GLOBAL QUEST
TECHNOLOGIES

PROJECT REPORT

TITLE : MEDICAL CLINIC APPOINTMENT SYSTEM

Submitted By : B. Geethika

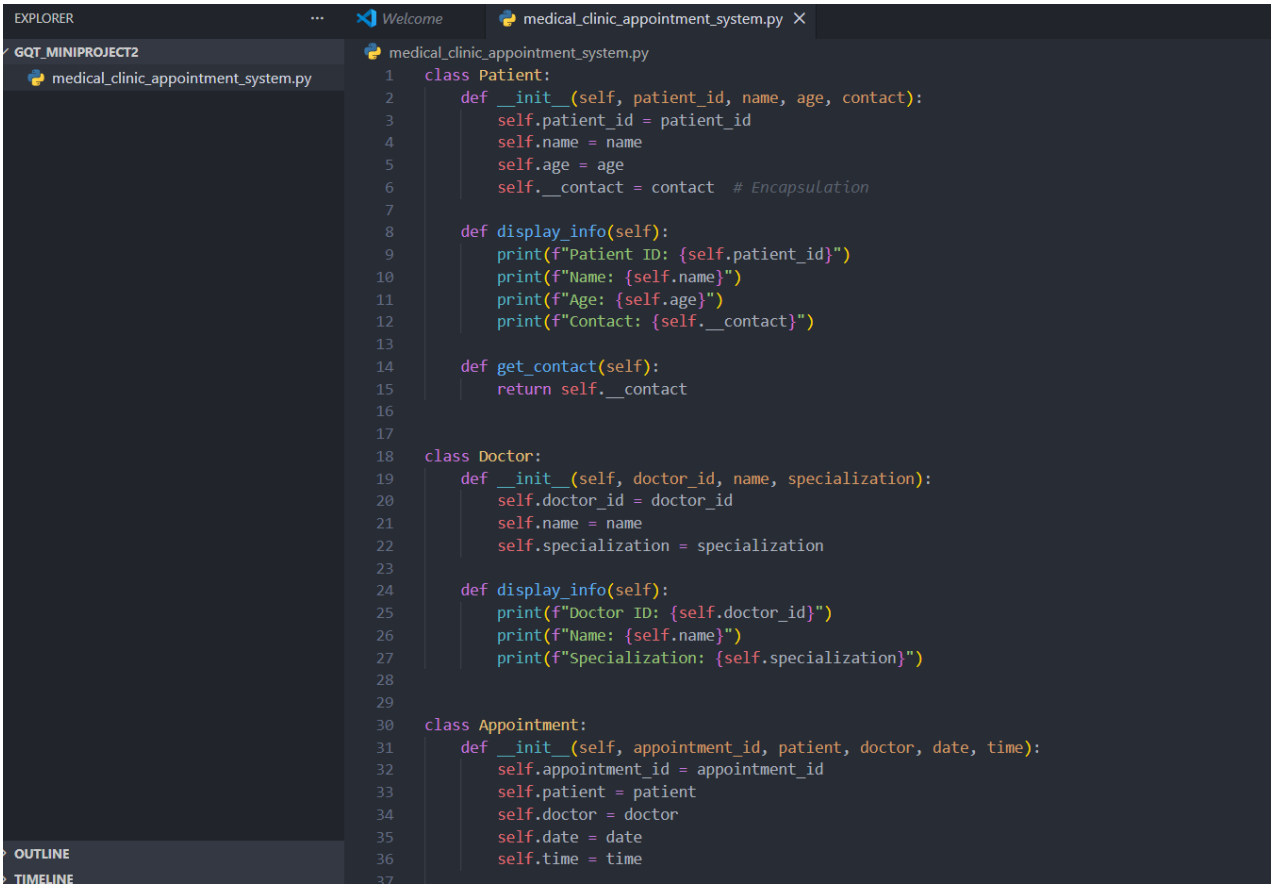
Submitted to : Global Quest Technologies

