1. Automate interactions with a webpage, focusing on window handling and navigating between them using Playwright

```
tests > Week4Assignment > TS AutomateWindow.spec.ts > ⊙ test.only('automate window using Playwright') callback
1    import { test, expect } from "@playwright/test";
2    test.only('automate window using Playwright', async ({ context, page }) => {
3      // Step 1: Open the Salesforce login page
4      await page.goto("https://login.salesforce.com");
5      // Step 2: Login
6      await page.getByLabel('Username').fill('dilip@testleaf.com');
7      await page.getByLabel('Password').fill('Leaf@123');
8      await page.locator("#Login").click();
9      // Step 3: Wait for dashboard to load
10     await page.waitForTimeout(5000);
11     // Step 4: Click "Learn More" and wait for new window/tab using Promise.all
12     const [newPage] = await Promise.all([
13       context.waitForEvent('page'),                            // Wait for the new tab
14       page.locator("//button[contains(text(),'Learn More')]").click() // Trigger the tab
15     ]);
16     // Step 5: Ensure the new page is fully loaded
17     await newPage.waitForLoadState();
18     // Step 6: Capture and log the new window title
19     const popupTitle = await newPage.title();
20     console.log(`New Window Title: ${popupTitle}`);
21     // Step 7: Click the 'Confirm' button
22     await newPage.locator("//button[text()='Confirm']").click();
23     // Step 8: Wait again in case it triggers navigation
24     await newPage.waitForLoadState();
25     // Step 9: Capture final title and URL
26     const finalTitle = await newPage.title();
27     const finalURL = newPage.url();
28     console.log(`Final Title: ${finalTitle}`);
29     console.log(`Final URL: ${finalURL}`);
30     // Step 10: Assertions to validate page title and URL
31     expect(finalTitle).toContain("Salesforce");
32     expect(finalURL).toContain("salesforce");
33   });
```

```
1 passed (16.9s)
● PS C:\Playwright-workspace\Playwright-Testleaf\PlaywrightJune2025\tests\Week4Assignment> npx playwright test AutomateWindow.spec.ts

  Running 1 test using 1 worker

    ✓  1 AutomateWindow.spec.ts:2:6 › automate window using Playwright (13.1s)
  New Window Title:
  Final Title: Service Cloud: AI-powered Customer Service Agent Console | Salesforce US
  Final URL: https://www.salesforce.com/service/cloud/?bc=HB

    1 passed (14.3s)
○ PS C:\Playwright-workspace\Playwright-Testleaf\PlaywrightJune2025\tests\Week4Assignment> ▮
```

2. Task is to merge the Accounts created

```typescript
tests > Week4Assignment > TS MergeCreatedAccounts.spec.ts > ...
 1   import { test, expect } from '@playwright/test';
 2   test('Leaftaps.com - Merge the Accounts created', async ({ page, context }) => {
 3       // Step 1: Launch application
 4       await page.goto('http://leaftaps.com/opentaps/control/main');
 5       // Step 2: Login
 6       await page.locator('#username').fill('DemoCSR2');
 7       await page.locator('#password').fill('crmsfa');
 8       await page.locator('.decorativeSubmit').click();
 9       // Step 3: Navigate to CRM/SFA > Accounts > Merge Accounts
10       await page.locator('text=CRM/SFA').click();
11       await page.locator('//a[text()="Accounts"]').click();
12       await page.locator('//a[text()="Merge Accounts"]').click();
13       // ========== FROM Account ==========
14       const [fromPopup] = await Promise.all
15       ([
16           context.waitForEvent('page'),
17           page.locator('(//img[@alt="Lookup"])[1]').click() // First widget (From Account)
18       ]);
19       await fromPopup.waitForLoadState();
20       await fromPopup.locator('(//a[@class="linktext"])[1]').click(); // Select first result
21       await fromPopup.close();
22       // ========== TO Account ==========
23       const [toPopup] = await Promise.all
24       ([
25           context.waitForEvent('page'),
26           page.locator('(//img[@alt="Lookup"])[2]').click() // Second widget (To Account)
27       ]);
28       await toPopup.waitForLoadState();
29       await toPopup.getByText(/^10165$/, { exact: true }).click();
30       await toPopup.close();
31       // Step 4: Click Merge
32       await page.locator('//a[text()="Merge"]').click();
33       // Step 5: Handle the confirmation alert
34       const dialog = await page.waitForEvent('dialog');
35       console.log('Alert message:', dialog.message());
36       console.log('Alert type:', dialog.type());
37       await dialog.accept();
```

```typescript
36       console.log('Alert type:', dialog.type());
37       await dialog.accept();
38       // Step 6: Wait for navigation to complete
39       await page.waitForLoadState('load');
40       // Step 7 : Verify the title
41       await expect(page).toHaveTitle('Account Details | opentaps CRM');
42   });
```

```
PS C:\Playwright-workspace\Playwright-Testleaf\PlaywrightJune2025> npx playwright test MergeCreatedAccounts.spec.ts

Running 1 test using 1 worker
[chromium] > tests\Week4Assignment\MergeCreatedAccounts.spec.ts:2:5 > Leaftaps.com - Merge the Accounts created
Alert message: Are you sure?
Alert type: confirm
  1 passed (16.5s)

  Serving HTML report at http://localhost:9323. Press Ctrl+C to quit.
```
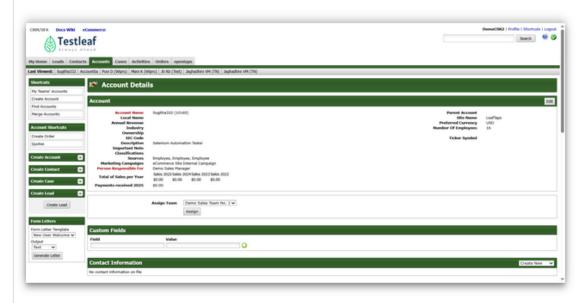
Leaftaps.com - Merge the Accounts created

