Declaration of Original Work for SC2002/CE2002/CZ2002 Assignment

We hereby declare that the attached group assignment has been researched, undertaken, completed, and submitted as a collective effort by the group members listed below.

We have honored the principles of academic integrity and have upheld Student Code of Academic Conduct in the completion of this work

We understand that if plagiarism is found in the assignment, then lower marks or no marks will be awarded for the assessed work. In addition, disciplinary actions may be taken.

	Course (SC2002/CE2002 CZ2002)	Lab Group	Date
Maximilian Wong Junlin	SC2002	SCE4 Group 1	17/11/2024
Chin Shao Yang			17/11/2024
Fernandez Anandaraj Joyanne			17/11/2024
Tan Xin Min			17/11/2024
Ma Jinlin			17/11/2024

SC2002 Health Management System

Principles Used

Encapsulation

Encapsulation restricts direct access to some components of objects, which is essential for protecting the integrity of an object's data. Here's how it applies in the HMS: **Patient Data**: Sensitive details in the Patient class, such as medical history, should be marked as private. Access to this information should be controlled through getter methods, ensuring that only authorized classes (like the Doctor class) can view or modify this information. **Protected Password Field**: The password field can be marked as protected to allow subclasses to access but keep it hidden from other classes.**System-Wide Variables**: Constants and other variables that need to be accessible across the application, like hospital ID patterns, are declared as public static final, ensuring that they remain consistent and unchanged.

Polymorphism

Polymorphism allows objects to be treated as instances of their parent class, enabling flexibility and reusability: **Role-Based Menus**: All user roles (Patient, Doctor, Pharmacist, Administrator) inherit from a User superclass. Each role can override methods to define specific functionalities, such as appointment handling for the Patient and Doctor classes. **Appointment Management**: Using polymorphism, the system can handle appointments in a way where each user interacts with them according to their role's specific requirements.

Inheritance

Inheritance enables new classes to adopt properties and methods of existing classes, promoting code reuse: **User Role Hierarchy**: The User class acts as a superclass with shared attributes like userID and password, which the Patient, Doctor, Pharmacist, and Administrator classes inherit. This reduces code duplication and centralizes user authentication features. **Staff Management**: By creating a general Staff class, both Doctor and Pharmacist can inherit common attributes (such as ID and name), with each subclass adding its role-specific details.

Abstraction

Abstraction hides the complexity of the system by focusing on the essential attributes and behaviors: **Abstract User Class**: The User class can be made abstract to define essential functionalities that all users must implement, such as login() and logout() methods. This way, each role-specific user only needs to implement methods that are unique to their role, ensuring that the HMS remains modular and maintainable. **Appointment and Inventory Management**: By abstracting out inventory and appointment management details, classes interacting with these functionalities don't need to know about the underlying database or storage details. They only interact with exposed interfaces or abstract methods.

Additional Details Based on Requirements

System Initialization: The HMS requires loading initial data for staff, patients, and inventory from files. This setup can be managed through encapsulation and abstraction by creating utility classes responsible for reading and setting up data without exposing details to the main program. **Error Handling and Data Integrity**: Each user's data access should be restricted through encapsulated methods to prevent unauthorized modification, e.g., patients cannot change medical records except for non-medical information.

SOLID Principles

Single Responsibility Principle (SRP)

The SRP states that a class should have only one reason to change, and should only have one job or responsibility. **User Classes**: Users (Patient, Doctor, Pharmacist, and Administrator classes) are encapsulated in their class, each responsible for handling actions specific to that role, like viewing or modifying medical records for Doctors. This ensures each user class only manages its behavior and data. **Separation of Objects**: Objects such as MedicalRecord, Appointment, and ReplenishmentRequest are managed by their respective classes. MedicalRecord handles patient

records, Appointment manages scheduling, and ReplenishmentRequest handles inventory requests. This ensures each class only has a single responsibility.

Open/Closed Principle (OCP)

The OCP states software entities should be open for extension but closed for modification. This means we should be able to add new functionality to a class without altering its existing code. **Abstract User Class**: The User class is defined as abstract, allowing the creation of new user roles (e.g., Nurse, Technician) by inheriting from this class. This keeps the original code and satisfies OCP.**Extensible Modules**: Appointment and Inventory are designed to handle various types of data and user interactions. If new functionalities are needed (adding new appointment types), developers can add new classes or methods inheriting from the existing Appointment class without altering its functionality.