Date : 15 February 2026

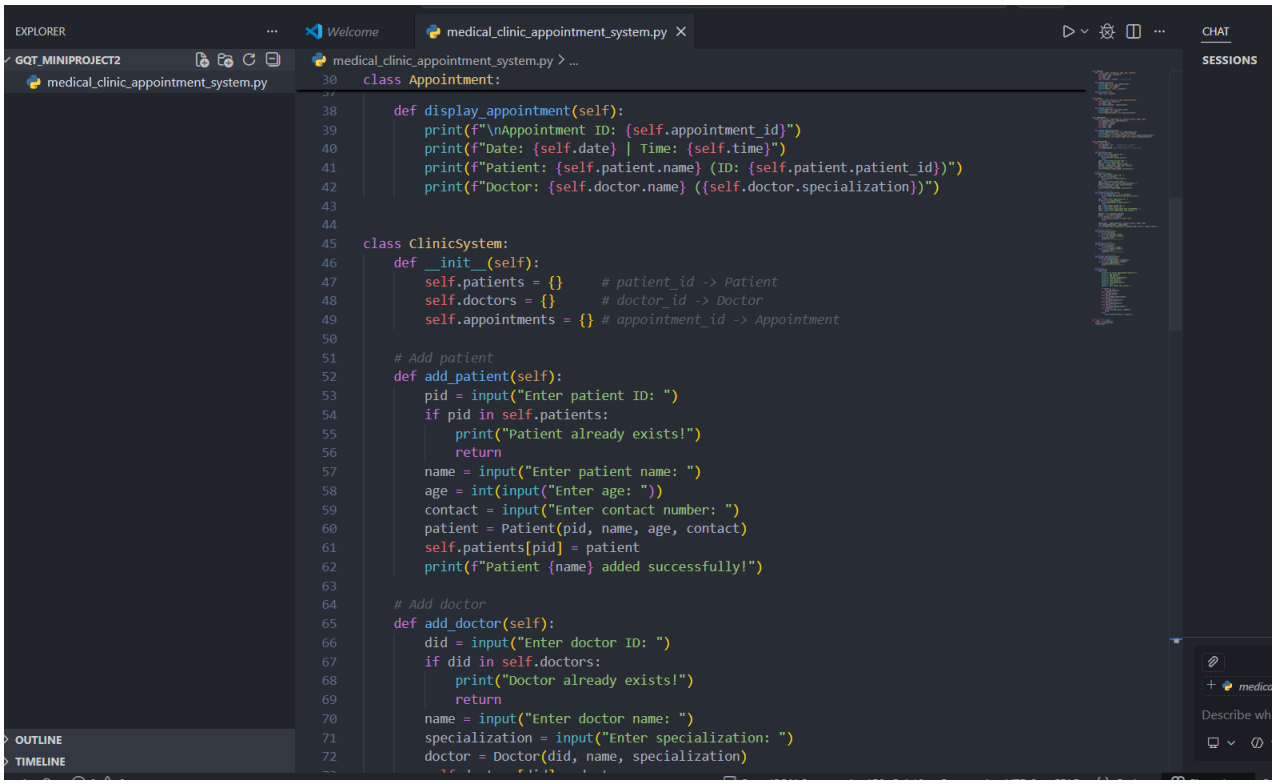
ABSTRACT

The Hotel Reservation System is a console-based Python application designed to automate and simplify hotel management tasks such as room management, guest management, and reservation scheduling. The system is built using Object-Oriented Programming (OOP) concepts, including classes, objects, inheritance, encapsulation, polymorphism, and composition, to ensure modularity, scalability, and maintainability.
The system allows hotel staff to add and manage rooms with details like room type, price, and availability, register guests with personal information securely stored, and schedule reservations while preventing double-booking. Guests' and rooms' information can be easily accessed and displayed using well-structured menus.
This system provides an efficient, interactive, and user-friendly console interface to manage multiple guests and rooms simultaneously, demonstrating real-world applications of OOP principles in software development. It serves as a foundation for more advanced hotel management systems with features like billing, online booking, and reports.

