UNIVERSITY OF ASIA PACIFIC

DEPARTMENT OF CSE

PROJECT SUBMISSION- SPRING 2024

COURSE CODE   : CSE-202

COURSE TITLE   : OBJECT ORIENTED PROGRAMMING-I LAB

PROJECT NO.   : 01

DATE OF DEMONSTRATION   : 30.12.2024

SUBMITTED BY:	**SUBMITTED TO:**
NAME : TASHNIMUL HASAN SHIFAT	**NAME: BAIVAB DAS**
MOHAIMINUL HAQ ALVI	LECTURER
ADIKA SULTANA	CSE, UAP

STUDENT ID.   : 23101142

23101158

23101162

SEMESTER   : 2nd YEAR 1st SEMESTER

SECTION   : C

GROUP   : C2

Class Diagram:

1.Class PersonAlreadyExistsException
public PersonAlreadyExistsException(String message)

2.Class InvalidDateFormatException
public InvalidDateFormatException(String message)

3. Class DataFileNotFoundException
public DataFileNotFoundException(String message)

4.Class InvalidAppointmentException
public InvalidAppointmentException(String message)

5. Class – (person)
Attribute:
id (String private)
name (String private)
phone (String private)
Constructor
Person(String id, String name, String phone)
Methods
public String getId()
public String getName()
public String getPhone()
public void setId(String id)
public void setName(String name)
public void setPhone(String phone)

6. Class-(patient)
Attribute:
Age(private int)
Address (private String)
Diagnoses (private String)
Constructor
Patient (String id, String name, String phone, int age, String address, String diagnosis)
Methods
public int getAge()
public String getAddress()
public String getDiagnosis()
public void setAge(int age)
public void setAddress(String address)
public void setDiagnosis(String diagnosis)
public String getInfo()

7.Class: Doctor
Attribute:
Specialization(private String)
Availability(private String)

Constructor
Doctor(String id, String name, String phone, String specialization, String availability)

Methods
public String getSpecialization()
public String getAvailability()
public void setSpecialization(String specialization)
public void setAvailability(String availability)
public String getInfo()

8. Class-(Nurse)

Attribute:

Shift *(private String)*

Specialization *(private String)*

Constructor

```java
public Nurse(String id, String name, String phone, String shift, String specialization)
```

Method

```java
public String getInfo()
```

Class9(Appointment) *Attribute:* *appointmentID(private String)* *patientID (private String)* *doctorID(private String)* *Date appointmentDate(private String)* *appointmentTime(private String)*
Constructor *Appointment(String appointmentID, String patientID, String doctorID, String appointmentDateStr, String appointmentTime) throws InvalidDateFormatException*
Methods *public String getAppointmentInfo()*

10. Class- (Billing)

Attribute:

billID(**private** String)
patientID(**private** String)
doctorFee **(private double)**
treatmentCost**(private double)**
totalAmount**(private double)**

Constructor

Billing(String billID, String patientID, **double** doctorFee, **double** treatmentCost)

Methods

public String getBillInfo()

11.Class(Main Class)

HospitalManagementSystem

public static void main(String[] args)

1.

```java
package hospitalManagement;

public class PersonAlreadyExistsException extends Exception {
    public PersonAlreadyExistsException(String message) {
        super(message);
    }
}
```

2.

```java
HospitalManagementSyste...    Patient.java    Doctor.java    Appointment.java    Billing.java    Person.java
1  package hospitalManagement;
2
3  public class PersonAlreadyExistsException extends Exception {
4      public PersonAlreadyExistsException(String message) {
5          super(message);
6      }
7  }
8
```

3.

```java
1 package hospitalManagement;
2
3 public class InvalidDateFormatException extends Exception
4 {
5     public InvalidDateFormatException(String message) {
6         super(message);
7     }
8 }
9
```

4.

```java
1 package hospitalManagement;
2
3 public class InvalidAppointmentException  extends Exception
4 {
5     public InvalidAppointmentException(String message) {
6         super(message);
7     }
8 }
9
10
11
```