Liskov Substitution Principle (LSP)

The LSP suggests that subclasses should be substitutable for their base classes without altering the correctness of the program. Derived classes should be able to operate in place of their base classes without unexpected behavior. **Consistency in Subclasses**: SUbclasses of User inherit its attributes and methods. ResetPassword() operates identically across all subclasses without needing any overriding, allowing any User object to be used interchangeably in contexts where these methods are needed. **Role-Specific Methods**: Each user subclass implements methods specific to themselves, Doctor has updateMedicalRecord() while Patient does not, this ensures each subclass behaves predictably and satisfies the LSP.

Interface Segregation Principle (ISP)

The ISP states Instead of having a large, generalized interface, smaller, more focused interfaces that are specific to the needs of the classes that use them are better. **INamable Interface**: Inamable interface is realized by all classes with names, mainly User and Medicine. The abstract User class makes its subclasses to be nameable.. This implements a focused small interface specific to nameable objects. **No Forced Implementation**: Each user class only implements the methods it needs. Patient and Doctor classes only need to retrieve records, so IGetRecord includes only the getRecord method, satisfying the ISP.

Dependency Inversion Principle (DIP)

The DIP states high-level modules shouldn't depend on low-level modules but on abstractions. **Minimal Direct Dependency Among Classes**: Classes interact with one another through HMSdatabase or abstract data access layers rather than directly with one another. Doctor and Patient classes don’t depend on each other directly for the viewing or scheduling of appointments, but with the AppointmentManager, which handles the interactions and data handling. **Abstractions over Implementations**: Doctor, Patient, Administrator, and Pharmacist depend on the abstract User class, allowing the system to scale or change independently. Implementations for each role's specific actions are encapsulated in their respective classes, ensuring modularity and extensibility.

Summary of SOLID Principles in HMS

By applying these SOLID principles, the HMS achieves:

1. **Modularity and Maintainability**: Changes to one part of the system (like adding a new user role) don't require rewriting other parts.
2. **Extensibility**: New features can be added easily without modifying existing code, ensuring the system remains stable.
3. **Flexibility**: Different classes can be used interchangeably without breaking functionality, enhancing reusability and reducing dependency.
4. **Reduced Complexity**: Each component has a single purpose, making the system easier to understand, debug, and test.

Assumptions Made

User Authentication

As a more complex database system is not used, sensitive information such as user passwords are not hashed when being stored and will be stored in plaintext within the CSVs in which they reside instead.

UserType Enumerator

As we have 2 different types of users (Staff and Patient), and 3 different types of Staff (Doctor, Pharmacist, and Administrator), an ENUM datatype is used to store their respective user types. The enum will have the values of PATIENT, DOCTOR, PHARMACIST, and ADMINISTRATOR. These values are defined at the system and Database Initialization.

Database Storage

Common data that is used systemwide by various separate classes are entirely stored within the HMS database class. These include arrays to store the various lists used within the system. These Arrays Include: (Users, Patients, Pharmacists, Doctors, Administrators, Staff, MedicineInventory, Appointments, replenishmentRequests, and Medical Records). Additionally, an instance of the ApptManager class is instantiated within the HMSDatabase class to allow management and interfacing of the Appointment Array with other classes. This approach allows all data and information to be centrally managed by the HMS database class and consolidated in one central location.

HMSDatabase

As a SQL database is not being used, all data used in the system is saved in separate CSV files i.e. Patient_List.csv, Staff_List.csv, Medicine_List.csv, etc These files are then processed upon system initialization into their respective object types and arrays then subsequently stored within the HMSdatabase Class.

DatabaseHelper

A Helper class is used to assist the HMS database class for data work. This includes the initialization of arrays, Modification of the database, saving of data to CSVs, etc. The helper comprises of entirely static variables and functions to allow modification of the database System Wide.

Working Hours

Doctor working hours are assumed to be 7 days a week, 9 am - 6 pm.

Medical Records

We assume that in real life, to have a medical record for a patient, the patient first needs to see a doctor to diagnose their issue. Similarly, for a patient to have a medical record, they will need to also have an appointment.