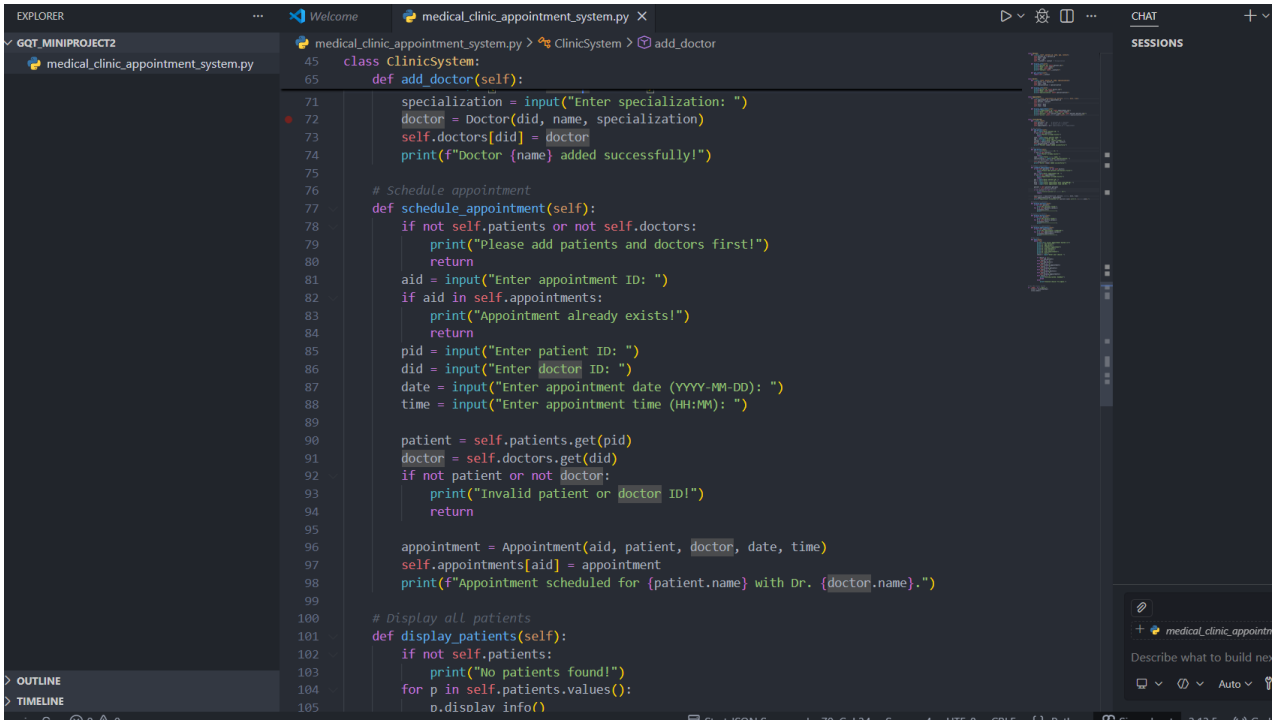
Source Code:-



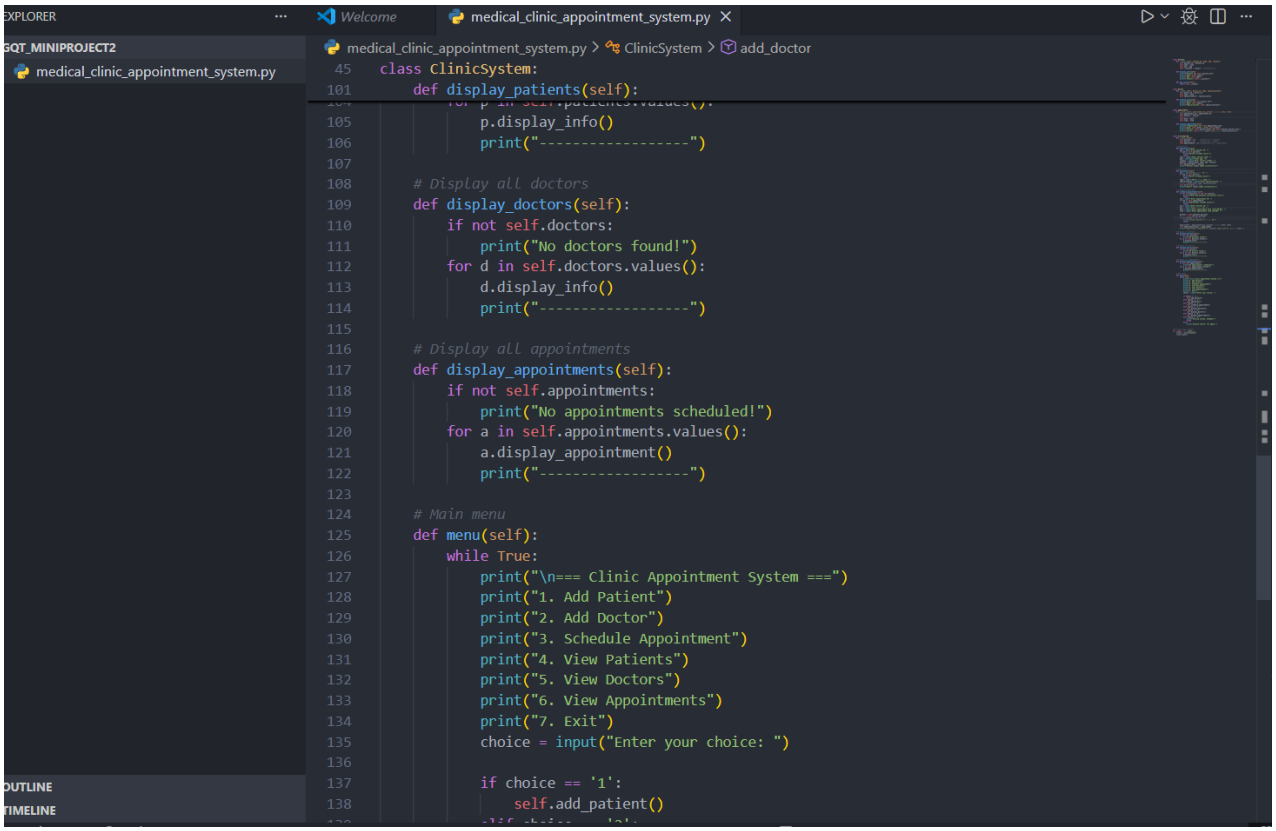

```python
class Patient:
    def __init__(self, patient_id, name, age, contact):
        self.patient_id = patient_id
        self.name = name
        self.age = age
        self.__contact = contact  # Encapsulation

    def display_info(self):
        print(f"Patient ID: {self.patient_id}")
        print(f"Name: {self.name}")
        print(f"Age: {self.age}")
        print(f"Contact: {self.__contact}")

    def get_contact(self):
        return self.__contact


class Doctor:
    def __init__(self, doctor_id, name, specialization):
        self.doctor_id = doctor_id
        self.name = name
        self.specialization = specialization

    def display_info(self):
        print(f"Doctor ID: {self.doctor_id}")
        print(f"Name: {self.name}")
        print(f"Specialization: {self.specialization}")


class Appointment:
    def __init__(self, appointment_id, patient, doctor, date, time):
        self.appointment_id = appointment_id
        self.patient = patient
        self.doctor = doctor
        self.date = date
        self.time = time
```



