

EXPERIMENT-11

AIM: Write a simple program in JavaScript and perform testing using Selenium.

Objective: To develop automated test cases using Selenium for validating the functionality of a previously containerized web application.

Software Requirements

- Python or Node.js environment (as per project)
 - Selenium WebDriver installed
 - Web browser (Chrome, Firefox, etc.)
 - WebDriver for the browser (e.g., ChromeDriver)
 - Docker installed and running
 - Code editor (VS Code or any)
 - Sample containerized web application running
-

Lab Procedure

1. Open VS code and create html and javascript files
2. Create html file and paste this code

index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Sum Calculator</title>
<style>
body{
text-align: center;
}
</style>
</head>
<body>
```

```
<h1>Sum Calculator</h1>
<input type="number" id="num1" placeholder="Enter first number">
<input type="number" id="num2" placeholder="Enter second number">
<button id="add">Add</button>
<p>Result: <span id="result">0</span></p>
<script>
    function calculateSum(a, b)
    { return a + b;
    }
    document.getElementById('add').addEventListener('click', function() { const
    num1
    = parseInt(document.getElementById('num1').value, 10); const num2 =
    parseInt(document.getElementById('num2').value, 10); const result =
    calculateSum(num1, num2); document.getElementById('result').textContent =
    result;
    });
</script>
</body>
</html>
```

script.js

```
import { fileURLToPath } from 'url';
import { dirname } from 'path';
import { Builder, By, until } from 'selenium-webdriver';
import assert from 'assert';

// Define __dirname manually
const __filename = fileURLToPath(import.meta.url);
const __dirname = dirname(__filename);
```

```
async function runTest() {  
  // Define driver here  
  let driver = await new Builder().forBrowser('chrome').build();  
  
  try {  
    // Use __dirname (not dirname)  
    await driver.get('file://' + __dirname + '/index.html');  
  
    // Find input elements  
    const num1 = await driver.findElement(By.id('num1'));  
    await num1.sendKeys('50');  
  
    const num2 = await driver.findElement(By.id('num2'));  
    await num2.sendKeys('10');  
  
    // Click the "Add" button  
    const addButton = await driver.findElement(By.id('add'));  
    await addButton.click();  
  
    // Verify the result  
    const result = await driver.findElement(By.id('result'));  
    const text = await result.getText();  
  
    assert.strictEqual(text, '60', 'Sum calculation is incorrect');  
    console.log('✓ Test passed: Sum is correct');  
  } catch (error) {  
    console.error('✗ Test failed:', error);  
  } finally {  
    console.log('Press any key to exit...');
```

```
process.stdin.setRawMode(true);
process.stdin.resume();
process.stdin.on('data', async () => {
  await driver.quit();
  process.exit(0);
});
}
}
runTest();
```

3. Open terminal

4. npm install selenium-webdriver

```
E:\CMRCET20240105\III Yr\Devops Lab Exp Final\Exp-11>npm install selenium-webdriver
up to date, audited 18 packages in 6s
1 package is looking for funding
  run `npm fund` for details
found 0 vulnerabilities
```

5. node script.js

```
E:\CMRCET20240105\III Yr\Devops Lab Exp Final\Exp-11>node script.js
(node:10864) [MODULE_TYPELESS_PACKAGE_JSON] Warning: Module type of file:///E:/CMRCET20240105/III%20Yr/Devops%20Lab%20Exp%20Final/Exp-11/script.js is not specified and it doesn't parse as CommonJS.
Reparsing as ES module because module syntax was detected. This incurs a performance overhead.
To eliminate this warning, add "type": "module" to E:\CMRCET20240105\III Yr\Devops Lab Exp Final\Exp-11\package.json.
(Use `node --trace-warnings ...` to show where the warning was created)
DevTools listening on ws://127.0.0.1:51648/devtools/browser/b1b48b41-ebb4-4c3e-94a2-d307ea682426
Test passed: Sum is correct
Press any key to exit...
```

OUTPUT

