**SDET/Automation Technical Test - Web Applications**

**Purpose**

As part of our hiring process for technical roles, we require all candidates to complete a small technical exercise. This is an integral part of our process to assess an individual’s technical ability when completing an activity in their own time.

Given the importance of this exercise to our hiring process, we highly recommend you give this adequate consideration and address this task as you would do any other professional assignment in your current/previous workplace.

**Instructions**

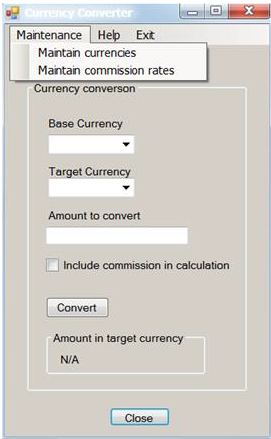
* There are two parts to this technical assessment
  + A written assessment of your ability to tackle a typical testing problem you might face as a test analyst
  + A code-based assessment, testing your ability to automate a test scenario
* Written answers for part 1 should be submitted in a suitable format - e.g. a word document
* Please work on a GitHub, create the New branch within this repository and when you are happy to submit, create a pull request back into the main branch
* Please use the pull request to comment on any aspects of your solution that you didn’t have time to complete, are not complete to your satisfaction or are not working
* Would you please add a markdown file with instructions on how to set up and run your solution to question 2

**Question 1**

**Introduction**

The following is the main form of an application that converts values from one currency to another. The application has the following characteristics:

* The collection of currencies, exchange rates, and rules for applying commission are held in a shared database, accessed by multiple users of this application on a network.
* The user of this form selects a base currency, a target currency, specifies an amount (expressed in the base currency), can choose to include or exclude commission, then hits the Convert button to generate the equivalent value in the target currency. That value is shown where “N/A” is displayed in the screenshot below.
* The separate “Maintain currencies” form (not shown) provides operations for adding, deleting and editing currencies.
* A separate process constantly updates the database with exchange rates for any recognised currencies (you do not need to worry about how this works, but you do have control over stopping and starting it).
* The separate “Maintain commission rates” form (not shown) provides operations for specifying a minimum commission value and a collection of commission rates for different ranges of values. The minimum commission value and ranges are defined in Sterling. For example, for “amount to convert” values in the range 0.01 to 100.00 pounds Sterling, the commission rate is 5%. An example minimum commission value is £10.



Given the information provided, please answer the following questions:

1. Using the database and the GUI, how would you test the functionality of this form?

Ans:

1. Validate all Base currency value which are available in the database should be present in the dropdown.
2. Validate all target currency value which are available in the database should be present in the dropdown.
3. Validate amount to convert should be same as target if user is unselect the checkbox for include commission in calculation.
4. Select the commission rate from database as per the currency select by customer. And valaidate the calculation.
5. What tests would you perform to test the operations supported by the “Maintain currencies” form?

Ans: We will do Database testing and Functional testing.

1. What tests would you perform to test the operations supported by the “Maintain commission rates” form?

Ans: We will do Database testing, Functional testing and Interface Testing

1. What tests would you perform on the form above in order to test accessibility?

Ans: Usability testing

1. If you were automating the testing of this form for regression test purposes, what would you hope the developer had done that would make the automation easier?

Ans: static xpath should be available.

1. If this application were used internationally, what further tests would you perform?

Ans: Performance testing

1. The screenshot of the form above shows what user interface issues are immediately apparent (you may have mentioned some of these in previous answers)?

Ans**: Timing/Performance Problems**

**Notes**

* In answering the questions, identifying tests is sufficient
* Details of test data and expected results are not required where they can be inferred
* Aim for an answer no longer than 1200 words for this entire section

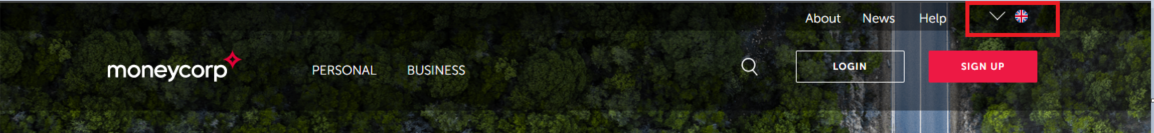
**Question 2**

**Introduction**

For this question, you can use any JavaScript framework of your choice. We want you to automate the scenarios below using the Chrome browser.

**Test Scenario**

1. Open the following URL: <https://www.moneycorp.com/en-gb/>
2. Change the language and region from the top right corner to USA (English). See example snippet.



1. Click Find out more for “Foreign exchange solutions” Validate if you have arrived on the page
2. Search for the word “international payments” using the search box
3. Validate if you have arrived on the result page
4. Validate that each article in the list displays a link that starts with <https://www.moneycorp.com/en-us/>

**Evaluation**

Evaluation will be done on the following criteria:

* Clean and tidiness of the solution
* Design and architecture
* Readability
* Maintainability
* Stability of the tests
* Clarity of setup and execution documentation

Would you please provide instructions for setting up an environment from scratch and how to run the tests?

**GIT URL :**

**Setting up an environment from scratch**

1. Download and Install Java
2. Download and Configure Intellij
3. Configure Selenium WebDriver
4. Set the environment variable
5. Open the Intellij
   1. Go to open 🡪 Create Project 🡪 Select Maven
   2. Click on Next button
   3. Provide the name of project and lick on Finish
   4. Add TestNG and Selenium dependency in POM.xml file

**Run The the test**

1. Download the project from the git or either you can select the project from git as well
2. Open the project in intellij and reload the project.
3. Execute the script by using TestNG or either by Mavan.