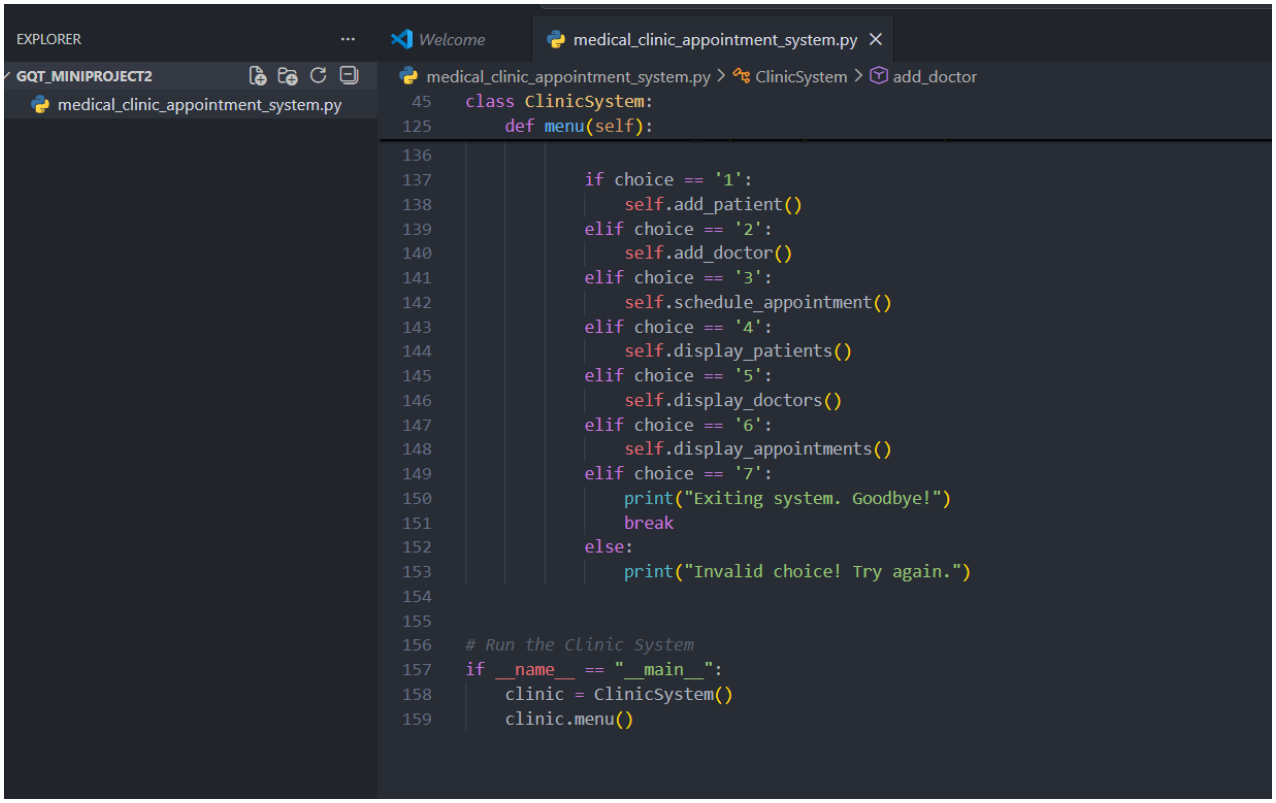
```
class Appointment:

    def display_appointment(self):
        print(f"\nAppointment ID: {self.appointment_id}")
        print(f"Date: {self.date} | Time: {self.time}")
        print(f"Patient: {self.patient.name} (ID: {self.patient.patient_id})")
        print(f"Doctor: {self.doctor.name} ({self.doctor.specialization})")


class ClinicSystem:
    def __init__(self):
        self.patients = {}      # patient_id -> Patient
        self.doctors = {}       # doctor_id -> Doctor
        self.appointments = {} # appointment_id -> Appointment

    # Add patient
    def add_patient(self):
        pid = input("Enter patient ID: ")
        if pid in self.patients:
            print("Patient already exists!")
            return
        name = input("Enter patient name: ")
        age = int(input("Enter age: "))
        contact = input("Enter contact number: ")
        patient = Patient(pid, name, age, contact)
        self.patients[pid] = patient
        print(f"Patient {name} added successfully!")

    # Add doctor
    def add_doctor(self):
        did = input("Enter doctor ID: ")
        if did in self.doctors:
            print("Doctor already exists!")
            return
        name = input("Enter doctor name: ")
        specialization = input("Enter specialization: ")
        doctor = Doctor(did, name, specialization)
```



```
class ClinicSystem:
    def add_doctor(self):
        specialization = input("Enter specialization: ")
        doctor = Doctor(did, name, specialization)
        self.doctors[did] = doctor
        print(f"Doctor {name} added successfully!")

    # Schedule appointment
    def schedule_appointment(self):
        if not self.patients or not self.doctors:
            print("Please add patients and doctors first!")
            return
        aid = input("Enter appointment ID: ")
        if aid in self.appointments:
            print("Appointment already exists!")
            return
        pid = input("Enter patient ID: ")
        did = input("Enter doctor ID: ")
        date = input("Enter appointment date (YYYY-MM-DD): ")
        time = input("Enter appointment time (HH:MM): ")

        patient = self.patients.get(pid)
        doctor = self.doctors.get(did)
        if not patient or not doctor:
            print("Invalid patient or doctor ID!")
            return

        appointment = Appointment(aid, patient, doctor, date, time)
        self.appointments[aid] = appointment
        print(f"Appointment scheduled for {patient.name} with Dr. {doctor.name}.")

    # Display all patients
    def display_patients(self):
        if not self.patients:
            print("No patients found!")
        for p in self.patients.values():
            p.display_info()
```

