```
import java.util.ArrayList;
import java.util.Scanner;

// Patient class
class Patient {
    int id;
    String name, gender, disease;
    int age;

    Patient(int id, String name, int age, String gender, String disease) {
        this.id = id;
        this.name = name;
        this.age = age;
        this.gender = gender;
        this.disease = disease;
    }

    public String toString() {
        return id + " | " + name + " | " + age + " | " + gender + " | " + disease;
    }
}

// Appointment class
class Appointment {
    int id, patientId;
    String doctor, date;
```

```
    Appointment(int id, int patientId, String doctor, String date) {
        this.id = id;
        this.patientId = patientId;
        this.doctor = doctor;
        this.date = date;
    }

    public String toString() {
        return id + " | Patient ID: " + patientId + " | Doctor: " + doctor + " | Date: " + date;
    }
}

// Billing class
class Billing {
    int billId, patientId;
    double amount;

    Billing(int billId, int patientId, double amount) {
        this.billId = billId;
        this.patientId = patientId;
        this.amount = amount;
    }

    public String toString() {
        return "Bill ID: " + billId + " | Patient ID: " + patientId + " | Amount: $" + amount;
```

```
    }
}

// Inventory class
class Inventory {
    int id, quantity;
    String itemName;

    Inventory(int id, String itemName, int quantity) {
        this.id = id;
        this.itemName = itemName;
        this.quantity = quantity;
    }

    public String toString() {
        return id + " | " + itemName + " | Quantity: " + quantity;
    }
}

// Staff class
class Staff {
    int id;
    String name, role;
    double salary;

    Staff(int id, String name, String role, double salary) {
```

```
        this.id = id;
        this.name = name;
        this.role = role;
        this.salary = salary;
    }

    public String toString() {
        return id + " | " + name + " | " + role + " | $" + salary;
    }
}

// Main Hospital Management System
public class HospitalManagementSystem {
    static ArrayList<Patient> patients = new ArrayList<>();
    static ArrayList<Appointment> appointments = new ArrayList<>();
    static ArrayList<Billing> bills = new ArrayList<>();
    static ArrayList<Inventory> inventoryList = new ArrayList<>();
    static ArrayList<Staff> staffList = new ArrayList<>();

    static Scanner scanner = new Scanner(System.in);
    static int patientIdCounter = 1, appointmentIdCounter = 1, billIdCounter = 1,
inventoryIdCounter = 1, staffIdCounter = 1;

    public static void main(String[] args) {
        while (true) {
            System.out.println("\nHospital Management System");
```

```
System.out.println("1. Add Patient");
System.out.println("2. View Patients");
System.out.println("3. Add Appointment");
System.out.println("4. View Appointments");
System.out.println("5. Add Bill");
System.out.println("6. View Bills");
System.out.println("7. Add Inventory Item");
System.out.println("8. View Inventory");
System.out.println("9. Add Staff");
System.out.println("10. View Staff");
System.out.println("11. Exit");
System.out.print("Enter choice: ");
int choice = scanner.nextInt();

switch (choice) {
   case 1 -> addPatient();
   case 2 -> viewPatients();
   case 3 -> addAppointment();
   case 4 -> viewAppointments();
   case 5 -> addBill();
   case 6 -> viewBills();
   case 7 -> addInventory();
   case 8 -> viewInventory();
   case 9 -> addStaff();
   case 10 -> viewStaff();
   case 11 -> {
```

```
            System.out.println("Exiting...");
            return;
        }
        default -> System.out.println("Invalid choice! Try again.");
      }
    }
}

public static void addPatient() {
    scanner.nextLine();
    System.out.print("Enter Patient Name: ");
    String name = scanner.nextLine();
    System.out.print("Enter Age: ");
    int age = scanner.nextInt();
    scanner.nextLine();
    System.out.print("Enter Gender: ");
    String gender = scanner.nextLine();
    System.out.print("Enter Disease: ");
    String disease = scanner.nextLine();

    patients.add(new Patient(patientIdCounter++, name, age, gender, disease));
    System.out.println("Patient added successfully!");
}

public static void viewPatients() {
    if (patients.isEmpty()) {
```

```
            System.out.println("No patients found.");
            return;
        }
        System.out.println("\nPatient List:");
        for (Patient p : patients) {
            System.out.println(p);
        }
    }

    public static void addAppointment() {
        System.out.print("Enter Patient ID: ");
        int patientId = scanner.nextInt();
        scanner.nextLine();
        System.out.print("Enter Doctor's Name: ");
        String doctor = scanner.nextLine();
        System.out.print("Enter Appointment Date (YYYY-MM-DD): ");
        String date = scanner.nextLine();

        appointments.add(new Appointment(appointmentIdCounter++, patientId, doctor,
date));
        System.out.println("Appointment added successfully!");
    }

    public static void viewAppointments() {
        if (appointments.isEmpty()) {
            System.out.println("No appointments found.");
```

```
        return;
    }
    System.out.println("\nAppointment List:");
    for (Appointment a : appointments) {
        System.out.println(a);
    }
}

public static void addBill() {
    System.out.print("Enter Patient ID: ");
    int patientId = scanner.nextInt();
    System.out.print("Enter Bill Amount: ");
    double amount = scanner.nextDouble();

    bills.add(new Billing(billIdCounter++, patientId, amount));
    System.out.println("Bill added successfully!");
}

public static void viewBills() {
    if (bills.isEmpty()) {
        System.out.println("No bills found.");
        return;
    }
    System.out.println("\nBilling Records:");
    for (Billing b : bills) {
        System.out.println(b);
```

```
    }
}

public static void addInventory() {
    scanner.nextLine();
    System.out.print("Enter Item Name: ");
    String itemName = scanner.nextLine();
    System.out.print("Enter Quantity: ");
    int quantity = scanner.nextInt();

    inventoryList.add(new Inventory(inventoryIdCounter++, itemName, quantity));
    System.out.println("Inventory item added successfully!");
}

public static void viewInventory() {
    if (inventoryList.isEmpty()) {
        System.out.println("No inventory found.");
        return;
    }
    System.out.println("\nInventory List:");
    for (Inventory i : inventoryList) {
        System.out.println(i);
    }
}

public static void addStaff() {
```

```
        scanner.nextLine();
        System.out.print("Enter Staff Name: ");
        String name = scanner.nextLine();
        System.out.print("Enter Role: ");
        String role = scanner.nextLine();
        System.out.print("Enter Salary: ");
        double salary = scanner.nextDouble();

        staffList.add(new Staff(staffIdCounter++, name, role, salary));
        System.out.println("Staff added successfully!");
    }

    public static void viewStaff() {
        if (staffList.isEmpty()) {
            System.out.println("No staff found.");
            return;
        }
        System.out.println("\nStaff List:");
        for (Staff s : staffList) {
            System.out.println(s);
        }
    }
}
```