Additional Features/Enhancements

Database Features

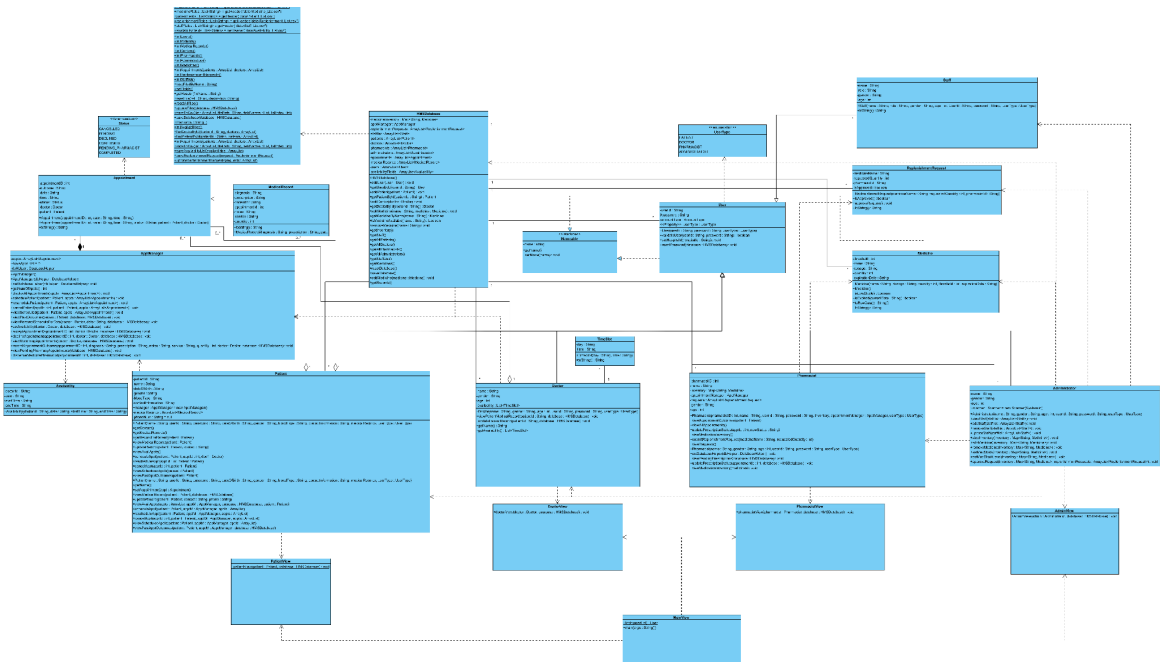
Persistent Data

Modifications that are made such as adding users, making appointments, etc. are persistent and changes can be saved into the database between sessions. This is done by saving changes to the respective arrays within the HMSdatabase class and rewriting them back into the CSVs where the initial data is being stored.

Resetting of database

Administrators can reset the database back to its initial state, copying a reference CSV which stays unchanged into the working sets of CSVs. This allows the system to be reset back to its original state for testing purposes or in case something breaks.

Detailed UML Class Diagram



Classes

Administrator,AdminView,Appointment,ApptManager,Availability,DatabaseHelper,Doctor,DoctorView,HMSDatabase,MainView,MedicalRecord,Medicine,Patient,PatientView,Pharmacist,PharmacistView,ReplenishmentRequest,Staff,TimeSlot,User

Interface

INameable

Enumerator

Status, UserType

Relationships

Dependency

Admin-> Doctor, AdminView, HMSDatabase, Medicine, Patient, Pharmacist, ReplenishmentRequest, Staff; Appointment-> Status; Doctor-> ApptManager, DoctorView; HMSDatabase->DatabaseHelper, ApptManager; MedicalRecord->Appointment, Patient-> PatientView, Pharmacist-> Medicine, PharmacistView, RepleneishmentRequest; User-> UserType; ApptManager-> Availability; MainView ->PatientView, DoctorView, PharmacistView, AdminView

Association

Admin-HMSDataBase (1-*), Doctor-Appointment, Doctor-HMSDatabase (*-1), User-HMSDatabase (*-1), Patient-HMSDatabase (*-1), Pharmacist-HMSDatabase (*-1),Staff-HMSDatabase (*-1), Medicine-HMSDatabase (*-1), TimeSlot-Doctor (1-*)

Aggregation

Patient->Appointment(1-0..*). patient->MedicalRecord(1-0..*)

Composition

ApptManager->Appointment(1-*)

Generalization

User->Admin,Doctor,Patient,Pharmacist,Staff

Realization

INameable->User, Medicine

Test Cases

Patient

Patient Menu

```
==================================================
                --- Patient Menu ---
==================================================
| 1  | View Medical Record                        |
| 2  | Update Personal Information                |
| 3  | View Available Appointment Slots           |
| 4  | Schedule an Appointment                    |
| 5  | Reschedule an Appointment                  |
| 6  | Cancel an Appointment                      |
| 7  | View scheduled Appointments                |
| 8  | View Past Appointment Outcome Records      |
| 9  | Exit                                       |
==================================================
Choose an option:
```