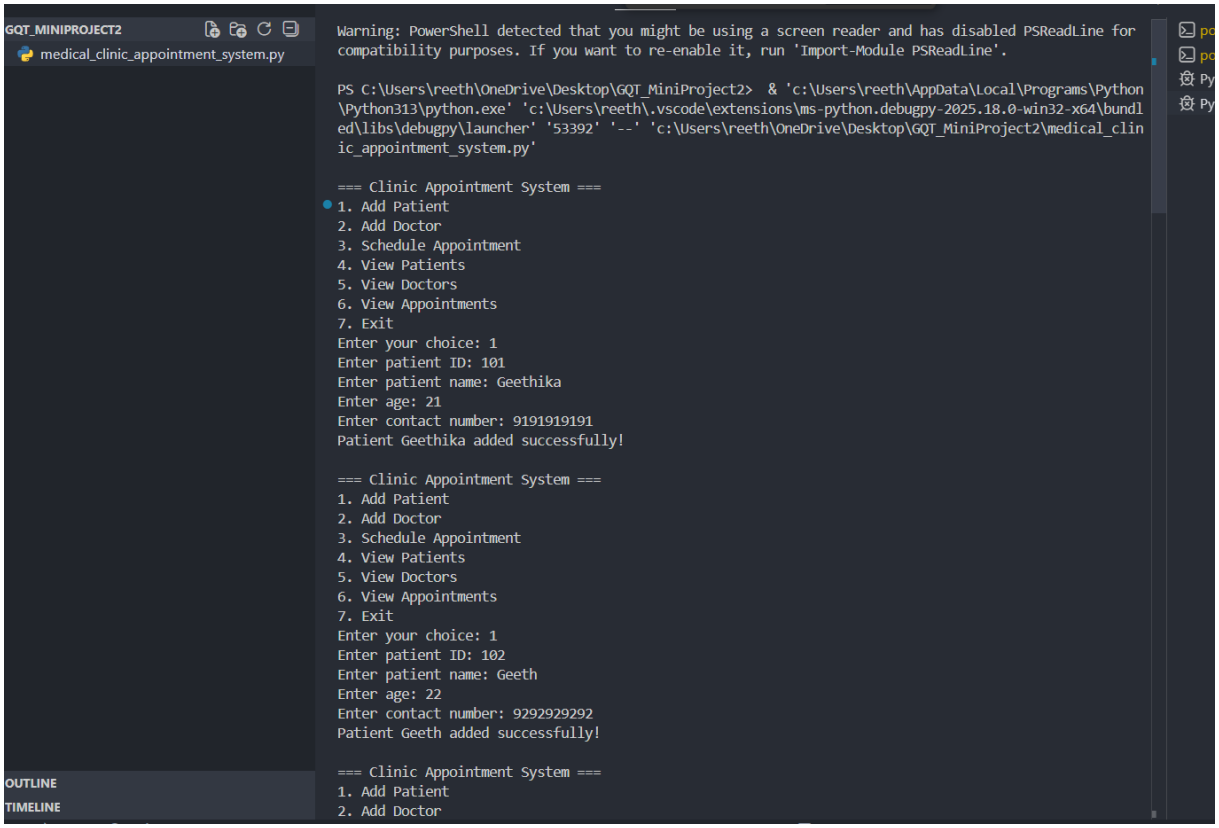


```
EXPLORER
GQT_MINIPROJECT2
medical_clinic_appointment_system.py
Welcome
medical_clinic_appointment_system.py
medical_clinic_appointment_system.py > ClinicSystem > add_doctor
45  class ClinicSystem:
101     def display_patients(self):
105             p.display_info()
106             print("------------------")
107
108     # Display all doctors
109     def display_doctors(self):
110         if not self.doctors:
111             print("No doctors found!")
112         for d in self.doctors.values():
113             d.display_info()
114             print("------------------")
115
116     # Display all appointments
117     def display_appointments(self):
118         if not self.appointments:
119             print("No appointments scheduled!")
120         for a in self.appointments.values():
121             a.display_appointment()
122             print("------------------")
123
124     # Main menu
125     def menu(self):
126         while True:
127             print("\n=== Clinic Appointment System ===")
128             print("1. Add Patient")
129             print("2. Add Doctor")
130             print("3. Schedule Appointment")
131             print("4. View Patients")
132             print("5. View Doctors")
133             print("6. View Appointments")
134             print("7. Exit")
135             choice = input("Enter your choice: ")
136
137             if choice == '1':
138                 self.add_patient()
OUTLINE
TIMELINE
```

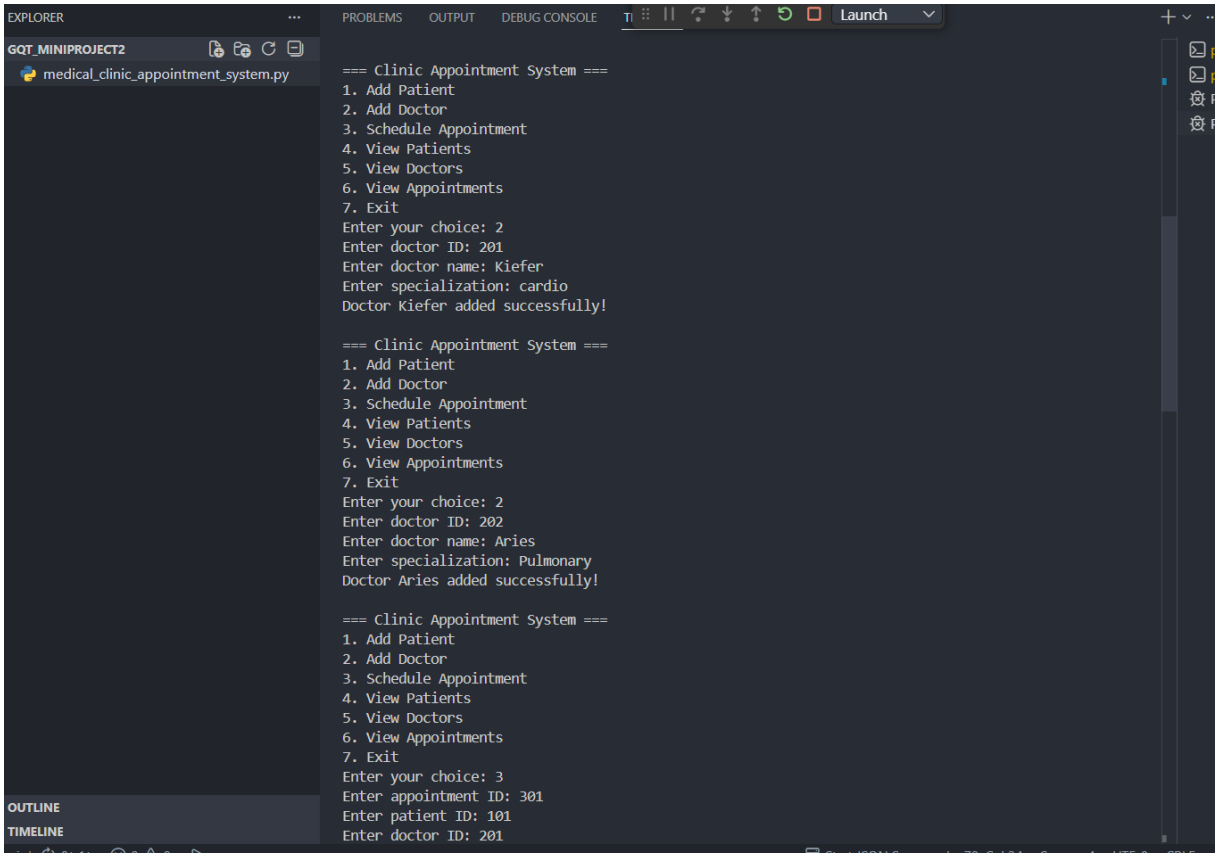


```
EXPLORER
GQT_MINIPROJECT2
medical_clinic_appointment_system.py
Welcome
medical_clinic_appointment_system.py
medical_clinic_appointment_system.py > ClinicSystem > add_doctor
45  class ClinicSystem:
125     def menu(self):
136
137             if choice == '1':
138                 self.add_patient()
139             elif choice == '2':
140                 self.add_doctor()
141             elif choice == '3':
142                 self.schedule_appointment()
143             elif choice == '4':
144                 self.display_patients()
145             elif choice == '5':
146                 self.display_doctors()
147             elif choice == '6':
148                 self.display_appointments()
149             elif choice == '7':
150                 print("Exiting system. Goodbye!")
151                 break
152             else:
153                 print("Invalid choice! Try again.")
154
155
156 # Run the Clinic System
157 if __name__ == "__main__":
158     clinic = ClinicSystem()
159     clinic.menu()
```

