RYAN INTERNATIONAL SCHOOL KUNDALAHALLI, BANGALORE

COMPUTER SCIENCE PROJECT 2021-2022

TOPIC: HOSPITAL MANAGEMENT STANDARD X

PREPAERED BY:

NEYSA MARY PRAMOD
CLASS X-E

SUBMITTED TO:
MS. PAMELA MAITI

ACKNOWLEDGEMENT

We would like to thank our principal Mrs. Vidya Guruprasad, Indian Board of Secondary Education, Supervisor Mrs. Archana and our computer teacher Ms. Pamela for giving us an opportunity and support to complete this project.

We would also like to thank our parents and friends who helped us to complete this project with their creative ideas.

Once again we would like thank our principal, supervisor, Ms. Pamela and last but not least our parents and friends.

INDEX

PROGRAM STATEMENT

This program is designed to develop a **Hospital Management System using Java** that helps in maintaining records of doctors and patients efficiently. The system allows the user to **view the updated list of doctors**, **check the list of existing patients**, and **admit new patients into the hospital**. It aims to reduce manual record keeping and provide a simple computerized solution for managing hospital data systematically.

Source Code

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

public class HospitalSystem {

   // ====================== INNER CLASSES ======================

   // ---- Doctor ----
   static class Doctor {
      String id;
      String name;
      String specialization;
      String timing;
      String qualification;
      int roomNo;

      Doctor(String id, String name, String specialization,
            String timing, String qualification, int roomNo) {
         this.id = id;
         this.name = name;
         this.specialization = specialization;
         this.timing = timing;
         this.qualification = qualification;
         this.roomNo = roomNo;
      }

      void display() {
         System.out.printf("%-5s %-15s %-18s %-12s %-12s %-5d%n",
               id, name, specialization, timing, qualification, roomNo);
```

```
    }
}

// ---- Patient ----
static class Patient {
    String id;
    String name;
    String disease;
    String gender;
    int age;
    String admitted; // Y / N

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

    void display() {
        System.out.printf("%-5s %-15s %-15s %-8s %-5d %-3s%n",
                id, name, disease, gender, age, admitted);
    }
}

// ---- Lab Test / Facility with cost ----
static class LabTest {
    String testName;
    int charge;
```

```
    LabTest(String testName, int charge) {
        this.testName = testName;
        this.charge = charge;
    }

    void display() {
        System.out.printf("%-20s %-5d%n", testName, charge);
    }
}

// ---- General Hospital Service / Facility (no cost) ----
static class Service {
    String description;

    Service(String description) {
        this.description = description;
    }

    void display() {
        System.out.println(description);
    }
}

// ---- Staff Member ----
static class StaffMember {
    String id;
    String name;
    String role;
    String gender;
    int salary;