Test Case 1: View Medical Record

```
Choose an option: 1

--- Medical Record: P1001 ---
==========================================================================================================
| Patient ID | Name             | DOB        | Gender | Blood Type | Email                     | Phone Number |
==========================================================================================================
| P1001      | Alice Brown      | 5/14/1980  | Female | A+         | alice.brown@example.com   | not available |
==========================================================================================================
```

Test Case 2: Update Personal Information

```
Choose an option: 2
Enter email: change@email.com
Enter phone number: 89898989
Updated medical record for patient ID:

--- Medical Record: P1001 ---
==========================================================================================================
| Patient ID | Name             | DOB        | Gender | Blood Type | Email                     | Phone Number |
==========================================================================================================
| P1001      | Alice Brown      | 5/14/1980  | Female | A+         | change@email.com          | 89898989     |
==========================================================================================================
```

Test Case 3: View Available Appointment Slots

```
Choose an option: 3


Appointment_List.csv updated successfully.
==========================================================================================================
| Appt ID    | Patient          | Patient Id | Doctor          | Doctor Id  | Status     | Date       | Time       |
==========================================================================================================
| 22         | NA               | NA         | John Smith      | D001       | PENDING    | 17-Nov     | 12:30 PM   |
| 23         | NA               | NA         | John Smith      | D001       | PENDING    | 17-Nov     | 05:30 PM   |
| 24         | NA               | NA         | John Smith      | D001       | PENDING    | 17-Nov     | 06:00 PM   |
| 25         | NA               | NA         | Emily Clarke    | D002       | PENDING    | 18-Nov     | 09:00 AM   |
| 26         | NA               | NA         | Emily Clarke    | D002       | PENDING    | 18-Nov     | 09:30 AM   |
| 27         | NA               | NA         | Emily Clarke    | D002       | PENDING    | 18-Nov     | 01:30 PM   |
| 28         | NA               | NA         | Emily Clarke    | D002       | PENDING    | 18-Nov     | 05:30 PM   |
| 29         | NA               | NA         | Emily Clarke    | D002       | PENDING    | 18-Nov     | 06:00 PM   |
==========================================================================================================
```

Test Case 4: Schedule an Appointment

```
Choose an option: 4
Enter an available appointment ID:
22
Appointment scheduled successfully for patient ID P1001 with Doctor John Smith on 17-Nov at 12:30 PM
```

Test Case 5: Reschedule an Appointment

```
Choose an option: 5
Your scheduled appointments:
==========================================================================================================
| Appt ID    | Patient          | Patient Id | Doctor          | Doctor Id  | Status     | Date       | Time       |
==========================================================================================================
| 22         | Alice Brown      | P1001      | John Smith      | D001       | CONFIRMED  | 17-Nov     | 12:30 PM   |
==========================================================================================================

Enter Appointment ID to reschedule:
22

Enter new Appointment ID:
27
Appointment rescheduled successfully to
Appointment ID: 27, Patient: Alice Brown, Patient ID: P1001, Doctor: Emily Clarke, Doctor ID: D002, Status: CONFIRMED, Date: 18-Nov, Time: 01:30 PM
```

Test Case 6: Cancel an Appointment

```
Choose an option: 6
Your scheduled appointments:
==================================================================================================================
| Appt ID    | Patient              | Patient Id | Doctor          | Doctor Id  | Status     | Date       | Time       |
==================================================================================================================
| 27         | Alice Brown          | P1001      | Emily Clarke    | D002       | CONFIRMED  | 18-Nov     | 01:30 PM   |
==================================================================================================================
Enter Appointment ID to cancel:
27
Appointment canceled successfully for patient ID P1001
```

Test Case 7: View Scheduled Appointments

```
Choose an option: 7
==================================================================================================================
| Appt ID    | Patient              | Patient Id | Doctor          | Doctor Id  | Status     | Date       | Time       |
==================================================================================================================
| 22         | Alice Brown          | P1001      | John Smith      | D001       | CONFIRMED  | 17-Nov     | 12:30 PM   |
==================================================================================================================
```

Test Case 8: View Past Appointment Outcome Records

```
Choose an option: 8
==================================================================================================================
| Appointment ID  | Doctor          | Diagnosis       | Prescription    | Service         | Notes                |
==================================================================================================================
| 6               | John Smith      | Cold            | Paracetamol     | General Checkup | Routine check        |
| 13              | Emily Clarke    | Allergy         | Cough Syrup     | Pee Test        | Routine check        |
| 19              | John Smith      | Cold            | Ibuprofen       | Pee Test        | Routine check        |
==================================================================================================================
```

