import java.util.ArrayList;

import java.util.List;

// User.java - Base class for users

abstract class User {

private String name;

private String email;

private String role;

public User(String name, String email, String role) {

this.name = name;

this.email = email;

this.role = role;

}

public abstract void login();

public void logout() {

System.out.println(name + " logged out.");

}

public String getName() { return name; }

public String getEmail() { return email; }

public String getRole() { return role; }

@Override

public String toString() {

return "Name: " + name + ", Email: " + email + ", Role: " + role;

}

}

// Patient.java

class Patient extends User {

public Patient(String name, String email) {

super(name, email, "Patient");

}

@Override

public void login() {

System.out.println("Patient " + getName() + " logged in.");

}

public void scheduleAppointment(String date) {

System.out.println("Appointment scheduled for " + getName() + " on " + date);

}

}

// Doctor.java

class Doctor extends User {

private String specialty;

public Doctor(String name, String email, String specialty) {

super(name, email, "Doctor");

this.specialty = specialty;

}

@Override

public void login() {

System.out.println("Doctor " + getName() + " logged in.");

}

public void accessDashboard() {

System.out.println("Accessing dashboard with patient data...");

}

}

// Admin.java

class Admin extends User {

public Admin(String name, String email) {

super(name, email, "Admin");

}

@Override

public void login() {

System.out.println("Admin " + getName() + " logged in.");

}

public void manageUserAccounts() {

System.out.println("Managing user accounts...");

}

}

// SystemManagement.java - Manages users and system functionalities

class SystemManagement {

private List<User> users;

public SystemManagement() {

users = new ArrayList<>();

}

// Register a new user

public void registerUser(User user) {

users.add(user);

System.out.println(user.getName() + " registered as " + user.getRole());

}

// Display all users

public void displayAllUsers() {

System.out.println("Displaying all registered users:");

for (User user : users) {

System.out.println(user);

}

}

// Login a user by email (for demonstration purposes)

public void loginUser(String email) {

for (User user : users) {

if (user.getEmail().equalsIgnoreCase(email)) {

user.login();

return;

}

}

System.out.println("User with email " + email + " not found.");

}

// Log out all users

public void logoutAllUsers() {

System.out.println("Logging out all users:");

for (User user : users) {

user.logout();

}

}

public void patientViewHealthData(String patientEmail) {

for (User user : users) {

if (user instanceof Patient && user.getEmail().equalsIgnoreCase(patientEmail)) {

System.out.println("Displaying health data for " + user.getName());

}

}

}

public void doctorPrescribeMedication(String doctorEmail, String medication) {

for (User user : users) {

if (user instanceof Doctor && user.getEmail().equalsIgnoreCase(doctorEmail)) {

System.out.println("Doctor " + user.getName() + " prescribed " + medication);

}

}

}

public void manageCompliance() {

System.out.println("Ensuring compliance with security and data regulations.");

}

public void emergencyAlerts() {

System.out.println("Handling emergency alerts for patients in critical condition.");

}

public void predictiveAnalytics() {

System.out.println("Running predictive analytics on patient data...");

}

}

// Main.java - Testing the functionality

public class Main {

public static void main(String[] args) {

SystemManagement system = new SystemManagement();

// Registering multiple users

system.registerUser(new Patient("Alice", "alice@example.com"));

system.registerUser(new Doctor("Dr. Smith", "drsmith@example.com", "Cardiology"));

system.registerUser(new Admin("AdminUser", "admin@example.com"));

// Display all users

system.displayAllUsers();

// Simulate logging in a user

system.loginUser("drsmith@example.com");

// Sample patient and doctor interactions with requirements

system.patientViewHealthData("alice@example.com"); // Patient views health data

system.doctorPrescribeMedication("drsmith@example.com", "Amoxicillin"); // Doctor prescribes medication

// Execute additional system requirements

system.manageCompliance();

system.emergencyAlerts();

system.predictiveAnalytics();

// Logging out all users

system.logoutAllUsers();

}

}