    StaffMember(String id, String name, String role,
                String gender, int salary) {
```

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

    void display() {
        System.out.printf("%-5s %-15s %-15s %-8s %-6d%n",
                id, name, role, gender, salary);
    }
}

// ==================== DATA LISTS ====================

static Scanner sc = new Scanner(System.in);

static ArrayList<Doctor> doctors = new ArrayList<>();
static ArrayList<Patient> patients = new ArrayList<>();
static ArrayList<LabTest> labTests = new ArrayList<>();
static ArrayList<Service> services = new ArrayList<>();
static ArrayList<StaffMember> staffMembers = new ArrayList<>();

// ==================== HELPER: SAFE INT INPUT
====================

static int readInt(String message) {
    while (true) {
        System.out.print(message);
        try {
            return Integer.parseInt(sc.nextLine());
        } catch (NumberFormatException e) {
            System.out.println("Please enter a valid number.");
```

```
        }
      }
    }

    // ==================== SEED SAMPLE DATA ====================

    static void loadSampleData() {
      // Doctors
      doctors.add(new Doctor("D01", "Liam", "Pediatrician", "9-11 AM",
"MBBS", 101));
      doctors.add(new Doctor("D02", "Eren", "Surgeon", "2-5 PM", "MS",
102));

      // Patients
      patients.add(new Patient("P01", "Louis", "Cardiac Arrest", "Male", 29,
"Y"));
      patients.add(new Patient("P02", "Mikasa", "Fever", "Female", 19, "Y"));

      // Lab tests
      labTests.add(new LabTest("Blood Test", 500));
      labTests.add(new LabTest("X-Ray", 1200));

      // Services
      services.add(new Service("Emergency Room"));
      services.add(new Service("24x7 Ambulance"));
      services.add(new Service("Pharmacy"));

      // Staff
      staffMembers.add(new StaffMember("S01", "James", "Attender",
"Male", 6000));
      staffMembers.add(new StaffMember("S02", "Eleanor", "Nurse",
"Female", 22000));
    }
```

```
// ==================== DOCTOR MENU ====================

static void doctorMenu() {
   int choice;
   do {
      System.out.println("\n---- DOCTOR MENU ----");
      System.out.println("1. Add new doctor");
      System.out.println("2. View all doctors");
      System.out.println("0. Back to main menu");
      choice = readInt("Enter your choice: ");

      switch (choice) {
         case 1:
            System.out.print("Enter Doctor ID: ");
            String id = sc.nextLine();
            System.out.print("Enter Name: ");
            String name = sc.nextLine();
            System.out.print("Enter Specialization: ");
            String spec = sc.nextLine();
            System.out.print("Enter Timing (e.g. 9-11 AM): ");
            String time = sc.nextLine();
            System.out.print("Enter Qualification: ");
            String qual = sc.nextLine();
            int room = readInt("Enter Room No.: ");

            doctors.add(new Doctor(id, name, spec, time, qual, room));
            System.out.println("Doctor added successfully.");
            break;

         case 2:
            if (doctors.isEmpty()) {
               System.out.println("No doctors found.");
```

```
            } else {
                System.out.println("\nID    Name          Specialization    Timing
Qual       Room");
                System.out.println("-----------------------------------------------------
-------------");
                for (Doctor d : doctors) {
                    d.display();
                }
            }
            break;

        case 0:
            // back
            break;

        default:
            System.out.println("Invalid choice.");
      }
    } while (choice != 0);
  }

  // ==================== PATIENT MENU ====================

  static void patientMenu() {
    int choice;
    do {
      System.out.println("\n---- PATIENT MENU ----");
      System.out.println("1. Admit new patient");
      System.out.println("2. View all patients");
      System.out.println("0. Back to main menu");
      choice = readInt("Enter your choice: ");

      switch (choice) {
```

```
            case 1:
               System.out.print("Enter Patient ID: ");
               String id = sc.nextLine();
               System.out.print("Enter Name: ");
               String name = sc.nextLine();
               System.out.print("Enter Disease: ");
               String dis = sc.nextLine();
               System.out.print("Enter Gender: ");
               String gen = sc.nextLine();
               int age = readInt("Enter Age: ");
               System.out.print("Admitted? (Y/N): ");
               String adm = sc.nextLine();

               patients.add(new Patient(id, name, dis, gen, age, adm));
               System.out.println("Patient admitted successfully.");
               break;

            case 2:
               if (patients.isEmpty()) {
                  System.out.println("No patients found.");
               } else {
                  System.out.println("\nID    Name          Disease       Gender
Age  Ad");
                  System.out.println("----------------------------------------------------
");
                  for (Patient p : patients) {
                     p.display();
                  }
               }
               break;

            case 0:
               // back
```

```
            break;

        default:
            System.out.println("Invalid choice.");
      }
   } while (choice != 0);
}

// ===================== LAB MENU ====================

static void labMenu() {
   int choice;
   do {
      System.out.println("\n---- LABORATORY MENU ----");
      System.out.println("1. Add new lab test");
      System.out.println("2. View all lab tests");
      System.out.println("0. Back to main menu");
      choice = readInt("Enter your choice: ");

      switch (choice) {
         case 1:
            System.out.print("Enter Test Name: ");