Doctor

Doctor Menu

```
=====================================
          --- Doctor Menu ---
=====================================
| 1  | View Patient Medical Record    |
| 2  | Update Patient Medical Record  |
| 3  | View Personal Schedule         |
| 4  | Set Availability               |
| 5  | Accept Appointment             |
| 6  | Decline Appointment            |
| 7  | View Upcoming Appointments     |
| 8  | Record Appointment Outcome     |
| 9  | Exit                           |
| 10 | Logout                         |
=====================================
```

Test Case 9: View Patient Medical Records

```
Choose an option: 1
Enter Patient ID to view their record: P1001

--- Medical Records for Patient ID: P1001 ---
==================================================================================================================
| Appointment ID  | Doctor          | Diagnosis       | Prescription    | Service         | Notes                |
==================================================================================================================
| 1               | John Smith      | Fever           | Antihistamine   | X-Ray           | Follow-up required   |
| 6               | John Smith      | Cold            | Ibuprofen       | Pee Test        | Routine check        |
==================================================================================================================
```

Test Case 10: Update Patient Record

```
Choose an option: 2
Enter Patient ID: P1001

What do you want to update?
1. Diagnosis
2. Prescription
3. Service
Enter choice (1/2/3): 1

--- Matching Records for Patient ID: P1001 and Field: Diagnosis ---
==================================================================
| No.   | Appointment ID  | Diagnosis           | Notes           |
==================================================================
| 1     | 3               | Allergy             | Routine check   |
| 2     | 1               | Fever               | Follow-up required |
| 3     | 6               | Cold                | Routine check   |
==================================================================
Enter the number of the record you want to update: 1
Enter new value for Diagnosis: Allergiess

Diagnosis updated successfully for Appointment ID: 3
```

Test Case 11: View Personal Schedule

```
Choose an option: 3
Enter the date to view your schedule (dd-MMM, e.g., 17-Nov): 17-Nov

--- Personal Schedule for Dr. John Smith on 17-Nov ---
=====================================
| Time       | Appointment          |
=====================================
| 09:00 AM   | Not Available        |
| 09:30 AM   | Not Available        |
| 10:00 AM   | Not Available        |
| 10:30 AM   | Not Available        |
| 11:00 AM   | Not Available        |
| 11:30 AM   | Not Available        |
| 12:00 PM   | Not Available        |
| 12:30 PM   | Free                 |
| 01:00 PM   | Not Available        |
| 01:00 PM   | Not Available        |
| 01:30 PM   | Not Available        |
| 02:00 PM   | Not Available        |
| 02:30 PM   | Not Available        |
| 03:00 PM   | Not Available        |
| 03:30 PM   | Not Available        |
| 04:00 PM   | Not Available        |
| 04:30 PM   | Free                 |
| 05:00 PM   | Bob Stone (P1002)    |
| 05:30 PM   | Free                 |
| 06:00 PM   | Free                 |
=====================================
```

Test Case 12: Set Availability for Appointments

```
Choose an option: 4
Enter date (dd-MMM, e.g., 17-Nov): 17-Nov
Enter start time (hh:mm AM/PM, e.g., 09:00 AM): 10:00 AM
Enter end time (hh:mm AM/PM, e.g., 05:00 PM): 12:00 PM
Availability_List.csv updated successfully.

--- Availability Added ---
==========================================================
| Doctor ID        | Date        | Start Time | End Time    |
==========================================================
| D002             | 17-Nov      | 10:00 AM   | 12:00 PM    |
==========================================================
```

Test Case 13: Accept or Decline Appointment Requests

```
Choose an option: 5
--- PENDING Appointments for Dr. Emily Clarke ---
==============================================================================
| Appointment ID  | Patient ID      | Patient Name    | Date       | Time       |
==============================================================================
| 2               | P1003           | Charlie White   | 10-Nov     | 4:30 PM    |
| 4               | P1002           | Bob Stone       | 19-Nov     | 11:00 AM   |
==============================================================================
Enter appointment ID to accept: 2

--- Appointment Accepted ---
==========================================================================
| Appointment ID  | Patient ID      | Date       | Time       | Status     |
==========================================================================
| 2               | P1003           | 10-Nov     | 4:30 PM    | CONFIRMED  |
==========================================================================
```

```
Choose an option: 6
--- PENDING Appointments for Dr. Emily Clarke ---
==============================================================================
| Appointment ID  | Patient ID      | Patient Name    | Date       | Time       |
==============================================================================
| 4               | P1002           | Bob Stone       | 19-Nov     | 11:00 AM   |
==============================================================================
Enter appointment ID to decline: 4

--- Appointment Declined ---
==========================================================================
| Appointment ID  | Patient ID      | Date       | Time       | Status     |
==========================================================================
| 4               | P1002           | 19-Nov     | 11:00 AM   | DECLINED   |
==========================================================================
```

