Technical Test

This exam is a small exercise for you to work on for the next few days. It will help us understand how you assure quality in products. At LEAP, we expect solid skills from our Quality Engineering team so this will help us better understand where your skills lie.

Task 1

This task refers to this website: https://demoqa.com/login

Your task is to write test-cases, in descriptive language, to test the **Book Store Application** pages of the above site as thoroughly as possible. You should cover the Login, Book Store and Profile sections of this application. You can use any formalised test-case template or come up with a non-standard way of documenting the test cases.

Hints

- Thoroughly testing does not necessarily mean creating lots of test cases
- Think testing levels
- Efficiency is key

Task 2

Rest API – Automated Testing

Using any API Testing tool of your choice ***OR*** if you really prefer, write your own code with any language, and create an automated API test suite utilising **ALL** the CRUD endpoints listed on https://crudcrud.com/

These endpoints are good for 24 hours with a dynamic unique id in the URL. Please factor this into your design so it can be easily run when the id changes. For this exercise, we want you build an API Suite that performs (C)reate, (R)ead, (U)pdate & (D)elete operations where the "resource" is an **Employee**. The following Employee properties should be included:

FirstName, LastName, DateOfBirth, StartDate, Department, JobTitle, Email, Mobile, Address & BaseSalary.

It's completely upto you on how you structure the data, but we do prefer JSON format.

The automated API test suite should demonstrate the following features:

- Perform schema validation on the response payload.
- Have a data driven approach using an appropriate data format to drive the tests.
- Include both positive and negative test scenarios
- Passing of data values extracted from one endpoint to another, so the data flows through the endpoints.
- Use design patterns to create test code that demonstrates maintainability, robustness, and repeatability as key design considerations.
- Provides test results with clear and concise assertion descriptions.
- Appropriate error handling in the case you encounter bugs!

Remember to push your code and related files to an online repository so we, at LEAP Dev, can download, review, and execute your code that you have spent time creating. If you have used a test tool, work out a way to export the tests to the repository and **make sure** to provide enough information about how the tests can be re-imported & executed.

Please let the interviewer know what the repository's URL is and provide proper permissions.