5.

```java
1  package hospitalManagement;
2
3  import java.io.Serializable;
4
5
6  public class Person implements Serializable {
7      private String id;
8      private String name;
9      private String phone;
10
11
12      public Person(String id, String name, String phone) {
13          this.id = id;
14          this.name = name;
15          this.phone = phone;
16      }
17
18
19      public String getId() {
20          return id;
21      }
22
23      public String getName() {
24          return name;
25      }
26
27      public String getPhone() {
28          return phone;
29      }
30
31      public void setId(String id) {
32          this.id = id;
33      }
34
35      public void setName(String name) {
36          this.name = name;
37      }
38
39      public void setPhone(String phone) {
40          this.phone = phone;
41      }
42
43
```

```java
6  public class Person implements Serializable {
7      private String id;
8      private String name;
9      private String phone;
10
11
12     public Person(String id, String name, String phone) {
13         this.id = id;
14         this.name = name;
15         this.phone = phone;
16     }
17
18
19     public String getId() {
20         return id;
21     }
22
23     public String getName() {
24         return name;
25     }
26
27     public String getPhone() {
28         return phone;
29     }
30
31     public void setId(String id) {
32         this.id = id;
33     }
34
35     public void setName(String name) {
36         this.name = name;
37     }
38
39     public void setPhone(String phone) {
40         this.phone = phone;
41     }
42
43
44     public String getInfo() {
45         return "ID: " + id + ", Name: " + name + ", Phone: " + phone;
46     }
47 }
48
```

6.

```java
package hospitalManagement;

import java.io.Serializable;


public class Patient extends Person {
    private int age;
    private String address;
    private String diagnosis;

    public Patient(String id, String name, String phone, int age, String address, String diagnosis) {
        super(id, name, phone);
        this.age = age;
        this.address = address;
        this.diagnosis = diagnosis;
    }


    |
    public int getAge() {
        return age;
    }

    public String getAddress() {
        return address;
    }

    public String getDiagnosis() {
        return diagnosis;
    }

    public void setAge(int age) {
        this.age = age;
    }

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

    public void setDiagnosis(String diagnosis) {
        this.diagnosis = diagnosis;
    }

    @Override
```

```java
6 public class Patient extends Person {
7      private int age;
8      private String address;
9      private String diagnosis;
10
11     public Patient(String id, String name, String phone, int age, String address, String diagnosis) {
12         super(id, name, phone);
13         this.age = age;
14         this.address = address;
15         this.diagnosis = diagnosis;
16     }
17
18
19     public int getAge() {
20         return age;
21     }
22
23     public String getAddress() {
24         return address;
25     }
26
27     public String getDiagnosis() {
28         return diagnosis;
29     }
30
31     public void setAge(int age) {
32         this.age = age;
33     }
34
35     public void setAddress(String address) {
36         this.address = address;
37     }
38
39     public void setDiagnosis(String diagnosis) {
40         this.diagnosis = diagnosis;
41     }
42
43     @Override
44     public String getInfo() {
45         return super.getInfo() + ", Age: " + age + ", Address: " + address + ", Diagnosis: " + diagnosis;
46     }
47 }
48
```

7.

```java
1  package hospitalManagement;
2
3  import java.io.Serializable;
4
5
6  public class Doctor extends Person {
7      private String specialization;
8      private String availability;
9
10     public Doctor(String id, String name, String phone, String specialization, String availability) {
11         super(id, name, phone);
12         this.specialization = specialization;
13         this.availability = availability;
14     }
15
16     public String getSpecialization() {
17         return specialization;
18     }
19
20     public String getAvailability() {
21         return availability;
22     }
23
24     public void setSpecialization(String specialization) {
25         this.specialization = specialization;
26     }
27
28     public void setAvailability(String availability) {
29         this.availability = availability;
30     }
31
32     @Override
33     public String getInfo() {
34         return super.getInfo() + ", Specialization: " + specialization + ", Availability: " + availability;
35     }
36 }
37
```