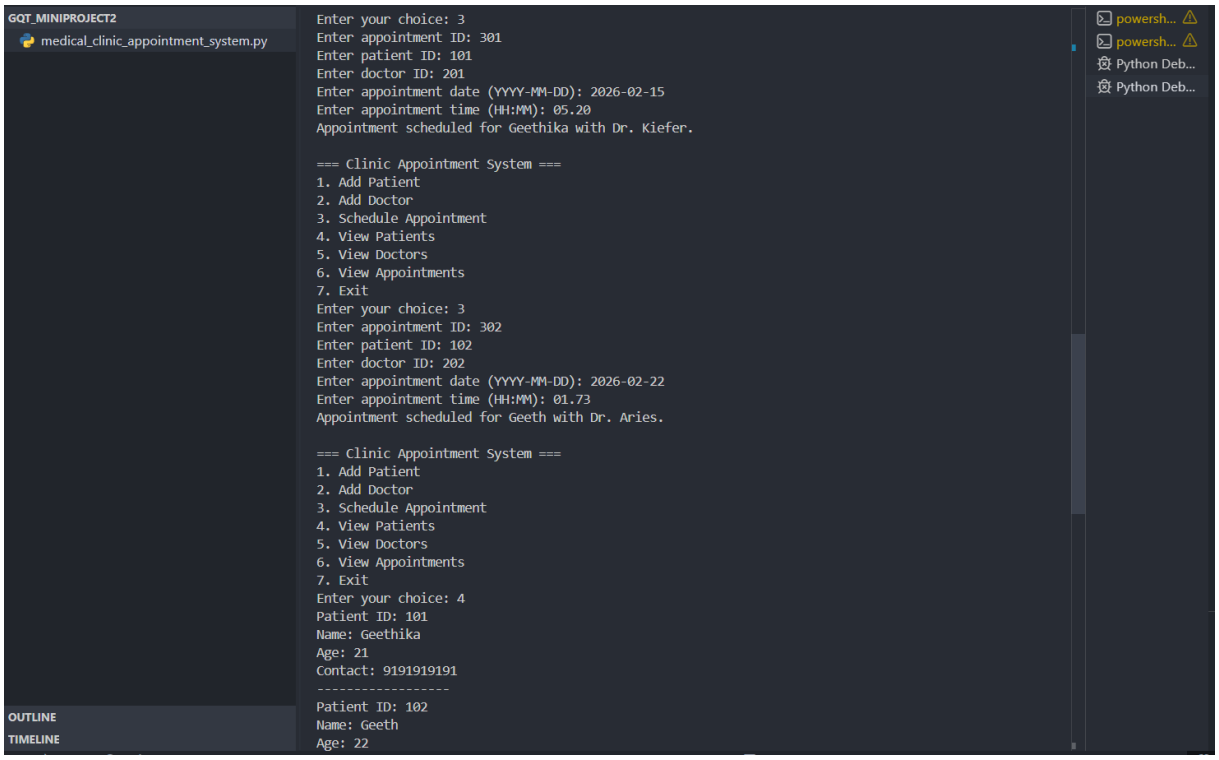
Output:-

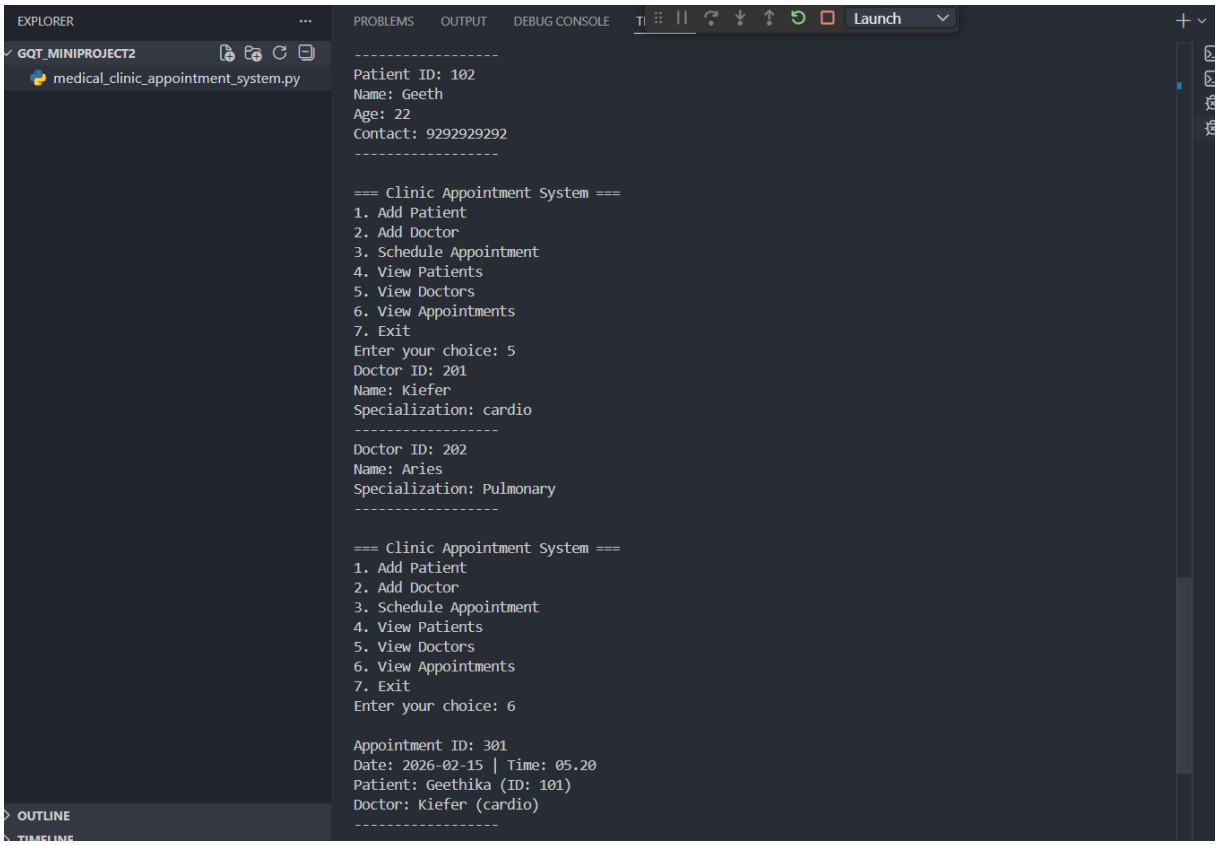


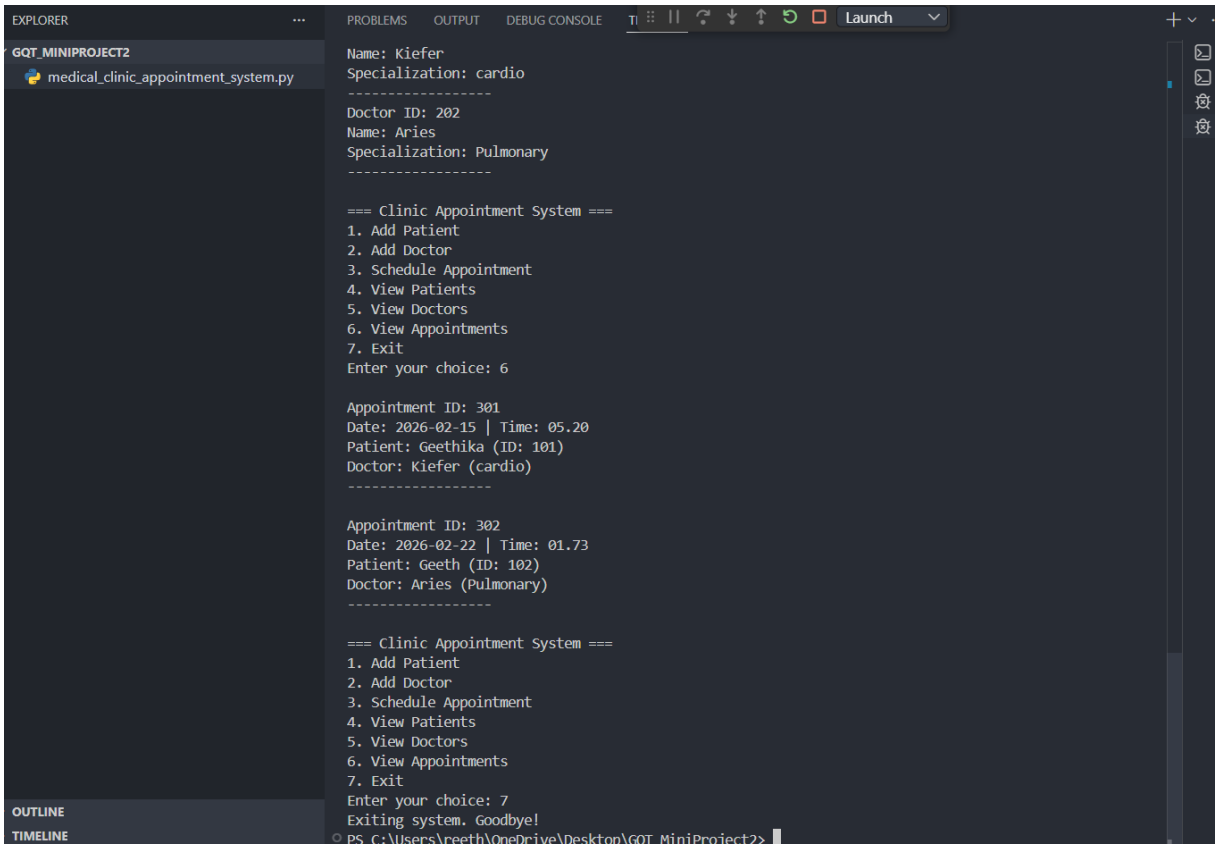
```
GQT_MINIPROJECT2
medical_clinic_appointment_system.py

Warning: PowerShell detected that you might be using a screen reader and has disabled PSReadLine for
compatibility purposes. If you want to re-enable it, run 'Import-Module PSReadLine'.

PS C:\Users\reeth\OneDrive\Desktop\GQT_MiniProject2>  & 'c:\Users\reeth\AppData\Local\Programs\Python
\Python313\python.exe' 'c:\Users\reeth\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundl
ed\libs\debugpy\launcher' '53392' '--' 'c:\Users\reeth\OneDrive\Desktop\GQT_MiniProject2\medical_clin
ic_appointment_system.py'

=== Clinic Appointment System ===
1. Add Patient
2. Add Doctor
3. Schedule Appointment
4. View Patients
5. View Doctors
6. View Appointments
7. Exit
Enter your choice: 1
Enter patient ID: 101
Enter patient name: Geethika
Enter age: 21
Enter contact number: 9191919191
Patient Geethika added successfully!

=== Clinic Appointment System ===
1. Add Patient
2. Add Doctor
3. Schedule Appointment
4. View Patients
5. View Doctors
6. View Appointments
7. Exit