            String tname = sc.nextLine();
            int cost = readInt("Enter Test Charge: ");
            labTests.add(new LabTest(tname, cost));
            System.out.println("Lab test added successfully.");
            break;

         case 2:
            if (labTests.isEmpty()) {
               System.out.println("No lab tests found.");
            } else {
               System.out.println("\nTest Name          Charge");
```

```
                System.out.println("-------------------------");
                for (LabTest lt : labTests) {
                    lt.display();
                }
            }
            break;

        case 0:
            // back
            break;

        default:
            System.out.println("Invalid choice.");
      }
   } while (choice != 0);
}

// ==================== SERVICE MENU ====================

static void serviceMenu() {
   int choice;
   do {
      System.out.println("\n---- HOSPITAL FACILITY MENU ----");
      System.out.println("1. Add new facility/service");
      System.out.println("2. View all facilities");
      System.out.println("0. Back to main menu");
      choice = readInt("Enter your choice: ");

      switch (choice) {
         case 1:
            System.out.print("Enter Facility Description: ");
            String desc = sc.nextLine();
            services.add(new Service(desc));
```

```
            System.out.println("Facility added successfully.");
            break;

        case 2:
            if (services.isEmpty()) {
                System.out.println("No facilities found.");
            } else {
                System.out.println("\nHospital Facilities:");
                System.out.println("--------------------");
                for (Service ser : services) {
                    ser.display();
                }
            }
            break;

        case 0:
            // back
            break;

        default:
            System.out.println("Invalid choice.");
      }
   } while (choice != 0);
}

// ====================== STAFF MENU ======================

static void staffMenu() {
   int choice;
   do {
      System.out.println("\n---- STAFF MENU ----");
      System.out.println("1. Add new staff member");
      System.out.println("2. View all staff members");
```

```
        System.out.println("0. Back to main menu");
        choice = readInt("Enter your choice: ");

        switch (choice) {
            case 1:
                System.out.print("Enter Staff ID: ");
                String id = sc.nextLine();
                System.out.print("Enter Name: ");
                String name = sc.nextLine();
                System.out.print("Enter Role (Nurse, Attender, etc.): ");
                String role = sc.nextLine();
                System.out.print("Enter Gender: ");
                String gender = sc.nextLine();
                int salary = readInt("Enter Salary: ");
                staffMembers.add(new StaffMember(id, name, role, gender,
salary));
                System.out.println("Staff member added successfully.");
                break;

            case 2:
                if (staffMembers.isEmpty()) {
                    System.out.println("No staff records found.");
                } else {
                    System.out.println("\nID    Name          Role          Gender
Salary");
                    System.out.println("-----------------------------------------------");
                    for (StaffMember sm : staffMembers) {
                        sm.display();
                    }
                }
                break;

            case 0:
```

```
                // back
                break;

            default:
                System.out.println("Invalid choice.");
        }
    } while (choice != 0);
}

// ==================== MAIN ====================

public static void main(String[] args) {

    loadSampleData(); // pre-fill some records


System.out.println("=============================================
");
    System.out.println("       HOSPITAL MANAGEMENT SYSTEM");

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

    int choice;
    do {
        System.out.println("\n******** MAIN MENU ********");
        System.out.println("1. Doctor Section");
        System.out.println("2. Patient Section");
        System.out.println("3. Laboratory Section");
        System.out.println("4. Hospital Facilities Section");
        System.out.println("5. Staff Section");
        System.out.println("0. Exit");
        choice = readInt("Enter your choice: ");
```

```
            switch (choice) {
                case 1:
                    doctorMenu();
                    break;
                case 2:
                    patientMenu();
                    break;
                case 3:
                    labMenu();
                    break;
                case 4:
                    serviceMenu();
                    break;
                case 5:
                    staffMenu();
                    break;
                case 0:
                    System.out.println("\nThank you for using Hospital Management
System.");
                    break;
                default:
                    System.out.println("Invalid choice. Try again.");
            }
        } while (choice != 0);

        sc.close();
    }
}
```

Output

```
==============================================
        HOSPITAL MANAGEMENT SYSTEM
==============================================

******** MAIN MENU ********
1. Doctor Section
2. Patient Section
3. Laboratory Section
4. Hospital Facilities Section
5. Staff Section
0. Exit
Enter your choice: 2

---- PATIENT MENU ----
1. Admit new patient
2. View all patients
0. Back to main menu
Enter your choice: 1
Enter Patient ID: 2345
Enter Name: XYZ
Enter Disease: Corona
Enter Gender: Female
Enter Age: 20
Admitted? (Y/N): Yes
Patient admitted successfully.
```