Test Case 14: View Upcoming appointments

```
Choose an option: 7

--- Upcoming Appointments for Dr. Emily Clarke ---
==========================================================================
| Appointment ID  | Patient Name        | Date            | Time       |
==========================================================================
| 5               | Bob Stone           | 01-Nov          | 10:30 AM   |
| 10              | Alice Brown         | 05-Nov          | 12:30 PM   |
==========================================================================
```

Test Case 15: Recording Appointment Outcomes

```
Choose an option: 8

--- CONFIRMED Appointments for Dr. Emily Clarke ---
==================================================================================
| Appointment ID  | Patient ID      | Patient Name        | Date       | Time       |
==================================================================================
| 5               | P1002           | Bob Stone           | 01-Nov     | 10:30 AM   |
| 10              | P1001           | Alice Brown         | 05-Nov     | 12:30 PM   |
==================================================================================
Enter appointment ID to record outcome: 5

--- Record Outcome Details ---
Enter Diagnosis: Flu
Enter Prescription: Telfast
Enter Notes: runny nose
Enter Service Provided: general consult
Enter Quantity Provided: 5

Appointment ID 5 marked as PENDING_PHARMACIST.
Medical record updated for Patient ID: P1002
```

Pharmacist

Pharmacist Menu

```
========================================
        --- Pharmacist Menu ---
========================================
| 1  | View Appointments               |
| 2  | Update Prescription Status      |
| 3  | View Medicine Inventory         |
| 4  | Submit Replenishment Request    |
| 5  | Logout                          |
========================================
```

Test Case 16: View Appointment Outcome Record

```
Select an option: 1

Viewing all Pending Pharmacy Appointments:
---------------------------------------
Appointment ID: 6
Date: 25-Nov
Time: 02:00 PM
Status: PENDING_PHARMACIST
Doctor: John Smith
Patient ID: P1001
Prescription: Paracetamol
Quantity: 8
---------------------------------------
Appointment ID: 13
Date: 5-Nov
Time: 09:30 AM
Status: PENDING_PHARMACIST
Doctor: Emily Clarke
Patient ID: P1001
Prescription: Cough Syrup
Quantity: 8
---------------------------------------
```

Test Case 17: Update Prescription Status

```
Select an option: 2
Enter Appointment ID:
6
Successfully dispensed 8 of Paracetamol. Updated inventory.
```

Test Case 18: View Medication Inventory

```
Select an option: 3
           --- Medication Inventory ---
==================================================
| Medicine Name        | Quantity    | Threshold  |
==================================================
| Paracetamol          | 100         | 20         |
| Ibuprofen            | 5           | 10         |
(Low Stock Warning)
| Amoxicillin          | 75          | 15         |
==============================================
```

Test Case 19: Submit Replenishment Request

```
Select an option: 4

Enter medicine name for replenishment: Ibuprofen
Enter quantity to request: 100

Replenishment Request Submitted Successfully
Medicine: Ibuprofen
Quantity Requested: 100
Request Status: Pending administrator approval
Replenishment request submitted successfully and saved.
```

Administrator

Admin Menu

```
==============================================
            --- Administrator Menu ---
==============================================
| 1  | View and Manage Hospital Staff          |
| 2  | View Appointment Details                |
| 3  | View and Manage Medication Inventory    |
| 4  | Approve Replenishment Requests          |
| 5  | Save Database                           |
| 6  | Reset Database                          |
| 7  | Logout                                  |
==============================================
```

Test Case 20: View Staff

```
                    --- Hospital Staff ---
==================================================================
| ID        | Name            | Role            | Gender | Age   |
==================================================================
| D001      | John Smith      | Doctor          | Male   | 45    |
| D002      | Emily Clarke    | Doctor          | Female | 38    |
| P001      | Mark Lee        | Pharmacist      | Male   | 29    |
| A001      | Sarah Lee       | Administrator   | Female | 40    |
==================================================================
Filter by:
0. No Filter
1. Role: Doctors
2. Role: Pharmacists
3. Role: Administrators
4. Gender: Male
5. Gender: Female
6. Age: 21-30
7. Age: 31-40
8. Age: 41-50
9. Exit View Staff
```

Filtering (i.e. Doctor)

```
                    --- Hospital Staff (Doctors) ---
==================================================================
| ID        | Name            | Role            | Gender | Age   |
==================================================================
| D001      | John Smith      | Doctor          | Male   | 45    |
| D002      | Emily Clarke    | Doctor          | Female | 38    |
==================================================================
```