Enter your choice: 1
Enter patient ID: 102
Enter patient name: Geeth
Enter age: 22
Enter contact number: 9292929292
Patient Geeth added successfully!

=== Clinic Appointment System ===
1. Add Patient
2. Add Doctor

OUTLINE
TIMELINE
```



```
EXPLORER
GQT_MINIPROJECT2
medical_clinic_appointment_system.py

PROBLEMS  OUTPUT  DEBUG CONSOLE  TI  Launch

=== Clinic Appointment System ===
1. Add Patient
2. Add Doctor
3. Schedule Appointment
4. View Patients
5. View Doctors
6. View Appointments
7. Exit
Enter your choice: 2
Enter doctor ID: 201
Enter doctor name: Kiefer
Enter specialization: cardio
Doctor Kiefer added successfully!

=== Clinic Appointment System ===
1. Add Patient
2. Add Doctor
3. Schedule Appointment
4. View Patients
5. View Doctors
6. View Appointments
7. Exit
Enter your choice: 2
Enter doctor ID: 202
Enter doctor name: Aries
Enter specialization: Pulmonary
Doctor Aries added successfully!

=== Clinic Appointment System ===
1. Add Patient
2. Add Doctor
3. Schedule Appointment
4. View Patients
5. View Doctors
6. View Appointments
7. Exit
Enter your choice: 3
Enter appointment ID: 301
Enter patient ID: 101
Enter doctor ID: 201

OUTLINE
TIMELINE
```