8.

```java
1  package hospitalManagement;     Java Project/src/hospitalManagement/Patient.java
2  import java.io.Serializable;
3
4  public class Nurse extends Person {
5      private String shift;
6      private String specialization;
7
8      public Nurse(String id, String name, String phone, String shift, String specialization) {
9          super(id, name, phone);
10         this.shift = shift;
11         this.specialization = specialization;
12     }
13
14     @Override
15     public String getInfo() {
16         return "ID: " + getId() + ", Name: " + getName() + ", Phone: " + getPhone() +
17                 ", Shift: " + shift + ", Specialization: " + specialization;
18     }
19 }
20
21
```

9.

```java
1  package hospitalManagement;
2
3  import java.io.Serializable;
4  import java.text.ParseException;
5  import java.text.SimpleDateFormat;
6  import java.util.Date;
7
8
9  public class Appointment implements Serializable {
10     private String appointmentID;
11     private String patientID;
12     private String doctorID;
13     private Date appointmentDate;
14     private String appointmentTime;
15
16     public Appointment(String appointmentID, String patientID, String doctorID, String appointmentDateStr, String appointmentTime) throws InvalidDateFormatException {
17         this.appointmentID = appointmentID;
18         this.patientID = patientID;
19         this.doctorID = doctorID;
20
21         // Convert the string date into a Date object
22         SimpleDateFormat sdf = new SimpleDateFormat("dd-MM-yyyy");
23         try {
24             this.appointmentDate = sdf.parse(appointmentDateStr);
25         } catch (ParseException e) {
26             throw new InvalidDateFormatException("Invalid date format! Please use DD-MM-YYYY.");
27         }
28
29         this.appointmentTime = appointmentTime;
30     }
31
32     public String getAppointmentInfo() {
33         return "Appointment ID: " + appointmentID + ", Patient ID: " + patientID + ", Doctor ID: " + doctorID +
34                 ", Date: " + (appointmentDate != null ? appointmentDate.toString() : "Invalid date") + ", Time: " + appointmentTime;
35     }
36  }
37
```

10.

```java
1  package hospitalManagement;
2  import java.io.Serializable;
3
4  public class Billing implements Serializable {
5      private String billID;
6      private String patientID;
7      private double doctorFee;
8      private double treatmentCost;
9      private double totalAmount;
10
11     public Billing(String billID, String patientID, double doctorFee, double treatmentCost) {
12         this.billID = billID;
13         this.patientID = patientID;
14         this.doctorFee = doctorFee;
15         this.treatmentCost = treatmentCost;
16         this.totalAmount = doctorFee + treatmentCost;
17     }
18
19     public String getBillInfo() {
20         return "Bill ID: " + billID + ", Patient ID: " + patientID + ", Doctor Fee: " + doctorFee +
21                 ", Treatment Cost: " + treatmentCost + ", Total Amount: " + totalAmount;
22     }
23 }
24
```