Test case 20: Manage Staff

```
--- Manage Staff ---
1. Add Staff
2. Remove Staff
3. Update Staff details
4. Exit
1
Enter name of staff to add: Jenny Tan
Enter ID of staff to add: A002
Enter role of staff to add: Administrator
Enter gender of staff to add: Female
Enter age of staff to add: 45
Staff member A002 has been added.
```

Remove Staff

```
Enter ID of staff to be removed: D002
Staff ID D002 has been removed.
```

Update Staff

```
Enter ID of staff to be updated: D003
Update:
1. ID
2. Name
3. Role
4. Gender
5. Age
6. Exit
2
Enter new name: Mary John
```

Test Case 21:View Appointments(with outcomes)

```
--- All Appointments ---
| Appt ID    | Patient Name         | Patient ID           | Doctor Name          | Doctor ID            | Status               | Date      | Time      |
| 1          | Bob Stone            | P1002                | John Smith           | D001                 | CONFIRMED            | 17-Nov    | 05:00 PM  |
| 2          | Charlie White        | P1003                | Emily Clarke         | D002                 | PENDING              | 10-Nov    | 04:30 PM  |
| 3          | Charlie White        | P1003                | Mary Jane            | D003                 | COMPLETED            | 21-Nov    | 11:30 AM  |
| OUTCOME    | Diagnosis: Allergy   | Prescription: Antibiotics | Service: General Checkup | Notes: Routine check |                      |           |           |
| 4          | Bob Stone            | P1002                | Emily Clarke         | D002                 | PENDING              | 19-Nov    | 11:00 AM  |
| 5          | Bob Stone            | P1002                | Emily Clarke         | D002                 | CONFIRMED            | 1-Nov     | 10:30 AM  |
| 6          | Alice Brown          | P1001                | John Smith           | D001                 | PENDING_PHARMACIST   | 25-Nov    | 02:00 PM  |
| 7          | Charlie White        | P1003                | Mary Jane            | D003                 | COMPLETED            | 20-Nov    | 02:30 PM  |
| OUTCOME    | Diagnosis: Headache  | Prescription: Ibuprofen | Service: Ultrasound | Notes: I think he sick |                      |           |           |
| 8          | Charlie White        | P1003                | John Smith           | D001                 | PENDING              | 28-Nov    | 04:00 PM  |
| 9          | Alice Brown          | P1001                | Mary Jane            | D003                 | DECLINED             | 23-Nov    | 10:00 AM  |
| 10         | Alice Brown          | P1001                | Emily Clarke         | D002                 | PENDING              | 5-Nov     | 12:30 PM  |
| 11         | Bob Stone            | P1002                | John Smith           | D001                 | DECLINED             | 15-Nov    | 05:30 PM  |
| 12         | Bob Stone            | P1002                | Mary Jane            | D003                 | CANCELLED            | 9-Nov     | 10:30 AM  |
| 13         | Alice Brown          | P1001                | Emily Clarke         | D002                 | PENDING_PHARMACIST   | 5-Nov     | 09:30 AM  |
| 14         | Bob Stone            | P1002                | Mary Jane            | D003                 | PENDING_PHARMACIST   | 7-Nov     | 11:00 AM  |
| 15         | Bob Stone            | P1002                | Emily Clarke         | D002                 | COMPLETED            | 1-Dec     | 10:30 AM  |
| OUTCOME    | Diagnosis: Headache  | Prescription: Ibuprofen | Service: Knee Surgery | Notes: Looks serious |                      |           |           |
| 16         | Alice Brown          | P1001                | Mary Jane            | D003                 | DECLINED             | 3-Nov     | 02:00 PM  |
| 17         | Bob Stone            | P1002                | Mary Jane            | D003                 | PENDING              | 20-Nov    | 09:30 AM  |
| 18         | Bob Stone            | P1002                | Emily Clarke         | D002                 | PENDING              | 24-Nov    | 05:30 PM  |
| 19         | Alice Brown          | P1001                | John Smith           | D001                 | PENDING_PHARMACIST   | 14-Nov    | 09:30 AM  |
| 20         | Alice Brown          | P1001                | Mary Jane            | D003                 | CANCELLED            | 16-Nov    | 02:30 PM  |
```

Test case 22: View Inventory

```
--- Medication Inventory ---
| Medicine Name        | Quantity    | Threshold   |
| Paracetamol          | 100         | 20          |
| Ibuprofen            | 5           | 10          |
| Amoxicillin          | 90          | 15          |
--- Manage inventory ---
1. Add Medicine
2. Remove Medicine
3. Update Medicine Stocks
4. Update Medicine Low Stock Level
5. Exit
```

