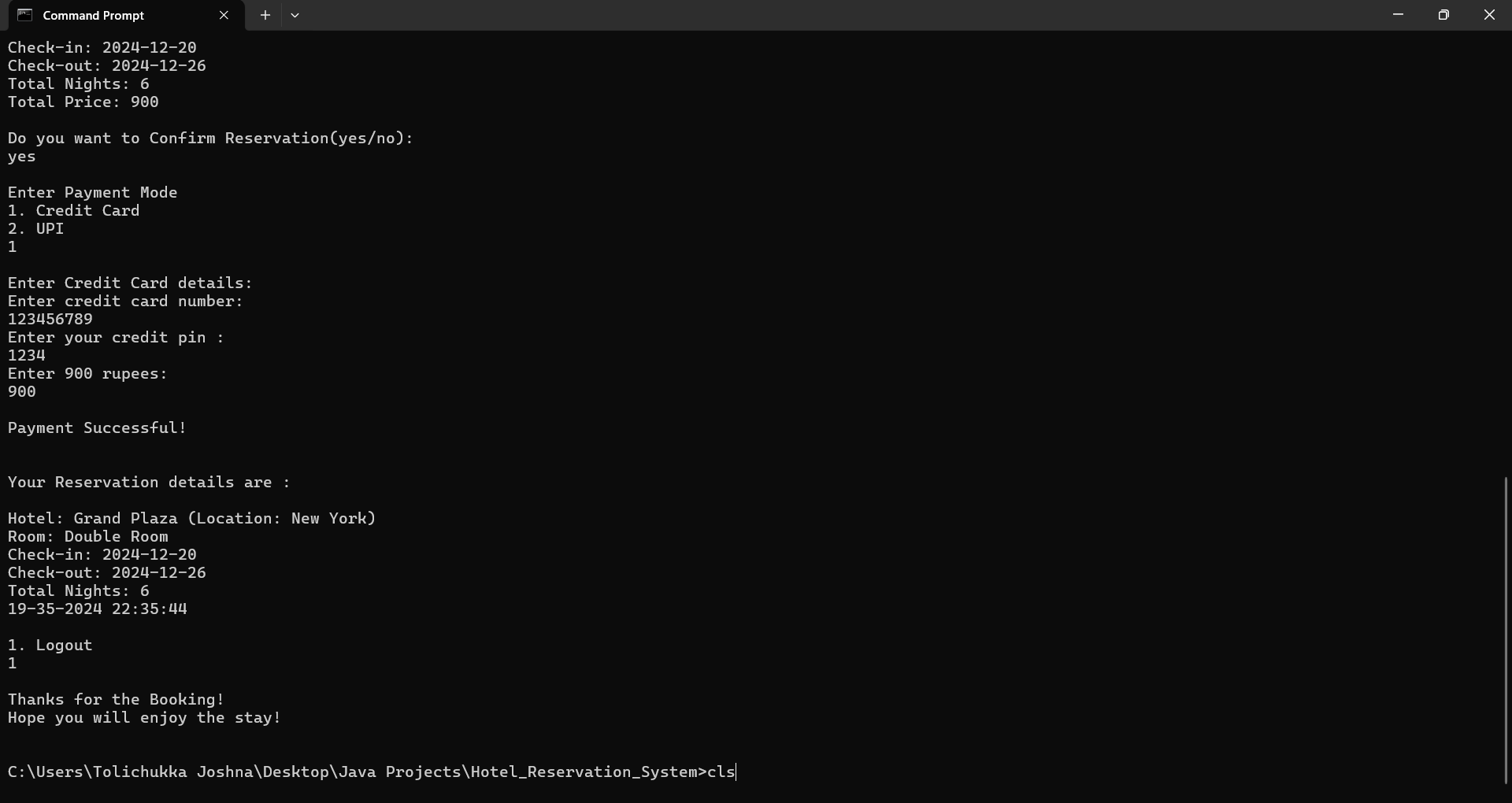
   }

}

**OUTPUT**:-







**Conclusion:-**

##### In conclusion, the **Hotel Reservation System** provides a comprehensive and user-friendly solution for booking hotel rooms, ensuring a seamless process from login to payment. By allowing users to register, browse hotels, select rooms, and make payments securely, the system automates key functions of hotel booking, including calculating the total cost based on stay duration and room type. The integration of date validation, secure OTP-based login, and flexible payment methods such as credit cards and UPI enhances both user experience and security. Ultimately, this system serves as an efficient and practical tool for both customers and hotel management.