**Selenium Java Assessment**

**Project : 1 “₹”**

**Guidelines**:

A. Create a Java Project in Eclipse to perform the following tasks

B. Use Selenium Library to create the scripts

C. Organize your code into multiple functions

D. Upload the project into GitHub repository

**Requirement / User Story:**

As A user I should be able to search for a product, verify its discount and add it to cart, so that I can eventually buy the product

**Scenario / Test Case to Automate:**

1. Navigate to Flipkart.com

2. Close the popup - Use Xpath and the text() method in xpath to close

3. Search for text "fitbit" and click on search icon

4. Verify that there are more than 900 results shown

5. Click on the first result link. A new tab will open for the product details page

6. Take screenshot of this page (details page) and store it under project folder inside screenshots folder with filename productdetails.jpg

7. On this page, capture the original price and final price. Then calculate the discount percentage and verify that the % displayed on the page is correct.

8. Click on Add to Cart button. This will open the MyCart Page in the same window

9. Verify that the total amount is equal to the amount captured on Step 6 (final price)

10. Close the new window

11. Quit the drive

**Project : 2**

**Guidelines**:

A. Create a Java Project in Eclipse to perform the following tasks

B. Use MAVEN to create folder structure and resolve dependencies

C. Use TestNG for creating the program flow, tests, suites and assertions

1. Airline Ticket Booking (https://www.cleartrip.com/ )

**Steps to Automate:**

a. Navigate to URL

b. Select Round Trip

c. Select From: New Delhi To: Mumbai

d. Select Depart on 1st July and Return on 8th July

e. Click on the More options URL

f. Select Class of Travel as “Business”

g. Click on Search Flights

h. Take a screenshot of the results page

i. Show only non-stop flights

j. Take a screenshot of this page with only non-stop flights.

k. Find the number of Flights available in each leg (onward and return)

l. Capture the total price of the return ticket

m. Click on Book

n. In the new Window, verify that the price is same as in step j

o. Select Standard Fare option

p. Click on Continue

q. Verify that the total price is still the same as in step n & j.

**Project : 3**

**Guidelines:**

A. Create a Java Project in Eclipse to perform the following tasks

B. Use MAVEN to create folder structure and resolve dependencies

C. Use TestNG for creating the program flow, tests, suites and assertions

1. Shopping Workflow Automation ( http://automationpractice.com )

**Pre-Requisite: Create a login to this site manually**

**Steps to Automate:**

a. Navigate to the url

b. Search for word 'dress'

c. In 'Sort By' drop down, choose 'Price: Lowest First'

d. Hover over the first image in the results (Mouse over only, not Click)

e. Capture the price of this product

f. Click on the ‘More’ Button. It should open the details page.

g. Here, there are multiple images available. For each image, click on the image and take a screenshot and store it separately.

h. Capture the price from this page. Verify that it is same as in step e.

i. Click 'Add to Cart' button

j. On confirmation popup verify same product price is displayed. Capture total price. Click 'Proceed to checkout'

k. Shopping cart summary page is displayed. Validate Total price is same as in step j). Click Proceed to checkout

l. On Shipping address page, 'Click Proceed to checkout'

m. On Shipping page, Agree terms of service. Click 'Proceed to checkout'

n. On Payment page, confirm Total price again. Click on 'Pay by check'. Click on 'I confirm my order'

o. Validate 'Your order on My Store is complete.' msg is displayed. Capture order reference number

p. Sign out

**Project : 4**

**Guidelines:**

1.Create a Java Project in Eclipse to perform the following tasks

2. Use Selenium library to automate the UI

3. Use Synchronization where required

4. Use MAVEN to create folder structure and resolve dependencies

5. Use TestNG for creating the program flow, tests, suites and assertions

6. Use Data Provider wherever necessary

**Test Cases to Automate:**

**Part A: Test Case 1:**

1. Navigate to <https://www.bigbasket.com/>

2. Select city as ‘Bangalore’ from the dropdown at the top of page and continue

3. Add the first product in the Best Sellers area to the Basket by clicking on the ‘Add’ button.

4. Assert that the ‘Add’ button that was clicked above is no more visible.

5. Assert that the ‘My Basket’ view on the top right of the webpage is updated to show ‘1 items’.

6. Hover over the ‘My Basket’ view.

7. Assert that the same product that was selected in step 2 is added.

8. Click on the ‘View Basket and Checkout’ button.

9. Assert that the login popup is displayed.

10. Take a screenshot of this page.

11. Close the popup. Close the driver.

**Part B:**

**Test Data:**

|  |  |
| --- | --- |
| **Product Name** | **Quantity** |
| **White Sandwich Bread** | **1** |
| **Onions** | **2** |
| **Amul Butter** | **1** |
| **Coffee** | **3** |
| **Marie Gold Biscuits** | **2** |
| **Cadbury Dairy Milk** | **4** |

Create the above Test Data in excel sheet named as ‘Products’.

**Test Case 2:**

1. Navigate to <https://www.bigbasket.com/>

2. Select city as ‘Bangalore’ from the dropdown at the top of page and continue

3. Read test data from MongoDB collection ‘products’

4. Search for the product using the database field ‘Product Name’

5. On the results page, select the first product and add the quantity mentioned in the database field ‘Quantity’

6. Assert that the above product and quantity is added to the cart / Basket.

7. Repeat the above steps for each product.

**Part C:**

Create a TestNG Suite for all the projects / tests that you have created so far:

1. Flipkart workflow

2. Clear trip OR Automation Practice workflow

3. Big Basket Workflow

4. Run the Suite together and Save the HTML Result