**JAVA MINI PROJECT CODE:**

**CODE FOR HOSPITAL MANAGEMENT**

import java.util.\*;

class Patient {

int patientId;

String name;

int age;

String disease;

public Patient(int patientId, String name, int age, String disease) {

this.patientId = patientId;

this.name = name;

this.age = age;

this.disease = disease;

}

}

class Doctor {

int doctorId;

String name;

String specialty;

public Doctor(int doctorId, String name, String specialty) {

this.doctorId = doctorId;

this.name = name;

this.specialty = specialty;

}

}

public class HospitalManagementSystem {

// Lists to store patients and doctors

private static List<Patient> patients = new ArrayList<>();

private static List<Doctor> doctors = new ArrayList<>();

private static Scanner scanner = new Scanner(System.in);

// Add a new patient

public static void addPatient() {

System.out.print("Enter Patient ID: ");

int patientId = scanner.nextInt();

scanner.nextLine(); // Consume the newline

System.out.print("Enter Patient Name: ");

String name = scanner.nextLine();

System.out.print("Enter Patient Age: ");

int age = scanner.nextInt();

scanner.nextLine(); // Consume the newline

System.out.print("Enter Disease: ");

String disease = scanner.nextLine();

Patient newPatient = new Patient(patientId, name, age, disease);

patients.add(newPatient);

System.out.println("Patient added successfully.");

}

// View all patients

public static void viewPatients() {

if (patients.isEmpty()) {

System.out.println("No patients found.");

} else {

System.out.println("\nPatient ID\tName\tAge\tDisease");

for (Patient patient : patients) {

System.out.println(patient.patientId + "\t" + patient.name + "\t" + patient.age + "\t" + patient.disease);

}

}

}

// Search for a patient by ID

public static void searchPatientById() {

System.out.print("Enter Patient ID to search: ");

int patientId = scanner.nextInt();

boolean found = false;

for (Patient patient : patients) {

if (patient.patientId == patientId) {

System.out.println("Patient Found: ");

System.out.println("Patient ID: " + patient.patientId);

System.out.println("Name: " + patient.name);

System.out.println("Age: " + patient.age);

System.out.println("Disease: " + patient.disease);

found = true;

break;

}

}

if (!found) {

System.out.println("Patient not found with ID: " + patientId);

}

}

// Add a new doctor

public static void addDoctor() {

System.out.print("Enter Doctor ID: ");

int doctorId = scanner.nextInt();

scanner.nextLine(); // Consume the newline

System.out.print("Enter Doctor Name: ");

String name = scanner.nextLine();

System.out.print("Enter Doctor Specialty: ");

String specialty = scanner.nextLine();

Doctor newDoctor = new Doctor(doctorId, name, specialty);

doctors.add(newDoctor);

System.out.println("Doctor added successfully.");

}

// View all doctors

public static void viewDoctors() {

if (doctors.isEmpty()) {

System.out.println("No doctors found.");

} else {

System.out.println("\nDoctor ID\tName\tSpecialty");

for (Doctor doctor : doctors) {

System.out.println(doctor.doctorId + "\t" + doctor.name + "\t" + doctor.specialty);

}

}

}

// Main menu for the hospital management system

public static void mainMenu() {

while (true) {

System.out.println("\n------ Hospital Management System ------");

System.out.println("1. Add Patient");

System.out.println("2. View Patients");

System.out.println("3. Search Patient by ID");

System.out.println("4. Add Doctor");

System.out.println("5. View Doctors");

System.out.println("6. Exit");

System.out.print("Enter your choice: ");

int choice = scanner.nextInt();

switch (choice) {

case 1:

addPatient();

break;

case 2:

viewPatients();

break;

case 3:

searchPatientById();

break;

case 4:

addDoctor();

break;

case 5:

viewDoctors();

break;

case 6:

System.out.println("Exiting system...");

return; // Exit the program

default:

System.out.println("Invalid choice! Please try again.");

}

}

}

public static void main(String[] args) {

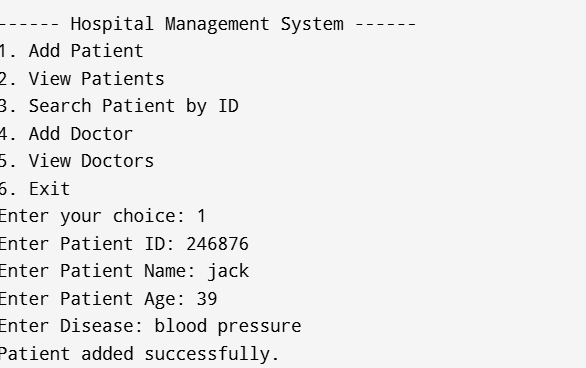
// Start the hospital management system

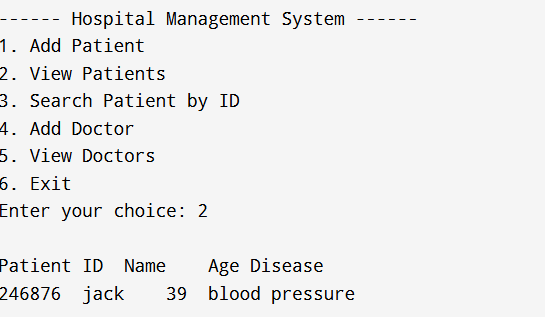
mainMenu();

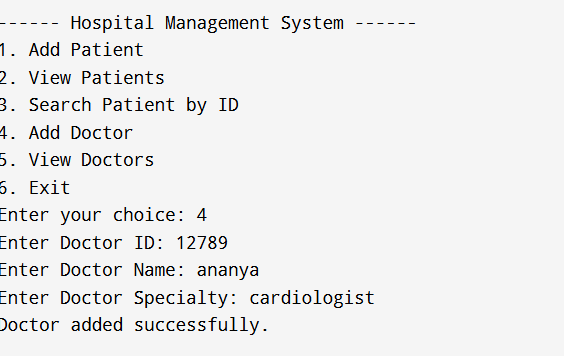
}

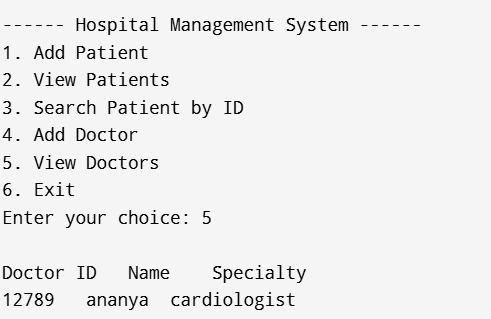
}

**OUTPUT:**

****

****

****

****