11.

```java
1  package hospitalManagement;
2
3  import java.io.*;
6
7  public class HospitalManagementSystem {
8      private List<Person> peopleList;
9      private List<Appointment> appointmentsList;
10
11     public HospitalManagementSystem() {
12         peopleList = new ArrayList<>();
13         appointmentsList = new ArrayList<>();
14     }
15
16
17     public void addPerson(Person person) throws PersonAlreadyExistsException {
18
19         if (peopleList.stream().anyMatch(p -> p.getId().equals(person.getId()))) {
20             throw new PersonAlreadyExistsException("Person with ID " + person.getId() + " already exists.");
21         } else {
22             peopleList.add(person);
23         }
24     }
25
26
27     public void saveData() throws IOException {
28
29         File peopleFile = new File("people.dat");
30         if (!peopleFile.exists()) {
31             peopleFile.createNewFile();
32         }
33
34         File appointmentsFile = new File("appointments.dat");
35         if (!appointmentsFile.exists()) {
36             appointmentsFile.createNewFile();
37         }
38
39
40         try (ObjectOutputStream peopleOut = new ObjectOutputStream(new FileOutputStream(peopleFile));
41              ObjectOutputStream appointmentsOut = new ObjectOutputStream(new FileOutputStream(appointmentsFile))) {
42             peopleOut.writeObject(peopleList);
43             appointmentsOut.writeObject(appointmentsList);
44         }
45     }
```

```java
39
40        try (ObjectOutputStream peopleOut = new ObjectOutputStream(new FileOutputStream(peopleFile));
41             ObjectOutputStream appointmentsOut = new ObjectOutputStream(new FileOutputStream(appointmentsFile))) {
42            peopleOut.writeObject(peopleList);
43            appointmentsOut.writeObject(appointmentsList);
44        }
45    }
46
47
48    public void displayPeople() {
49        if (peopleList.isEmpty()) {
50            System.out.println("No people to display.");
51            return;
52        }
53        for (Person person : peopleList) {
54            System.out.println(person.getInfo());
55        }
56    }
57
58
59    public void displayAppointments() {
60        if (appointmentsList.isEmpty()) {
61            System.out.println("No appointments to display.");
62            return;
63        }
64        for (Appointment appointment : appointmentsList) {
65            System.out.println(appointment.getAppointmentInfo());
66        }
67    }
68
69
70    public static void main(String[] args) {
71        HospitalManagementSystem hms = new HospitalManagementSystem();
72
73
74        Scanner sc = new Scanner(System.in);
75        boolean exit = false;
76
77        while (!exit) {
78            try {
79                System.out.println("\n----- Hospital Management System -----");
80                System.out.println("1. Add Patient");
81                System.out.println("2. Add Doctor");
```

```java
        while (!exit) {
            try {
                System.out.println("\n----- Hospital Management System -----");
                System.out.println("1. Add Patient");
                System.out.println("2. Add Doctor");
                System.out.println("3. Schedule Appointment");
                System.out.println("4. Display People");
                System.out.println("5. Display Appointments");
                System.out.println("6. Exit");
                System.out.print("Enter your choice: ");
                int choice = sc.nextInt();
                sc.nextLine();

                switch (choice) {
                    case 1:
                        hms.addPatientInput(sc);
                        break;

                    case 2:
                        hms.addDoctorInput(sc);
                        break;

                    case 3:
                        hms.scheduleAppointmentInput(sc);
                        break;

                    case 4:
                        hms.displayPeople();
                        break;

                    case 5:
                        hms.displayAppointments();
                        break;

                    case 6:
                        try {
                            hms.saveData();
                        } catch (IOException e) {
                            System.err.println("Error saving data: " + e.getMessage());
                        }
                        exit = true;
                        System.out.println("Exiting the system. Goodbye!");
```

```java
111                 case 6:
112                     try {
113                         hms.saveData();
114                     } catch (IOException e) {
115                         System.err.println("Error saving data: " + e.getMessage());
116                     }
117                     exit = true;
118                     System.out.println("Exiting the system. Goodbye!");
119                     break;
120
121                 default:
122                     System.out.println("Invalid choice. Please try again.");
123                     break;
124             }