Week4Assignment/MergeCreatedAccounts.spec.ts:2   15.2s

`chromium`

✓ Run

✓ Test Steps	
> ✓ Before Hooks	329ms
> ✓ Navigate to "/opentaps/control/main" — Week4Assignment/MergeCreatedAccounts.spec.ts:4	1.0s
> ✓ Fill "DemoCSR2" locator('#username') — Week4Assignment/MergeCreatedAccounts.spec.ts:6	42ms
> ✓ Fill "crmsfa" locator('#password') — Week4Assignment/MergeCreatedAccounts.spec.ts:7	23ms
> ✓ Click locator('.decorativeSubmit') — Week4Assignment/MergeCreatedAccounts.spec.ts:8	243ms
> ✓ Click locator('text=CRM/SFA') — Week4Assignment/MergeCreatedAccounts.spec.ts:10	407ms
> ✓ Click locator('//a[text()="Accounts"]') — Week4Assignment/MergeCreatedAccounts.spec.ts:11	1.1s
> ✓ Click locator('//a[text()="Merge Accounts"]') — Week4Assignment/MergeCreatedAccounts.spec.ts:12	518ms
> ✓ Wait for event "page" — Week4Assignment/MergeCreatedAccounts.spec.ts:16	468ms
> ✓ Click locator('xpath=(//img[@alt="Lookup"])[1]') — Week4Assignment/MergeCreatedAccounts.spec.ts:17	365ms
> ✓ Wait for load state "load" — Week4Assignment/MergeCreatedAccounts.spec.ts:19	347ms
> ✓ Click locator('xpath=(//a[@class="linktext"])[1]') — Week4Assignment/MergeCreatedAccounts.spec.ts:20	78ms
> ✓ Wait for event "page" — Week4Assignment/MergeCreatedAccounts.spec.ts:25	261ms
> ✓ Click locator('xpath=(//img[@alt="Lookup"])[2]') — Week4Assignment/MergeCreatedAccounts.spec.ts:26	83ms
> ✓ Wait for load state "load" — Week4Assignment/MergeCreatedAccounts.spec.ts:28	187ms
> ✓ Click getByText(/^10165$/) — Week4Assignment/MergeCreatedAccounts.spec.ts:29	207ms
> ✓ Click locator('//a[text()="Merge"]') — Week4Assignment/MergeCreatedAccounts.spec.ts:32	70ms
> ✓ Wait for event "dialog" — Week4Assignment/MergeCreatedAccounts.spec.ts:34	5.8s
> ✓ Accept dialog — Week4Assignment/MergeCreatedAccounts.spec.ts:37	4ms
> ✓ Wait for load state "load" — Week4Assignment/MergeCreatedAccounts.spec.ts:39	1ms
> ✓ Expect "toHaveTitle" — Week4Assignment/MergeCreatedAccounts.spec.ts:41	3.7s
> ✓ After Hooks	📄 539ms

Screenshots

screenshot