Practical Task: Doctor Direct Platform Testing

Objective:

Test the consultation booking and scheduling functionality of the Doctor Direct platform. Ensure the process works correctly, from logging in to booking and managing consultations. Candidates should automate the tests, execute them, and document their findings.

Platform Link:

- Doctor Direct Platform Login

Instructions:

1. **Environment Setup**
    - Set up your testing environment, including any tools, frameworks, or libraries you plan to use for automation (e.g., Selenium, Cypress, Playwright).
    - Document the setup steps in your submission so that the environment can be replicated if needed.
2. **Login Functionality**
    - Automate test cases for the login functionality, including scenarios for:
        - **Successful login** (valid username and password)
        - **Failed login** (invalid username or password)
        - **Edge cases** (e.g., empty fields, special characters in input fields)
3. **Consultation Booking and Scheduling**
    - After logging in, test the following key features related to booking consultations:
        - **Book a Consultation**: Automate the steps to schedule a consultation with a doctor. Verify that:
            - A confirmation is displayed upon booking.
            - The consultation details (doctor name, date, time) are accurate.
4. **Test Scenarios**
    - Design and document at least **7 test cases** that include positive, negative, and edge cases for booking, canceling, and rescheduling consultations.
5. **Automation Requirements**
    - Write scripts to automate each test case, ensuring:
        - Tests are independent and repeatable.
        - Proper assertions are in place to verify expected outcomes.
        - Errors or unexpected behaviors are logged clearly.
        - Screenshots are captured at key steps or upon test failures.
6. **Execution and Reporting**
    - Execute your test suite and document the results.
    - Submit a **test summary report** that includes:

        - Test case names, descriptions, and expected outcomes
        - Results (pass/fail) with explanations for failures, if any
        - A list of any issues or bugs discovered, with reproduction steps

7. **Optional Enhancements**
    - Implement a data-driven approach to run tests with multiple inputs.
    - Add tests to check UI responsiveness on different screen sizes if time allows.

Submission Requirements:

By Friday, please submit:

- Source code for your automated tests
- Instructions to set up and run your tests
- Test summary report
- Any screenshots, logs, or additional supporting files