**Case Study: Health Clinic Management System**

**Section 1: Python Standalone Console Application**

Design and implement a standalone console application for a Health Clinic Management System using Python. The application should utilize collections, object-oriented programming (OOP), and exception handling to manage patient records, doctor schedules, and appointments.

**Requirements:**

1. **Patient Management:**

* Implement the functionality to add, update, and delete patient records.
* Each patient should have attributes such as patient\_id, name, date\_of\_birth, gender, and contact\_info.

1. **Doctor Management:**

* Implement the functionality to manage doctors.
* Each doctor should have attributes such as doctor\_id, name, specialty, contact\_info, and available\_dates.

1. **Appointment Management:**

* Implement the functionality to handle appointments.
* Each appointment should have attributes such as appointment\_id, patient\_id, doctor\_id, appointment\_date, and reason.

**Business Functionalities:**

1. **Manage Patients:**
   * Create a class Patient with attributes patient\_id, name, date\_of\_birth, gender, and contact\_info.
   * Implement methods to add a new patient, update patient details, and delete a patient from the system.
2. **Manage Doctors:**
   * Create a class Doctor with attributes doctor\_id, name, specialty, contact\_info, and available\_dates.
   * Implement methods to add a new doctor, update doctor details, and delete a doctor.
3. **Manage Appointments:**
   * Create a class Appointment with attributes appointment\_id, patient\_id, doctor\_id, appointment\_date, and reason.
   * Implement methods to schedule a new appointment, update appointment details, and cancel an appointment.

**Section 2: MySQL Database Management**

Design a MySQL database schema to support the Health Clinic Management System and provide solutions for the problem statements.

**Table Structures:**

1. **Patients Table:**
   * patient\_id: INT, Primary Key
   * name: VARCHAR(255)
   * date\_of\_birth: DATE
   * gender: VARCHAR(10)
   * contact\_info: VARCHAR(255)
2. **Doctors Table:**
   * doctor\_id: INT, Primary Key
   * name: VARCHAR(255)
   * specialty: VARCHAR(255)
   * contact\_info: VARCHAR(255)
   * available\_dates: TEXT
3. **Appointments Table:**
   * appointment\_id: INT, Primary Key
   * patient\_id: INT, Foreign Key References Patients(patient\_id)
   * doctor\_id: INT, Foreign Key References Doctors(doctor\_id)
   * appointment\_date: DATE
   * reason: VARCHAR(255)

**Problem Statements:**

* Write a query to find the number of appointments scheduled for each doctor.
* Write a query to find the patients who have appointments in the next week.
* Write a query to find the doctors who specialize in a specific field.
* Write a query to find the available dates for a particular doctor.
* Write a query to find the appointment history of a specific patient.