125         } catch (InputMismatchException e) {
126             System.err.println("Invalid input! Please enter valid data.");
127             sc.nextLine();
128         } catch (Exception e) {
129             System.err.println("Error: " + e.getMessage());
130         }
131     }
132
133     sc.close();
134 }
135
136
137⊖ private void addPatientInput(Scanner sc) {
138     System.out.println("Enter Patient ID: ");
139     String patientId = sc.nextLine();
140     System.out.println("Enter Patient Name: ");
141     String patientName = sc.nextLine();
142     System.out.println("Enter Patient Phone: ");
143     String patientPhone = sc.nextLine();
144     System.out.println("Enter Patient Age: ");
145     int patientAge = sc.nextInt();
146     sc.nextLine();
147     System.out.println("Enter Patient Address: ");
148     String patientAddress = sc.nextLine();
149     System.out.println("Enter Patient Diagnosis: ");
150     String patientDiagnosis = sc.nextLine();
151
152     Patient patient = new Patient(patientId, patientName, patientPhone, patientAge, patientAddress, patientDiagnosis);
153     try {
```

```java
 41            String patientName = sc.nextLine();
 42            System.out.println("Enter Patient Phone: ");
 43            String patientPhone = sc.nextLine();
 44            System.out.println("Enter Patient Age: ");
 45            int patientAge = sc.nextInt();
 46            sc.nextLine();
 47            System.out.println("Enter Patient Address: ");
 48            String patientAddress = sc.nextLine();
 49            System.out.println("Enter Patient Diagnosis: ");
 50            String patientDiagnosis = sc.nextLine();
 51
 52            Patient patient = new Patient(patientId, patientName, patientPhone, patientAge, patientAddress, patientDiagnosis);
 53            try {
 54                addPerson(patient);
 55            } catch (PersonAlreadyExistsException e) {
 56                System.err.println(e.getMessage());
 57            }
 58        }
 59
 60
 61⊖    private void addDoctorInput(Scanner sc) {
 62            System.out.println("Enter Doctor ID: ");
 63            String doctorId = sc.nextLine();
 64            System.out.println("Enter Doctor Name: ");
 65            String doctorName = sc.nextLine();
 66            System.out.println("Enter Doctor Phone: ");
 67            String doctorPhone = sc.nextLine();
 68            System.out.println("Enter Doctor Specialization: ");
 69            String doctorSpecialization = sc.nextLine();
 70            System.out.println("Enter Doctor Availability: ");
 71            String doctorAvailability = sc.nextLine();
 72
 73            Doctor doctor = new Doctor(doctorId, doctorName, doctorPhone, doctorSpecialization, doctorAvailability);
 74            try {
 75                addPerson(doctor);
 76            } catch (PersonAlreadyExistsException e) {
 77                System.err.println(e.getMessage());
 78            }
 79        }
 80
 81
 82⊖    private void scheduleAppointmentInput(Scanner sc) {
 83            System.out.println("Enter Appointment ID: ");
```

```java
160
161⊖     private void addDoctorInput(Scanner sc) {
162         System.out.println("Enter Doctor ID: ");
163         String doctorId = sc.nextLine();
164         System.out.println("Enter Doctor Name: ");
165         String doctorName = sc.nextLine();
166         System.out.println("Enter Doctor Phone: ");
167         String doctorPhone = sc.nextLine();
168         System.out.println("Enter Doctor Specialization: ");
169         String doctorSpecialization = sc.nextLine();
170         System.out.println("Enter Doctor Availability: ");
171         String doctorAvailability = sc.nextLine();
172
173         Doctor doctor = new Doctor(doctorId, doctorName, doctorPhone, doctorSpecialization, doctorAvailability);
174         try {
175             addPerson(doctor);
176         } catch (PersonAlreadyExistsException e) {
177             System.err.println(e.getMessage());
178         }
179     }
180
181
182⊖     private void scheduleAppointmentInput(Scanner sc) {
183         System.out.println("Enter Appointment ID: ");
184         String appointmentId = sc.nextLine();
185         System.out.println("Enter Patient ID for Appointment: ");
186         String appointmentPatientId = sc.nextLine();
187         System.out.println("Enter Doctor ID for Appointment: ");
188         String appointmentDoctorId = sc.nextLine();
189         System.out.println("Enter Appointment Date (DD-MM-YYYY): ");
190         String appointmentDate = sc.nextLine();
191         System.out.println("Enter Appointment Time: ");
192         String appointmentTime = sc.nextLine();
193
194         try {
195             Appointment appointment = new Appointment(appointmentId, appointmentPatientId, appointmentDoctorId, appointmentDate, appointmentTime);
196             appointmentsList.add(appointment);
197         } catch (Exception e) {
198             System.err.println("Error scheduling appointment: " + e.getMessage());
199         }
200     }
201 }
202
```