Test Case 22: Manage Inventory

```
Enter name of new medicine: Panadol
Enter stock of new medicine: 50
Enter low stock alert level of new medicine: 20
Panadol has been added to the inventory
```

Remove Medicine

```
Enter name of medicine to be removed: Amoxicillin
Amoxicillin has been removed.
```

Updating stock

```
Enter name of medicine to update stock: Paracetamol
Enter new stock level: 90
Paracetamol's stock has been updated to 90
```

Changing Low stock Level Alert

```
Enter name of medicine to change low stock alert level: Paracetamol
Enter new alert level:25
Paracetamol's low stock alert level has been updated to 25
```

Test case 23: Replenishment Request

```
--- Replenishment Requests ---
| Name            | Requested Quantity   | Pharmacist ID   | Status      |
| Paracetamol     | 100                  | P001            | DECLINED    |
| Ibuprofen       | 100                  | P001            | PENDING     |
Enter the name of the medicine whose request is to be approved/declined: Ibuprofen
Request for Ibuprofen has been approved. The medicine's stock is now 105
```

Login System and Password Management

Test case 25: User Login

```
1. Login
2. Reset Password
3. Exit
Please Select Option:
1
 _    _ __  __  _____
| |  | |  \/  |/ ____|
| |__| | \  / | (___
|  __  | |\/| |\___ \
| |  | | |  | |____) |
|_|  |_|_|  |_|_____/

=====================================
        Welcome to the
   Hospital Management System
=====================================

Enter your hospital ID: D001
Enter your password: password1!

Login successful! Welcome, D001.
```

```
1. Login
2. Reset Password
3. Exit
Please Select Option:
1
 _    _ __  __  _____
| |  | |  \/  |/ ____|
| |__| | \  / | (___
|  __  | |\/| |\___ \
| |  | | |  | |____) |
|_|  |_|_|  |_|_____/

=====================================
        Welcome to the
   Hospital Management System
=====================================

Enter your hospital ID: P1003
Enter your password: Password

First login detected. Please reset your password.
New password: :
password1!
Password reset

Login successful! Welcome, P1003.
```

Test case 26: Wrong Password

```
1. Login
2. Reset Password
3. Exit
Please Select Option:
1
 _    _ __  __  _____
| |  | |  \/  |/ ____|
| |__| | \  / | (___
|  __  | |\/| |\___ \
| |  | | |  | |____) |
|_|  |_|_|  |_|_____/

=====================================
        Welcome to the
   Hospital Management System
=====================================

Enter your hospital ID: D001
Enter your password: aaaaaa

Invalid login credentials. Please try again.
```

Reflection

Difficulties encountered:

- Keeping track of available appointment slots, handling booking and rescheduling appointments
  We store a list of appointments in a file called Appointment_List.csv. There is a class called ApptManager to handle the logic for booking and rescheduling.
- Importing Staff/Patient etc lists from files
- We made sure that the columns in the CSV files matched the fields for the objects in our HMS database. Errors or incorrect data placement may result from improper matching.

Knowledge gained:

- We learned how to design a hospital management system in an object-oriented way using concepts, like patients, doctors, appointments, and bills, which can be modeled as objects.
- **Abstraction**: You use interfaces and abstract classes to keep the functionality and data apart. An abstract class, like the User class, might specify the user-related actions without disclosing the implementation specified in the Patient, Doctor, Pharmacist, and Administrator accounts.
- **Inheritance**: Specialized classes like Patient, Doctor, Pharmacist, and Administrator inherit base classes like User. The specialized classes can inherit variables from the base class, like userId, password, and userType.
- **Encapsulation:** Every system entity, such as a patient, doctor, appointment, etc, has its unique data and methods to interact with it. For instance, you could have methods like getPatientId() so that the variable patientId is read-only.

We learned that the capacity to load and preserve data from external files is frequently necessary for an HMS. This could be accomplished using text-based formats like CSV to store the basic data in text files.

Further Improvement Suggestions:

- **User-Friendly Interface**: A responsive and user-friendly graphical user interface, for instance, a central dashboard that shows important data and real-time indicators, such as future appointments and prescriptions for patients.
- **Database improvements**: Using a database like MongoDB can provide several benefits like scalability, as a developing hospital administration system that handles substantial volumes of patient data can benefit from having a dedicated database system.
- **Enhanced Error Handling**: We can record information like problem messages, impacted modules, timestamps, and the user who experienced the error. We can implement a centralized logging system that keeps track of all errors.