class name: ClinicManagerController
method signature: private void handleScheduleOffice()

Test Case #	Requirement	Test description and Input Data	Expected result/output
1	Ensure all required fields are filled	User clicks on "Submit" with all fields empty.	Error message "Please fill in all fields." should appear in the error label.
2	Validate date field format	User enters "hey" in the "Date" field and clicks "Submit".	Error message "Invalid date." should appear in the error label.
3	Validate date field	User enters "1/1/2023" in the "Date" field and clicks "Submit".	Error message "Invalid date." should appear in the error label.
4	Validate timeslot field range	User enters "13" in the "Timeslot" field and clicks "Submit".	Error message "Invalid timeslot." should appear in the error label.

5	Validate timeslot field format	User enters "hey" in the "Timeslot" field and clicks "Submit".	Error message "Invalid timeslot." should appear in the error label.
6	Validate dob	User enters "1/1/2025" in the "Date" field and clicks "Submit".	Error message "Invalid dob." should appear in the error label.
7	Handle invalid NPI format	User enters "ABC123" in the "Doctor's NPI" field and clicks "Submit".	Error message "Invalid NPI. Please enter a numeric value." should appear in the error label.
8	Handle nonexistent NPI	User enters "2" in the "Doctor's NPI" field and clicks "Submit".	Error message "Doctor not found." should appear in the error label.
9	Handle doctor unavailability	User enters valid patient details and doctor NPI, but the doctor is not available for the selected date and timeslot.	Error message "Doctor not available" should appear in the error label.

10	Ensure appointment is added for valid fields	User enters valid details for patient, doctor, and timeslot, with no existing conflicts.	Appointment is successfully added, and a success message "Appointment submitted!" should be shown.

class name: ClinicManagerController

method signature: private void handleScheduleImaging()

Test Case #	Requirement	Test description and Input Data	Expected result/output
1	Ensure appointment is added for valid fields	User enters valid details for patient, date, and timeslot, with no existing conflicts.	Appointment is successfully added, and a success message "Appointment submitted!" should be shown.

class name: ClinicManagerController

method signature: private void handleCancelAppointment()

Test Case #	Requirement	Test description and Input Data	Expected result/output
1	Ensure appointment is cancelled for valid fields	User enters valid details for patient, date, and timeslot, with no existing conflicts.	Appointment is successfully added, and a success message "Appointment cancelled!"
2	Handle nonexistent appointment	User enters valid patient details and time, but no appointment exists at that time.	Error message "Appointment does not exist." should appear in the error label.

class name: ClinicManagerController

method signature: private void handleRescheduleAppointment()

Test Case #	Requirement	Test description and Input Data	Expected result/output
1	Ensure appointment is rescheduled for valid fields	User enters valid details for patient, date, and timeslots, with no existing conflicts.	Appointment is successfully rescheduled, and a success message "Appointment rescheduled!"

class name: ClinicManagerController

method signature: private void handlePA()

Test Case #	Requirement	Test description and Input Data	Expected result/output
1	Handle appointment display with no appointments	User selects "Display Appointments by Date (PA) from main dashboard while no appointments are existent	An alert stating "No Appointments Available" pops up.