GQT_MINIPROJECT2
medical_clinic_appointment_system.py
OUTLINE
TIMELINE
Enter your choice: 3
Enter appointment ID: 301
Enter patient ID: 101
Enter doctor ID: 201
Enter appointment date (YYYY-MM-DD): 2026-02-15
Enter appointment time (HH:MM): 05.20
Appointment scheduled for Geethika with Dr. Kiefer.
=== Clinic Appointment System ===
1. Add Patient
2. Add Doctor
3. Schedule Appointment
4. View Patients
5. View Doctors
6. View Appointments
7. Exit
Enter your choice: 3
Enter appointment ID: 302
Enter patient ID: 102
Enter doctor ID: 202
Enter appointment date (YYYY-MM-DD): 2026-02-22
Enter appointment time (HH:MM): 01.73
Appointment scheduled for Geeth with Dr. Aries.
=== Clinic Appointment System ===
1. Add Patient
2. Add Doctor
3. Schedule Appointment
4. View Patients
5. View Doctors
6. View Appointments
7. Exit
Enter your choice: 4
Patient ID: 101
Name: Geethika
Age: 21
Contact: 9191919191
Patient ID: 102
Name: Geeth
Age: 22
powersh...
powersh...
Python Deb...
Python Deb...


EXPLORER
GQT_MINIPROJECT2
medical_clinic_appointment_system.py
OUTLINE
PROBLEMS
OUTPUT
DEBUG CONSOLE
Launch
Patient ID: 102
Name: Geeth
Age: 22
Contact: 9292929292
=== Clinic Appointment System ===
1. Add Patient
2. Add Doctor
3. Schedule Appointment
4. View Patients
5. View Doctors
6. View Appointments
7. Exit
Enter your choice: 5
Doctor ID: 201
Name: Kiefer
Specialization: cardio
Doctor ID: 202
Name: Aries
Specialization: Pulmonary
=== Clinic Appointment System ===
1. Add Patient
2. Add Doctor
3. Schedule Appointment
4. View Patients
5. View Doctors
6. View Appointments
7. Exit
Enter your choice: 6
Appointment ID: 301
Date: 2026-02-15 | Time: 05.20
Patient: Geethika (ID: 101)
Doctor: Kiefer (cardio)


EXPLORER
GQT_MINIPROJECT2
medical_clinic_appointment_system.py
OUTLINE
TIMELINE
PROBLEMS
OUTPUT
DEBUG CONSOLE
Launch
Name: Kiefer
Specialization: cardio
Doctor ID: 202
Name: Aries
Specialization: Pulmonary
=== Clinic Appointment System ===
1. Add Patient
2. Add Doctor
3. Schedule Appointment
4. View Patients
5. View Doctors
6. View Appointments
7. Exit
Enter your choice: 6
Appointment ID: 301
Date: 2026-02-15 | Time: 05.20
Patient: Geethika (ID: 101)
Doctor: Kiefer (cardio)
Appointment ID: 302
Date: 2026-02-22 | Time: 01.73
Patient: Geeth (ID: 102)
Doctor: Aries (Pulmonary)
=== Clinic Appointment System ===
1. Add Patient
2. Add Doctor
3. Schedule Appointment
4. View Patients
5. View Doctors
6. View Appointments
7. Exit
Enter your choice: 7
Exiting system. Goodbye!
PS C:\Users\reeth\OneDrive\Desktop\GQT MiniProject2>

GLOBAL QUEST
TECHNOLOGIES

fuel your
passion for
IT with
our **guidance.**

Global Quest Technologies

#324,2nd Floor, 3 A Cross, Near
Seshadripuram First Grade College,
Above City Union Bank,
Yelahanka New Town,
Bengaluru-560064

+91 9448 403 469 | 080-49720009
info@gqtech.in | www.gqtech.in