OUTPUT:

```
Console ×
HospitalManagementSystem [Java Application] C:\Program Files\Java\jdk-23\bin\javaw.exe  (Jan 1, 2025, 2:29:55 PM) [pid: 18360]

----- Hospital Management System -----
1. Add Patient
2. Add Doctor
3. Schedule Appointment
4. Display People
5. Display Appointments
6. Add Billing
7. Display Billing
8. Add Nurse
9. Display Nurses
10. Exit
Enter your choice: 1
Enter Patient ID:
162
Enter Patient Name:
adika
Enter Patient Phone:
0181
Enter Patient Age:
15
Enter Patient Address:
dilu
Enter Patient Diagnosis:
eye

----- Hospital Management System -----
1. Add Patient
2. Add Doctor
3. Schedule Appointment
4. Display People
5. Display Appointments
6. Add Billing
7. Display Billing
8. Add Nurse
9. Display Nurses
10. Exit
```

HospitalManagementSystem [Java Application] C:\Program Files\Java\jdk-23\bin\javaw.exe (Jan 1, 2025, 2:29:55 PM) [pid: 18

```
10. Exit
Enter your choice: 1
Enter Patient ID:
142
Enter Patient Name:
shifat
Enter Patient Phone:
45621378
Enter Patient Age:
14
Enter Patient Address:
farmgate
Enter Patient Diagnosis:
fever

----- Hospital Management System -----
1. Add Patient
2. Add Doctor
3. Schedule Appointment
4. Display People
5. Display Appointments
6. Add Billing
7. Display Billing
8. Add Nurse
9. Display Nurses
10. Exit
Enter your choice: 2
Enter Doctor ID:
158
Enter Doctor Name:
alvi
Enter Doctor Phone:
789456
Enter Doctor Specialization:
chest
Enter Doctor Availability:
yes
```

yes

----- Hospital Management System -----
1. Add Patient
2. Add Doctor
3. Schedule Appointment
4. Display People
5. Display Appointments
6. Add Billing
7. Display Billing
8. Add Nurse
9. Display Nurses
10. Exit
Enter your choice: 2
Enter Doctor ID:
160
Enter Doctor Name:
limon
Enter Doctor Phone:
159753
Enter Doctor Specialization:
leg
Enter Doctor Availability:
no

----- Hospital Management System -----
1. Add Patient
2. Add Doctor
3. Schedule Appointment
4. Display People
5. Display Appointments
6. Add Billing
7. Display Billing
8. Add Nurse
9. Display Nurses
10. Exit
Enter your choice: 3

HospitalManagementSystem [Java Application] C:\Program Files\Java\jdk-23\bin\javaw.exe (Jan 1, 2025, 2:29:55 PM) [pid: 18360]

```
Enter your choice: 3
Enter Appointment ID:
162
Enter Patient ID for Appointment:
142
Enter Doctor ID for Appointment:
158
Enter Appointment Date (DD-MM-YYYY):
12-12-24
Enter Appointment Time:
8

----- Hospital Management System -----
1. Add Patient
2. Add Doctor
3. Schedule Appointment
4. Display People
5. Display Appointments
6. Add Billing
7. Display Billing
8. Add Nurse
9. Display Nurses
10. Exit
Enter your choice: 4
ID: 162, Name: adika, Phone: 0181, Age: 15, Address: dilu, Diagnosis: eye
ID: 142, Name: shifat, Phone: 45621378, Age: 14, Address: farmgate, Diagnosis: fever

----- Hospital Management System -----
1. Add Patient
2. Add Doctor
3. Schedule Appointment
4. Display People
5. Display Appointments
6. Add Billing
7. Display Billing
8. Add Nurse
9. Display Nurses
```

