# Playwright for Java - Hands-On Assignment 6

# Project Structure

# Problem Statement 1: Validating a Simple Interest Calculator with Playwright

## Objective:

Practice using Playwright's locators and assertions to automate and validate the functionality of a simple interest calculator web page.

## Steps:

1. Use the @UsePlaywright annotation to enable fixture-based injection of a BrowserContext object.
2. Write a test method that:
3. Sets a default assertion timeout for the test using PlaywrightAssertions.setDefaultAssertionTimeout(2000).
4. Creates a new Page instance from the BrowserContext.
5. Navigates to the simple interest calculator URL: <https://www.calculator.net/simple-interest-calculator.html>.
6. Locates and fills the principal amount field with "1000", setting a specific timeout of 5000ms for this action.
7. Locates and fills the interest rate field with "5", also with a timeout of 5000ms.
8. Locates and fills the time period field with "2" (for two years), also with a timeout of 5000ms.
9. Clicks the "Calculate" button, setting a specific timeout for this action.
10. Locates the End Balance result from the table and asserts that its text content is "$1,100.00".
11. Locates the Total Interest result from the table and asserts that its text content is "$100.00".

## Expected Outcome:

* The test should pass successfully, confirming that the calculator correctly computes the simple interest and end balance based on the provided inputs.
* The console output should show a successful test run, indicating that all assertions passed. The test verifies that for a principal of $1,000 at a 5% rate over 2 years, the total interest is $100.00 and the end balance is $1,100.00.

# Problem Statement 2: Testing with Disabled JavaScript in Playwright

## Objective:

Practice simulating a user with JavaScript disabled in their browser context and asserting that a web application correctly displays a warning message.

**Steps:**

1. Use the @UsePlaywright annotation to enable fixture-based injection of a Playwright object.

Write a test method that:

1. Launches a new browser instance.
2. Creates a new BrowserContext and explicitly disables JavaScript.
3. Creates a new Page instance within this context.
4. Navigates the page to a public URL (e.g., <https://www.enable-javascript.com/>). This site is designed to show different content based on whether JavaScript is enabled.
5. Locates the warning message element that appears when JavaScript is disabled.
6. Asserts that the warning message is both visible and contains the correct text.

## Expected Outcome:

* The test should pass successfully if:
* The test proves that Playwright can launch a browser with specific features disabled, such as JavaScript.
* The assertions confirm that the website's warning message for disabled JavaScript is correctly displayed, validating the application's behavior in a non-standard browser environment.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*