HospitalManagementSystem [Java Application] C:\Program Files\Java\jdk-23\bin\javaw.exe (Jan 1, 2025, 2:29:55 PM) [pid: 18360]

```
9. Display Nurses
10. Exit
Enter your choice: 5
Appointment ID: 162, Patient ID: 142, Doctor ID: 158, Date: Tue Dec 12 00:00:00 GMT+06:00 24, Time: 8

----- Hospital Management System -----
1. Add Patient
2. Add Doctor
3. Schedule Appointment
4. Display People
5. Display Appointments
6. Add Billing
7. Display Billing
8. Add Nurse
9. Display Nurses
10. Exit
Enter your choice: 6
Enter Bill ID:
157
Enter Patient ID:
162
Enter Doctor Fee:
1500
Enter Treatment Cost:
75468
Billing record added successfully.

----- Hospital Management System -----
1. Add Patient
2. Add Doctor
3. Schedule Appointment
4. Display People
5. Display Appointments
6. Add Billing
7. Display Billing
8. Add Nurse
9. Display Nurses
```

HospitalManagementSystem [Java Application] C:\Program Files\Java\jdk-23\bin\javaw.exe (Jan 1, 2025, 2:29:55 PM) [pid: 18360]

```
9. Display Nurses
10. Exit
Enter your choice: 7
Bill ID: 157, Patient ID: 162, Doctor Fee: 1500.0, Treatment Cost: 75468.0, Total Amount: 76968.0

----- Hospital Management System -----
1. Add Patient
2. Add Doctor
3. Schedule Appointment
4. Display People
5. Display Appointments
6. Add Billing
7. Display Billing
8. Add Nurse
9. Display Nurses
10. Exit
Enter your choice: 8
Enter Nurse ID:
478
Enter Nurse Name:
dalila
Enter Nurse Phone:
478555555
Enter Nurse Shift:
day
Enter Nurse Specialization:
ear

----- Hospital Management System -----
1. Add Patient
2. Add Doctor
3. Schedule Appointment
4. Display People
5. Display Appointments
6. Add Billing
7. Display Billing
8. Add Nurse
```

HospitalManagementSystem [Java Application] C:\Program Files\Java\jdk-23\bin\javaw.exe (Jan 1, 2025, 2:29:55

```
478555555
Enter Nurse Shift:
day
Enter Nurse Specialization:
ear

----- Hospital Management System -----
1. Add Patient
2. Add Doctor
3. Schedule Appointment
4. Display People
5. Display Appointments
6. Add Billing
7. Display Billing
8. Add Nurse
9. Display Nurses
10. Exit
Enter your choice: 9
ID: 478, Name: dalila, Phone: 478555555, Shift: day, Specialization: ear

----- Hospital Management System -----
1. Add Patient
2. Add Doctor
3. Schedule Appointment
4. Display People
5. Display Appointments
6. Add Billing
7. Display Billing
8. Add Nurse
9. Display Nurses
10. Exit
Enter your choice: 1
Enter Patient ID:
162
Enter Patient Name:
adika
Enter Patient Phone:
```

4. Display People
5. Display Appointments
6. Add Billing
7. Display Billing
8. Add Nurse
9. Display Nurses
10. Exit
Enter your choice: 9
ID: 478, Name: dalila, Phone: 478555555, Shift: day, Specialization: ear

----- Hospital Management System -----
1. Add Patient
2. Add Doctor
3. Schedule Appointment
4. Display People
5. Display Appointments
6. Add Billing
7. Display Billing
8. Add Nurse
9. Display Nurses
10. Exit
Enter your choice: 1
Enter Patient ID:
162
Enter Patient Name:
adika
Enter Patient Phone:
558555
Enter Patient Age:
78
Enter Patient Address:
asfghjkkl
Enter Patient Diagnosis:
eye
Person with ID 162 already exists.

----- Hospital Management System -----

THANKS SIR

THE END